How Causality Bridges the Semantic Gap

Anonymous authors
Paper under double-blind review

Abstract

Numerical measurements capture how a system behaves, but often leave the meanings of its variables unspecified. Some variables are measured but never labeled, and others are never measured at all. Existing methods assign semantics to such variables by consulting general human knowledge, but this inherits its biases where that knowledge exists and offers nothing where it does not. We bridge this gap between measurements and their meanings with causal structure instead, reading a variable's semantics from how it acts on other variables. We formalize this as structure-constrained semantic alignment, in which the embedding of each unnamed variable is solved under the dependence relations implied by the causal graph, with the embeddings of a few known names as anchors. Accordingly, we build **CausalBridge**, a framework that discovers the causal graph from the measurements, latent variables included, solves for the embeddings under those relations, and expresses them as names through a language model. The causal structure reflects the mechanism that generated the measurements and is recovered from the measurements alone, which may make it the one source of information free of bias from human knowledge. We evaluate CausalBridge on five questionnaires and three robotics scenarios, with 20 to 90% of the variable names masked. It recovers the semantics of observed and latent variables more accurately than existing methods that rely on association, and its lead widens as less of the system is documented. The graph it discovers names variables as accurately as the documented one, and a new system is named in minutes and at a fraction of the cost of sampling methods. Once the semantic gap is bridged faithfully, machines can understand the world and take actions causally.

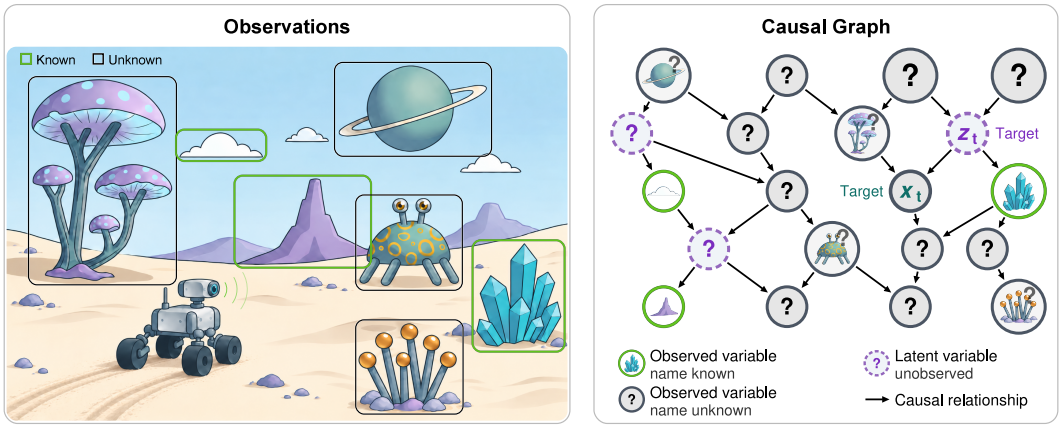

Figure 1: A system meets a world it has no words for. Left: a rover on a new planet, whose sensors return numbers for everything in view, can name a few things (green boxes) and has no name for most (gray boxes), and nobody wrote a manual for this world. Right: the causal graph that the numbers alone reveal, with latent variables (dashed purple) never measured yet acting on what is. We ask how this structure, anchored by the few known names, lets the system name the unnamed observed variable x_t and the latent variable z_t. The arrows are the relations that the numbers alone reveal, and the question marks stand for the names the system does not have.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

1 INTRODUCTION

An intelligent system sees the real world as numbers, while people understand the same world with semantics, and between the two sit unnamed variables with no semantics assigned. Some are measured but never documented, as in a shared corpus of robot demonstrations whose observation and action spaces differ from robot to robot (O'Neill et al., 2024), so the meaning of each channel must be found again for every robot. Others, such as the latent variable behind a group of questionnaire items and the feature inside a network that a set of activations reflects, are never measured at all and appear only through their effect on what is measured. Causal discovery can recover such a latent variable from data (Silva et al., 2006), but only as a numerical variable with a structural position in a graph and no name (Figure 1). A name turns that variable into a hypothesis a person can test or a concept a system can communicate. This semantic gap appears where human knowledge covers only a few of the variables, and closing it lets a system know the world it measures beyond what people have already written down. We therefore ask how a system can come to know such a world from its data alone, with few priors.

Existing strategies assign semantics to an unnamed variable by reading human knowledge. Some propagate semantics from named neighbors over a supplied graph (Rossi et al., 2022; Hou et al., 2022). Others match the activation profile of the variable against a fixed concept bank (Bau et al., 2017; Oikarinen & Weng, 2023). Still others ask a language model what the active cases of the variable suggest (Bills et al., 2023; Paulo et al., 2025; Singh et al., 2023), or what the variables around a latent variable in a recovered graph suggest (Du et al., 2026). Causal discovery and causal representation learning recover the structure among observed and latent variables and state when the latent ones are identifiable (Kummerfeld & Ramsey, 2016; Xie et al., 2020; Schölkopf et al., 2021; Khemakhem et al., 2020; Zheng et al., 2022). ThoughtComm (Zheng et al., 2025) carries recovered latent variables into a language model, but none of these methods names what it recovers.

One instance shows where these strategies fail (Figure 2(b)). Consider a drive recorded from the CAN bus of a car (Schafer et al., 2018), on which the car's controllers exchange messages. Six of its twelve channels are named, and one of the unnamed ones is the longitudinal acceleration. AutoInterp names it pitch rate and gives the same name to another unnamed channel, while the SASC-style method names it front left wheel speed and gives that name to three other unnamed channels. Neither name is correct. The unnamed channels carry different signals, yet each method gives one name to several of them. The name therefore reflects the language prior about what a car records, and the measurements have little effect on it. Where human knowledge covers the variables, the inferred name inherits its biases, and where it does not, there is nothing to retrieve.

How can a system recover the semantics of the variables it measures, where human knowledge covers few of them?

We answer this question with **CAUSALBRIDGE**, which first recovers the causal structure of a system from its measurements alone, without reading any name, and adds a latent variable wherever the dependence among the observed variables requires one (Figure 2b). It then solves for the meaning of every unnamed variable under that structure, where the vectors of the few named variables serve as anchors and the relations determine every unnamed vector relative to them. Finally, a frozen language model translates each vector into a name, and existing semantics enter at this step. In the graph discovered for that drive, the vertical acceleration causes the unnamed channel, which in turn causes the pitch rate. Both arise when the car brakes or accelerates, so the channel between them is its acceleration along the road, and CAUSALBRIDGE writes longitudinal acceleration. The structure supplies what human knowledge does not cover, and the language model supplies the words.

This paper makes four contributions. **First**, we *discover the semantics of a system from its causal structure*, and the language model only writes the name. A system can therefore come to know the world it measures from its own data. **Second**, we formalize the task as *structure-constrained semantic alignment*. In this task, the embedding of every unnamed variable, observed or latent, is solved under the relations that the causal graph implies. The few known names serve as anchors. **Third**, we realize the alignment as CAUSALBRIDGE, which is *a framework with discovered structure, causally grounded names and efficient deployment*. **Fourth**, we evaluate CAUSALBRIDGE on three robotics scenarios and five questionnaires with 20 to 90% of the true names masked. We show that *the lead is largest where the least has been written down*. At 50% masking, CAUSALBRIDGE outperforms the strongest baseline by 2.2 times on the channels of an undocumented vehicle CAN. On latent

variables, the lead grows from 1.5 to 1.9 times as the masking rises from 20 to 75%. The graph CAUSALBRIDGE discovers matches the documented graph in naming accuracy, and one recipe fits any backbone. A new system is named in minutes, at 2.6 to 56.6 times fewer tokens per latent variable than the methods that describe sampled cases.

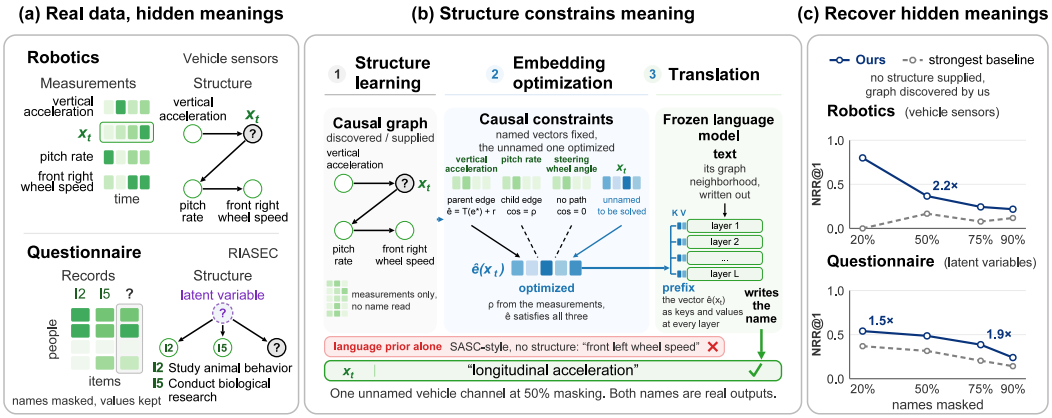

Figure 2: (a) Two systems from the evaluation at 50% masking: four channels of one vehicle CAN drive and three items of the RIASEC scale (Holland, 1997) with the latent variable that causes them, each with its structure. A question mark marks an unknown name, and x_t is the channel to be named. (b) The three steps of CAUSALBRIDGE on x_t: the structure is learned from the measurements; the vector of x_t is optimized under its structural relations to the named variables, whose vectors stay fixed; and a frozen language model takes this vector as a prefix and writes the name, shown below with the one the strongest baseline wrote without structure. (c) NRR@1, the share of unnamed variables whose written name matches the true name best among all names of the same system, for CAUSALBRIDGE and the strongest baseline at each masking level with no structure supplied.

2 PROBLEM FORMULATION

Given the measurements of a system and the names of a few of its variables, we aim to recover the semantics of the rest, including the latent variables that are never measured.

Key idea. The semantics of a variable can be recovered from its causal relations to the other variables. A causal graph contains no names, but once a few variables are named, the relations of an unnamed variable to them constrain its meaning.

Data-generating process. Consider a system with variables $V = O \cup L$, where O and L denote the observed and the latent variables, respectively. We assume that the variables are generated by a structural causal model (Pearl, 2009) with a directed acyclic graph G, in which an edge from u to v means that u is a direct cause of v. The measurements X contain n samples of the observed variables, while the latent variables are not measured. Each variable v has a name $\ell(v)$, but the names are known only for a subset V_{N} of the observed variables, which we call the named variables.

$$V_i = f_i\big(\mathrm{pa}(V_i), \varepsilon_i\big), \qquad X \in \mathbb{R}^{n \times |O|}, \qquad \ell : V \to \mathcal{T}, \qquad V_{\mathrm{N}} \subseteq O, \quad V_{\mathrm{U}} = (O \setminus V_{\mathrm{N}}) \cup L, \tag{1}$$

where $\mathrm{pa}(V_i)$ denotes the parents of V_i in G, ε_i is the noise term, $\mathcal{T}$ is the space of text, and V_{U} collects the unnamed variables. Variables with a common parent are siblings. In our evaluation, L is empty for a robot or a vehicle CAN, and contains the latent variables of a questionnaire.

Causal graphs and d-separation. Following Spirtes et al. (2000), the causal graph G characterizes the causal relations among the variables, and d-separation derives from G the conditional independence relations among them, associating dependence with connectedness in the graph and independence with separation. A path between two variables is any consecutive sequence of edges, disregarding their directions. A set S blocks a path if the path contains a chain $a \to m \to b$ or a fork $a \leftarrow m \to b$ whose middle variable m is in S, or a collider $a \to m \leftarrow b$ such that neither m nor any descendant of m is in S. S d-separates a and b if every path between them is blocked by S. The

collider case reflects that conditioning on a common effect makes its causes dependent. In Figure 3, x and y are d-separated by the empty set, since the only path between them passes the collider at t, while s and y are d-connected given $\{r, p\}$ because p is a descendant of t and is in the conditioning set. Under the Markov and faithfulness assumptions, S d-separates a and b if and only if they are independent given S, denoted by $a \perp\!\!\!\perp b \mid S$ (Definition 1). These relations are determined by the causal graph without semantic information, and we use them to find the semantics.

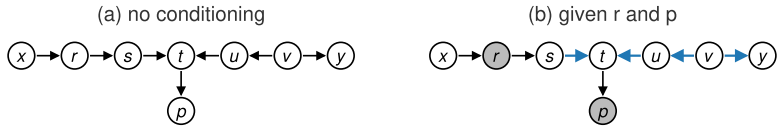

Figure 3: d-separation on a graph adapted from Pearl (2009, Section 11.1.2). A shaded variable is in the conditioning set S. (a) With S empty, x and y are d-separated by the collider at t. (b) Given $\{r, p\}$, s and y are d-connected, since p is a descendant of the collider t and is in S.

The naming task. We formulate the naming task as finding vectors that realize the causal structure and translating the vector of each unnamed variable into a name. To this end, we assign each variable v a vector e_v in the embedding space of a text encoder $\phi : \mathcal{T} \to \mathbb{R}^d$, so that each named variable keeps the vector $e_v^\star = \phi(\ell(v))$ of its name, which we call an anchor. In this space the cosine similarity between two vectors measures how related their meanings are, so we use it as a surrogate for the dependence between the corresponding variables. We say that an assignment $e : V \to \mathbb{R}^d$ realizes the structure if it satisfies three conditions. First, the vector of each variable is generated from the vectors of its parents. Second, the cosine similarities among the vectors reflect the dependence and independence relations that G implies, with the strength of each dependence estimated from the measurements X. Third, the vectors of the named variables stay fixed to their anchors. Let $\mathcal{A}(G, X)$ denote the feasible set of all assignments that realize the structure, which Section 3.2 states formally. The goal is to find an assignment $\hat{e} \in \mathcal{A}(G, X)$ and to translate $\hat{e}_v$ into a name $\hat{\ell}(v)$ that agrees with the true name $\ell(v)$ for each unnamed variable $v \in V_{\mathrm{U}}$.

The first two conditions constrain only the relations among the vectors. Edges determine which vectors should be similar, and d-separations determine which should remain distinct. The anchors remove part of this freedom, and the choice that remains within the feasible set is determined by existing knowledge, which enters as a regularizer toward English phrases (Section 3.2) and as the language model that translates the vector (Section 3.3).

3 METHOD

Based on the formulation above, we propose the framework CAUSALBRIDGE, which names the unnamed variables of a system in three steps (Figure 2(b) and Figure 17). First, we learn the causal structure $\hat{G}$ from the measurements alone so that its relations are free of the biases of human knowledge. Second, we optimize the vectors of the unnamed variables under this structure, with the named variables as anchors. Third, to write the meaning in words without losing the relations that determined it, a frozen language model takes the vector of each unnamed variable directly as input and translates it into a name. The steps are modular, so a supplied causal graph can substitute for the discovered one at the second step.

Running example. This section follows the vehicle CAN drive of the introduction at 50% masking (Figure 2(a)). Its named channels include the vertical acceleration, the pitch rate and the steering wheel angle, and x_t is an unnamed channel whose true name is longitudinal acceleration.

3.1 CAUSAL STRUCTURE LEARNING

The causal structure G describes how the variables of a system cause one another, and it is essential for our task, since it represents the true mechanism underlying the measurements and is therefore agnostic to the biases of existing knowledge. On a vehicle the vertical acceleration and the pitch rate both arise when the car brakes or accelerates, so an unnamed channel between them on a causal path measures the acceleration along the road, whatever a language prior expects it to be.

To learn the causal structure with guarantees, we apply causal discovery algorithms, which under their assumptions identify it from the measurements without any semantic information (Appendix C). Specifically, when all variables are observed, as on a robot or a vehicle CAN, we use best order score search (BOSS) (Andrews et al., 2023), a score-based search over the orderings of the variables. Since these measurements form a time series, we search over two consecutive time steps and keep the edges that agree with the temporal order. When latent variables exist, as on a questionnaire, we use rank-based latent causal discovery (RLCD) (Dong et al., 2024), given its flexibility to handle general structures that involve both latent and observed variables. Intuitively, respondents who rate "Study animal behavior" highly on the RIASEC questionnaire of Figure 2(a) tend to rate "Conduct biological research" highly as well, because one interest causes both. RLCD detects such a latent variable from the rank of the covariance of the measurements. Both algorithms are theoretically supported and empirically verified on various setups (Appendices C and E), and the output is the causal structure $\hat{G}$ over the observed and the latent variables, with no semantics attached.

Takeaway. In the running example, the discovered graph places x_t on a causal path from the vertical acceleration to the pitch rate and separates it from the steering wheel angle. Because these relations are learned from the measurements without reading any name, they already restrict what x_t can measure. None of the baselines learns a structure, so none of them can use such relations.

3.2 Constrained Embedding Optimization

We use the relations that the causal graph implies to constrain where the vector of each unnamed variable can lie in the embedding space, given the vectors of the named variables. In the running example, x_t is generated from the vertical acceleration, generates the pitch rate and is d-separated from the steering wheel angle. Specifically, let $\rho_{ab\cdot S}$ denote the strength of the dependence between a and b given S, estimated from the measurements, and let $e_{v\cdot S}$ denote the vector of v after removing the part explained by S. When S is empty, $e_{v\cdot S}$ is simply e_v, and when S consists of the parents of a and b, $e_{v\cdot S}$ is the residual $r_v = e_v - g_v(e_{\mathrm{pa}(v)})$. Here g_v is the embedding-space counterpart of the causal mechanism f_v in Eq. 1, generating the vector of v from those of its parents, and r_v captures what the parents do not explain, as the noise term ε_v does in the measurements. An assignment e realizes the three conditions if it satisfies

$$\begin{aligned} &e_v = g_v\big(e_{\mathrm{pa}(v)}\big) + r_v && \text{generation} \qquad && \cos(e_{a\cdot S}, e_{b\cdot S}) = 0 \text{ if } a \perp\!\!\!\perp b \mid S && \text{independence} \\ &\cos(e_{a\cdot S}, e_{b\cdot S}) = \psi(\rho_{ab\cdot S}) \text{ if } a \not\perp\!\!\!\perp b \mid S && \text{dependence} \qquad && e_v = e_v^\star \;\; \forall v \in V_{\mathrm{N}} && \text{anchoring} \end{aligned} \tag{2}$$

where ψ denotes a monotone function that maps a dependence to a cosine (Appendix C). We turn these conditions into an optimization problem by fixing each named variable at the vector $e_v^\star$ of its name and optimizing a vector $e_v \in \mathbb{R}^d$ for each unnamed variable, observed or latent. The objective:

$$\begin{aligned} \mathcal{L}(e) = &\underbrace{\sum\nolimits_{v:\,\mathrm{pa}(v)\neq\emptyset} \|r_v\|^2}_{\text{generation}} + \underbrace{\tfrac{\lambda_{\mathrm{ind}}}{|\mathcal{P}_{\mathrm{ind}}|} \sum\nolimits_{\mathcal{P}_{\mathrm{ind}}} (\cos(e_a, e_b) - \rho_{ab})^2}_{\text{marginal independence}} \\ &+ \underbrace{\tfrac{\lambda_{\mathrm{dep}}}{|\mathcal{P}_{\mathrm{dep}}|} \sum\nolimits_{\mathcal{P}_{\mathrm{dep}}} [\kappa|\rho_{ab}| - |\cos(e_a, e_b)|]_+^2}_{\text{marginal dependence}} + \underbrace{\tfrac{\lambda_{\mathrm{con}}}{|\mathcal{P}_{\mathrm{con}}|} \sum\nolimits_{\mathcal{P}_{\mathrm{con}}} (\cos(r_a, r_b) - \rho_{ab\cdot \mathrm{pa}})^2}_{\text{conditional dependence}} + \mathcal{R}(e), \end{aligned} \tag{3}$$

where $\mathcal{P}_{\mathrm{ind}}$ contains the pairs that $\hat{G}$ d-separates by the empty set, $\mathcal{P}_{\mathrm{dep}}$ the strongly dependent pairs, $\mathcal{P}_{\mathrm{con}}$ the pairs whose variables both have parents, ρ_{ab} denotes the distance correlation between a and b (Székely et al., 2007), and $[x]_+ = \max(x, 0)$. The weights λ scale the relation terms and $\kappa < 1$ is a fixed fraction. The four terms encode a common principle: the semantics of a variable should reflect those of its causes, causally related variables should be semantically related and unrelated ones distinct, and the semantics not explained by its causes should be specific to the variable itself. The generation term instantiates g_v as $g_v(e_{\mathrm{pa}(v)}) = \sum_{u\in\mathrm{pa}(v)} T(e_u; \rho_{uv})$, where T is a learned operator shared across all edges. T scales the parent's vector by $|\rho_{uv}|$ after a learned correction. On a negatively correlated edge, it applies a learned semantic negation map first, so the parent contributes its opposite meaning (Appendix F). The marginal dependence term imposes only a lower bound, which leaves the vectors the freedom to fit the other constraints. The conditional dependence term matches $\cos(r_a, r_b)$ to the dependence that remains between a and b given their parents. Because the constraints admit multiple solutions, the regularizer $\mathcal{R}$ favors unit-norm vectors

close to English phrases and to the named variables they depend on most. On questionnaires, the generation constraints also let latent variables carry semantics from named to unnamed items.

The operator T, the weight network that predicts per-variable weights for the terms of Eq. 3, and the encoder fine-tuning are fitted on training scales and on generated arms and vehicles (Appendix F). The operator and the weight network are trained with names masked as at evaluation, minimizing the cosine distance between each optimized vector and the embedding of its true name. For a new system, we jointly optimize the vectors of the unnamed variables by gradient descent.

Takeaway. In the running example, Eq. 2 turns the three relations of x_t into three constraints on its vector. The vector must be generated from the anchor of the vertical acceleration, generate the anchor of the pitch rate, and be orthogonal to the anchor of the steering wheel angle. The more relations tie an unnamed variable to the named ones, the more tightly its vector is constrained.

3.3 Translation via Prefix Adaptation

We translate $\hat{e}_v$ into a name by passing it to a frozen language model as a prefix (Li & Liang, 2021), so the model, which keeps its knowledge of words, attends to the whole vector and thus to every relation while it writes the name. Decoding $\hat{e}_v$ into words first would replace it by its few nearest phrases and drop its relations to every named variable, which the graph determined.

To build the prefix, we concatenate $\hat{e}_v$ with the vectors of the latent parent of v, when it has one, and a mean of the vectors of the named variables most correlated with v, into z_v. A learned adapter q maps z_v to the prefix $P_v = q(z_v) \in \mathbb{R}^{m \times d_p}$, where m is the prefix length and d_p the dimension of a prefix vector. Following Liu et al. (2022), learned matrices project the prefix into the key-value (KV) cache of every layer j as $\mathbf{K}_j = P_v W_j^K$ and $\mathbf{V}_j = P_v W_j^V$. The prompt text t_v carries the names around v. It lists the directed edges incident to v, and an evidence line gives the named variables most correlated with v and the phrases nearest to $\hat{e}_v$.

We keep the language model frozen except for a low-rank update (Hu et al., 2022) and minimize

$$\mathcal{L}_{\text{adapter}} = -\log p_\theta\big(\ell(v) \mid P_v, t_v\big) + \lambda_{\text{sim}}\big(1 - \cos(\bar{\phi}_{\text{LM}}(\hat{y}),\, \bar{\phi}_{\text{LM}}(\ell(v)))\big), \tag{4}$$

where θ collects the adapter and the low-rank update and $\hat{y}$ is the predicted name. The first term supervises the words of the name, and the second its meaning under the mean token embeddings $\bar{\phi}_{\text{LM}}$. Two further terms separate the name from those of the siblings of v (Appendix G). The adapter is trained on synthetic data only, and only the shape of the KV cache depends on the language model.

Takeaway. In the running example, the phrases nearest to the optimized vector of x_t are heading angle and lateral acceleration. Both describe the motion of the car body, yet neither names x_t. Decoding the vector into words discards the relations that distinguish x_t from the other channels. Reading the vector directly as a prefix, the frozen model writes longitudinal acceleration. The meaning of a name comes from the structure, and the language model supplies only the words.

Generality of the framework. The graph and the vectors are computed before the language model is called. Any model that takes vectors as input, such as a world model or a policy, can therefore receive the causal structure of its system in this form. The framework applies to any system with measurements and a few named variables, under the conditions of causal discovery (Appendix C).

4 Experiments

RQ1 asks whether causal structure recovers the semantics of the variables a language prior cannot, and whether CausalBridge can discover that structure itself. RQ2 to RQ4 ask where the gain comes from, whether one recipe carries across backbones and what naming a new system costs.

Data. We evaluate on two kinds of data that differ in how much text is available for naming, and their contrast tests the claim that causal structure supplies what a language prior does not cover. Robot and vehicle channels present the semantic gap in practice, since channel definitions vary across platforms and vehicle bus signals are often undocumented. Their names are short labels, which provide a language prior with little semantic information. We use six simulated arms (Zhu et al., 2020; Mandlekar et al., 2022), three logs of real arms

(O'Neill et al., 2024; Zhao et al., 2023) and ten drives of one car recorded from its CAN bus. The three scenarios differ in how much of the platform is documented, from a simulator with the true graph of its physics to a vehicle CAN with no documentation from the maker of the car. The graph derived from the physics or the documentation of a platform serves as the supplied graph. Questionnaire items are statements that respondents rate, so the text of the named items indicates what an unnamed neighbor measures. A latent variable causes the answers to several items, so the questionnaires also test the claim on variables that no one measured. We use five published scales, DASS (Lovibond & Lovibond, 1995), HEXACO (Ashton et al., 2007), KIMS (Baer et al., 2004), RIASEC and the World Values Survey (Welzel, 2013; Haerpfer et al., 2024). The supplied graph is the scoring key of a scale, which names every latent variable and lists its items. Every variable has a true name, so we mask 20 to 90% of the names and score the name a method writes against the masked names. Besides generated arms and vehicles, the operator and the weights of the objective are fitted on sixteen further public scales and the adapter is trained on synthetic scales. No training set overlaps the evaluation data (Appendix D).

Protocol. Masking hides 20, 50, 75 or 90% of the names. Every method receives the same measurements, known names and unnamed variables. When no structure is supplied, CAUSALBRIDGE discovers the graph from the measurements and each baseline uses the task adaptation described in Appendix I. When a structure is supplied, every method that accepts a graph as input receives the same graph. The name a method writes is scored as text in a frozen public embedding space (Wang et al., 2022) against every name of the same arm, log, drive or scale, so a copied known name scores zero. NRR@1 is the rate at which the true name is ranked first among the candidates, and MRR is the mean reciprocal rank of the true name (Voorhees, 1999). Identification is the rate at which a judge selects the target from five candidates. Naming quality, reported for latent variables, requires the output to be a name and rejects a copied item text. Under the discovered graph a discovered latent variable is scored against the published latent variable whose key contains most of its items (Appendix D.2). Appendix H gives the judge, the pools and the auxiliary metrics.

Baselines. Every baseline receives the variable's data and the known names. CLIP-Dissect-e5 selects a concept from a fixed bank. Automated Interpretability (AutoInterp), Delphi and SASC-style prompt a language model to describe the variable. DeViSE-style (Frome et al., 2013) regresses the data into the embedding space of the names. When a structure is supplied, the base language model is given the graph as text and GraphMAE reconstructs the masked node from its neighbors. Naming a latent variable requires a grouping of items. With no structure supplied, every questionnaire baseline therefore takes its item groups from a shared factor analysis (Appendix I).

RQ1: Does causal structure recover the semantics of the variables a language prior cannot, and can CAUSALBRIDGE discover that structure from the measurements themselves?
CAUSALBRIDGE gives the most accurate names on every set of variables, including the latent variables that no one measured, and it needs no supplied structure to do so.

The gain of CAUSALBRIDGE *is largest on the robotics tasks, where a language prior provides the least information.* Without structure, a baseline relies on this prior. A language model carries broad knowledge of psychological scales, so the text of one item already indicates which latent variable it measures. A channel label provides no comparable information. On the three robot columns, the baselines therefore remain low, while our method derives the meaning of each channel from its relations to the named channels. The lead narrows at 90% masking, where few anchors remain.

CAUSALBRIDGE *also names the latent variables more accurately than every baseline at every masking level.* The baselines name a latent variable from its items, so their evidence shrinks as the items lose their names. Our method relies on the causal relations between a latent variable and its items, and these relations are estimated from the measurements and remain available under masking. On naming quality CAUSALBRIDGE also outperforms the baselines, since the name is written from the vector of the latent variable and its relations to every item.

CAUSALBRIDGE *gains more over the baselines when no structure is supplied, since it discovers the causal graph itself while the baselines cannot* (Appendix J.2). The latent variables show this pattern most clearly, since our method discovers each of them from the measurements together with the items it causes. In Figure 4 (bottom), CAUSALBRIDGE on its discovered graph closely tracks its run on the supplied graph and outperforms it on the vehicle CAN. On the latent variables, as on

the vehicle CAN, its lead is wider than under the supplied graph at every masking level. On every set of observed variables, our method on its own graph reaches an NRR@1 at least as high as any baseline given the supplied graph. A method that waits for a supplied structure can know only what someone has already written down, with the purpose and the bias of whoever wrote it. The graph CAUSALBRIDGE discovers reflects the mechanism that produced the measurements, so a system can come to know its world from its own data alone.

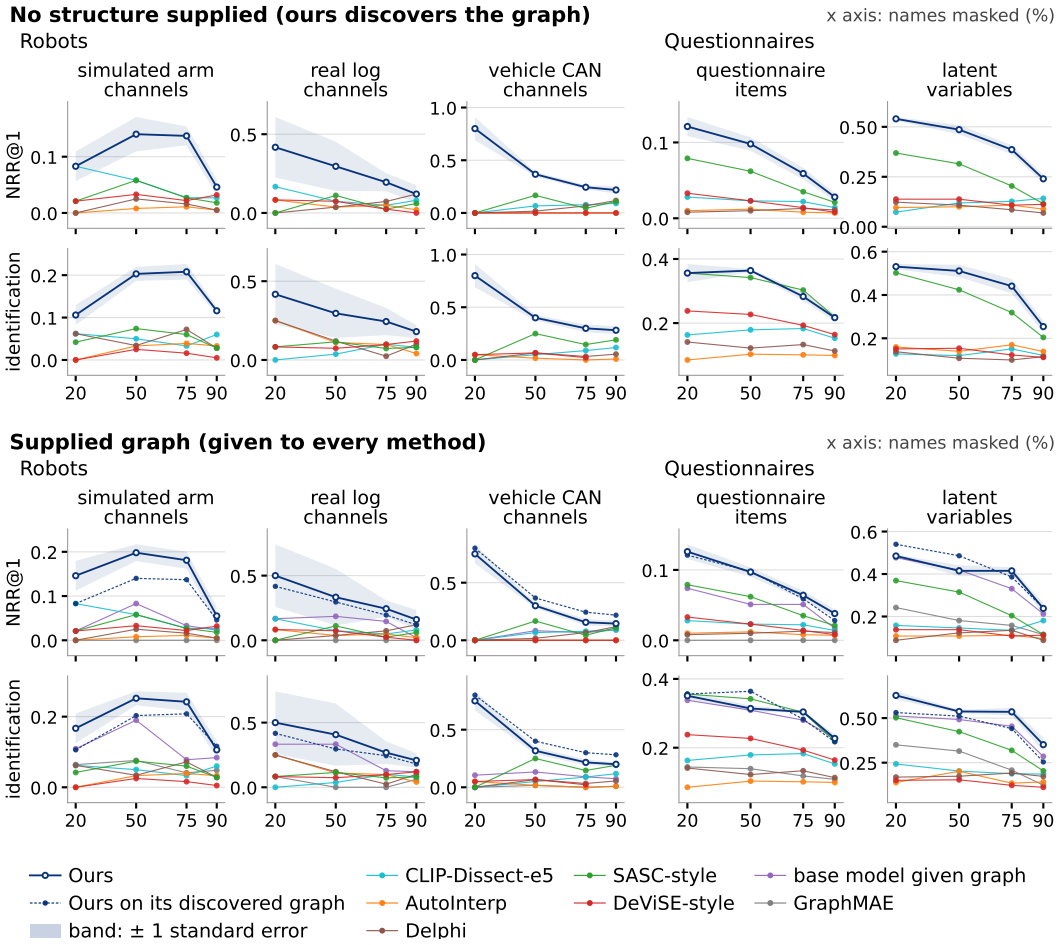

Figure 4: CAUSALBRIDGE against every baseline on the five sets of variables. Top, no structure is supplied, so our method runs on the graph it discovers. Bottom, the supplied graph is given to every method that accepts a graph as input. The thin dashed line with filled markers is our method on its discovered graph, whose latent variables differ from those of the supplied graph. Rows are NRR@1 and identification. Bands are one standard error over folds or over arms, logs and drives.

RQ2: Where does the gain come from?
CAUSALBRIDGE gains most from the prefix, which carries the structure into the language model, while the evidence line supplies the vocabulary of the name (Figure 19).

Removing the prefix causes the largest loss. The prefix passes the vector $\hat{e}_v$ to every attention layer through the KV cache, so the model conditions the name on the relations of the variable to all named variables. Without the prefix, the model receives $\hat{e}_v$ only as the phrases nearest to it, which preserve the meaning of $\hat{e}_v$ but lose these relations. The evidence line lists the named variables most correlated with the unnamed one in the measurements, so its value depends on how many names remain. On the simulated arms and the real logs, removing the evidence line costs more than removing the prefix through 50% masking, and the order reverses from 75% on. The two inputs also resolve different parts of a channel name, which consists of a type and a coordinate (Appendix H). The prefix determines the type on every robot column, and the evidence line resolves the coordinate on the vehicle CAN and the real logs. On latent variables, the prefix contributes most under the

discovered graph, since a discovered latent variable exists only as a vector, whereas a latent variable of a supplied key can enter the text as its list of items (Appendix J.4). The low-rank update adds little on its own. Its weights are shared across all variables and therefore cannot represent variable-specific structural relations, which enter the model through the prefix and the prompt.

RQ3: Does one recipe carry across backbones?
CAUSALBRIDGE generalizes across backbones with one adapter recipe, since the graph and the vectors are computed before the language model is called and only the adapter is trained per backbone.

The same recipe trains an adapter for the 3B model (Meta, 2024) and the 4B and 8B models (Qwen Team, 2025). The causal structure and the embedding do not depend on the backbone, and only the shape of the KV cache enters the adapter. Across the backbones, CAUSALBRIDGE yields the largest improvements on the channels, whose short labels provide the backbone alone with little semantic information. The improvements are smaller on latent variables and small on items, where the item text already provides strong semantic cues. At 90% masking the three backbones converge under both the supplied and the discovered graph. As the names become scarce, the structure determines performance, and the differences among the backbones shrink (Figure 19, Appendix J.5).

RQ4: What does naming a new system cost?
CAUSALBRIDGE uses fewer tokens than any sampling method and names a new system in minutes, since it writes each name from one prompt, whereas a sampling method describes many examples.

Our method spends 2.6 to 56.6 times fewer tokens on the same latent variable than the sampling methods and names the latent variables more accurately (Figure 20). A sampling method draws and describes examples of each variable, whereas our method writes each name from a single prompt. The three steps name ten drives from their raw traces in about seven minutes (Appendix K). The cost of a sampling method grows with the text that describes a variable, from the vehicle CAN to the questionnaires, whereas ours stays within a few hundred tokens on every column.

5 CONCLUSION AND DISCUSSION

We asked how causality bridges the semantic gap. Our answer is that, once a few variables are named, the meaning of another can be read from its causal relations, namely what causes it, what it causes and what it is independent of. These relations come from the mechanism behind the measurements and are recovered before any name is read, whereas association reaches a name through human knowledge and inherits its biases. The lead of CAUSALBRIDGE is largest where the least has been written down, and the discovered graph names variables as accurately as the documented one.

The main limitation of this work is its scope. We evaluated CAUSALBRIDGE only on systems whose true names are known but masked, so that the names it writes can be scored. We have not yet applied it to the settings that motivate it, such as the features of neural networks, latent variables in scientific data and recursively self-improving systems, which we discuss below.

The same semantic gap appears wherever a system produces measurements faster than its variables are named, and it includes every latent variable, which causal discovery returns without a name. Some of these systems were built by people who left them undocumented, such as the robots of a shared corpus (O'Neill et al., 2024), the CAN bus of a vehicle or a system under audit. A language prior knows only what such systems usually record and can assign one name to several channels, whereas their causal relations distinguish them. Other variables have no name yet, such as a feature that a sparse autoencoder recovers from a neural network (Bricken et al., 2023; Huben et al., 2024), the states behind a population of neurons or the modules of a gene regulatory network. Where no text describes such a variable, a language prior has nothing to read, but its causal relations remain in the measurements, and the framework turns the variable into a named hypothesis that can be tested.

A system that collects its own data meets the semantic gap again with each new variable, and a closed loop forms: the system measures, recovers the structure of what it measures and names the variables in that structure, and each recovered name can then serve as an anchor for the next. Repeating the loop expands the system's model of the world and its vocabulary from its own data. RSI (**R**ecursively **S**elf-**I**mproving) systems require this capability on their perceptual side. The loop is also a step from measuring the world to knowing it, and from signals to concepts that agree with the underlying causes. Causal relations are written in the data even when semantics are not.

AI USE STATEMENT

In this work, we used generative AI tools to generate a synthetic data set and to aid and polish the writing. A local open-weight language model (Qwen3-8B) wrote the latent variable names and the item texts of the synthetic questionnaire scales on which the adapter is trained (Appendix D), and a leakage filter removes every generated text that is close to a real one. A language model aided and polished the writing of this manuscript. We did not use generative AI tools to develop the conceptual framework or theoretical models, to formulate mathematical claims, to provide ingredients for proving them or to write proofs, to propose or refine the hypotheses, to design or provide feedback on the methodology or the experiments, to implement methods, or to interpret results, which are the authors' own. Translation, dataset cleaning and qualitative data analysis do not apply to this work. We have reviewed all AI-assisted work. Audit scripts recompute every number in the paper from the stored records, and the authors revised every AI-assisted passage. The language models that serve as a component of the method, as baselines and as the identification judge are described in Sections 3 and 4 and Appendices G to I. We take responsibility for the final content of this work, including text, claims and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work uses public questionnaire response data and public robot and vehicle telemetry. The questionnaire responses are anonymous and were released for research use. We do not attempt to re-identify any respondent, and we report results only at the level of items and latent variables. Naming variables in a system that a person did not document can make that system easier to audit, and it can also make an undocumented system easier to reverse engineer without the owner's consent. We consider the audit case the motivating one and note the other.

REPRODUCIBILITY STATEMENT

Appendix D gives the data, their preprocessing, the masking procedure and the training data of every trained part. Appendix H fixes the evaluation, including the scoring encoder, the candidate pools and the judge protocol. Appendix E gives the discovery algorithms and their tests, Appendix F the objective and the trained parts of the optimization, and Appendix G the adapter architecture and its training objective. Appendix K lists the settings of every stage. We will open-source the framework and the related data in the future. Every number in the tables and figures is computed from the stored records of the runs.

REFERENCES

Bryan Andrews, Joseph Ramsey, Ruben Sanchez Romero, Jazmin Camchong, and Erich Kummerfeld. Fast scalable and accurate discovery of DAGs using the best order score search and grow shrink trees. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, pp. 63945–63956, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/c9cde817d04811ba28e44071bd9f76a5-Abstract-Conference.html.

Michael C. Ashton, Kibeom Lee, and Lewis R. Goldberg. The IPIP–HEXACO scales: An alternative, public-domain measure of the personality constructs in the HEXACO model. *Personality and Individual Differences*, 42(8):1515–1526, 2007. doi: 10.1016/j.paid.2006.10.027.

Ruth A. Baer, Gregory T. Smith, and Kristin B. Allen. Assessment of mindfulness by self-report: The Kentucky Inventory of Mindfulness Skills. *Assessment*, 11(3):191–206, 2004. doi: 10.1177/1073191104268029.

David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pp. 3319–3327. IEEE Computer Society, 2017. doi: 10.1109/CVPR.2017.354.

Suneel Belkhale, Yuchen Cui, and Dorsa Sadigh. HYDRA: Hybrid robot actions for imitation learning. In *Proceedings of The 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pp. 2113–2133. PMLR, 2023. URL https://proceedings.mlr.press/v229/belkhale23a.html.

Steven Bills, Nick Cammarata, Dan Mossing, Henk Tillman, Leo Gao, Gabriel Goh, Ilya Sutskever, Jan Leike, Jeff Wu, and William Saunders. Language models can explain neurons in language models. https://openaipublic.blob.core.windows.net/neuron-explainer/paper/index.html, 2023.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Zac Hatfield-Dodds, Alex Tamkin, Karina Nguyen, Brayden McLean, Josiah E. Burke, Tristan Hume, Shan Carter, Tom Henighan, and Christopher Olah. Towards monosemanticity: Decomposing language models with dictionary learning. Transformer Circuits Thread, 2023. URL https://transformer-circuits.pub/2023/monosemantic-features/index.html.

Robert Brotherton, Christopher C. French, and Alan D. Pickering. Measuring belief in conspiracy theories: The Generic Conspiracist Beliefs Scale. *Frontiers in Psychology*, 4:279, 2013. doi: 10.3389/fpsyg.2013.00279.

Remi Cadene, Simon Aliberts, Francesco Capuano, Michel Aractingi, Adil Zouitine, Pepijn Kooijmans, Jade Choghari, Martino Russi, Caroline Pascal, Steven Palma, Mustafa Shukor, Jess Moss, Alexander Soare, Dana Aubakirova, Quentin Lhoest, Quentin Gallouédec, and Thomas Wolf. LeRobot: An open-source library for end-to-end robot learning. In *The Fourteenth International Conference on Learning Representations, ICLR 2026*, 2026. URL https://arxiv.org/abs/2602.22818.

Raymond B. Cattell, Herbert W. Eber, and Maurice M. Tatsuoka. *Handbook for the Sixteen Personality Factor Questionnaire (16 PF)*. Institute for Personality and Ability Testing, Champaign, IL, 1970.

Lawrence Yunliang Chen, Simeon Adebola, and Ken Goldberg. Berkeley UR5 demonstration dataset. https://sites.google.com/view/berkeley-ur5/home, 2023. Released with Open X-Embodiment.

David Maxwell Chickering. Optimal structure identification with greedy search. *Journal of Machine Learning Research*, 3:507–554, 2002. URL https://jmlr.org/papers/v3/chickering02b.html.

Richard Christie and Florence L. Geis. *Studies in Machiavellianism*. Academic Press, New York, 1970.

Nello Cristianini, John Shawe-Taylor, André Elisseeff, and Jaz S. Kandola. On kernel-target alignment. In Thomas G. Dietterich, Suzanna Becker, and Zoubin Ghahramani (eds.), *Advances in Neural Information Processing Systems 14 [Neural Information Processing Systems: Natural and Synthetic, NIPS 2001, December 3-8, 2001, Vancouver, British Columbia, Canada]*, pp. 367–374. MIT Press, 2001. URL https://proceedings.neurips.cc/paper/2001/hash/1f71e393b3809197ed66df836fe833e5-Abstract.html.

Justin Domke. Generic methods for optimization-based modeling. In Neil D. Lawrence and Mark A. Girolami (eds.), *Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2012, La Palma, Canary Islands, Spain, April 21-23, 2012*, volume 22 of *Proceedings of Machine Learning Research*, pp. 318–326. PMLR, 2012. URL http://proceedings.mlr.press/v22/domke12.html.

Xinshuai Dong, Biwei Huang, Ignavier Ng, Xiangchen Song, Yujia Zheng, Songyao Jin, Roberto Legaspi, Peter Spirtes, and Kun Zhang. A versatile causal discovery framework to allow causally-related hidden variables. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=FhQSGhBlqv.

Huaming Du, Tao Hu, Yijie Huang, Yu Zhao, Guisong Liu, Tao Gu, Gang Kou, and Carl Yang. Traceable latent variable discovery based on multi-agent collaboration. In Hakim Hacid, Yoelle Maarek, Francesco Bonchi, Ido Guy, and Emine Yilmaz (eds.), *Proceedings of the ACM Web Conference 2026, WWW 2026*, pp. 3732–3743. ACM, 2026. doi: 10.1145/3774904.3792244.

John Fox, Zhenghua Nie, and Jarrett Byrnes. sem: Structural equation models. R package version 3.1-16, `https://CRAN.R-project.org/package=sem`, 2024.

Jerome H. Friedman. Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29(5):1189–1232, 2001. doi: 10.1214/aos/1013203451.

Andrea Frome, Gregory S. Corrado, Jonathon Shlens, Samy Bengio, Jeffrey Dean, Marc'Aurelio Ranzato, and Tomás Mikolov. DeViSE: A deep visual-semantic embedding model. In Christopher J. C. Burges, Léon Bottou, Max Welling, Zoubin Ghahramani, and Kilian Q. Weinberger (eds.), *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pp. 2121–2129, 2013. URL `https://proceedings.neurips.cc/paper/2013/hash/7cce53cf90577442771720a370c3c723-Abstract.html`.

Gemma Team. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025. URL `https://arxiv.org/abs/2503.19786`.

Lewis R. Goldberg. The development of markers for the Big-Five factor structure. *Psychological Assessment*, 4(1):26–42, 1992. doi: 10.1037/1040-3590.4.1.26.

Lewis R. Goldberg, John A. Johnson, Herbert W. Eber, Robert Hogan, Michael C. Ashton, C. Robert Cloninger, and Harrison G. Gough. The international personality item pool and the future of public-domain personality measures. *Journal of Research in Personality*, 40(1):84–96, 2006. doi: 10.1016/j.jrp.2005.08.007.

Christian Haerpfer, Ronald Inglehart, Alejandro Moreno, Christian Welzel, Kseniya Kizilova, Jaime Diez-Medrano, Marta Lagos, Pippa Norris, Eduard Ponarin, and Bi Puranen. World Values Survey Wave 7 (2017–2022) Cross-National Data-Set, version 6.0.0. World Values Survey Association, 2024. URL `https://doi.org/10.14281/18241.24`.

Holly M. Hendin and Jonathan M. Cheek. Assessing hypersensitive narcissism: A reexamination of Murray's narcism scale. *Journal of Research in Personality*, 31(4):588–599, 1997. doi: 10.1006/jrpe.1997.2204.

Samsad Afrin Himi, Markus Bühner, Matthias Schwaighofer, Anna Klapetek, and Sven Hilbert. Multitasking behavior and its related constructs: Executive functions, working memory capacity, relational integration, and divided attention. *Cognition*, 189:275–298, 2019a. doi: 10.1016/j.cognition.2019.04.010.

Samsad Afrin Himi, Markus Bühner, Matthias Schwaighofer, Anna Klapetek, and Sven Hilbert. Multitasking behaviour and its related constructs: Executive functions, working memory capacity, relational integration, and divided attention. OSF project archive, `https://osf.io/tn6hp/`, 2019b.

John L. Holland. *Making Vocational Choices: A Theory of Vocational Personalities and Work Environments*. Psychological Assessment Resources, 3rd edition, 1997.

Karl J. Holzinger and Frances Swineford. A study in factor analysis: The stability of a bi-factor solution. Supplementary Educational Monographs 48, University of Chicago, 1939.

John L. Horn. A rationale and test for the number of factors in factor analysis. *Psychometrika*, 30 (2):179–185, 1965. doi: 10.1007/BF02289447.

Zhenyu Hou, Xiao Liu, Yukuo Cen, Yuxiao Dong, Hongxia Yang, Chunjie Wang, and Jie Tang. GraphMAE: Self-supervised masked graph autoencoders. In Aidong Zhang and Huzefa Rangwala (eds.), *KDD '22: The 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, August 14 - 18, 2022*, pp. 594–604. ACM, 2022. doi: 10.1145/3534678.3539321.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Robert Huben, Hoagy Cunningham, Logan Smith, Aidan Ewart, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. In *The Twelfth International Conference on Learning Representations, ICLR 2024*, 2024. URL https://openreview.net/forum?id=F76bwRSLeK.

I. T. Jolliffe. *Principal Component Analysis*. Springer Series in Statistics. Springer, New York, 2nd edition, 2002. doi: 10.1007/b98835.

Peter K. Jonason and Gregory D. Webster. The dirty dozen: A concise measure of the dark triad. *Psychological Assessment*, 22(2):420–432, 2010. doi: 10.1037/a0019265.

Daniel N. Jones and Delroy L. Paulhus. Introducing the Short Dark Triad (SD3): A brief measure of dark personality traits. *Assessment*, 21(1):28–41, 2014. doi: 10.1177/1073191113514105.

Henry F. Kaiser. The varimax criterion for analytic rotation in factor analysis. *Psychometrika*, 23 (3):187–200, 1958. doi: 10.1007/BF02289233.

Seth C. Kalichman and David Rompa. Sexual sensation seeking and sexual compulsivity scales: Validity, and predicting HIV risk behavior. *Journal of Personality Assessment*, 65(3):586–601, 1995. doi: 10.1207/s15327752jpa6503_16.

Oussama Khatib. A unified approach for motion and force control of robot manipulators: The operational space formulation. *IEEE Journal on Robotics and Automation*, 3(1):43–53, 1987. doi: 10.1109/JRA.1987.1087068.

Ilyes Khemakhem, Diederik P. Kingma, Ricardo Pio Monti, and Aapo Hyvärinen. Variational autoencoders and nonlinear ICA: A unifying framework. In Silvia Chiappa and Roberto Calandra (eds.), *The 23rd International Conference on Artificial Intelligence and Statistics, AISTATS 2020, 26-28 August 2020, Online [Palermo, Sicily, Italy]*, volume 108 of *Proceedings of Machine Learning Research*, pp. 2207–2217. PMLR, 2020. URL http://proceedings.mlr.press/v108/khemakhem20a.html.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6980.

Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. URL https://openreview.net/forum?id=SJU4ayYgl.

Erich Kummerfeld and Joseph D. Ramsey. Causal clustering for 1-factor measurement models. In Balaji Krishnapuram, Mohak Shah, Alexander J. Smola, Charu C. Aggarwal, Dou Shen, and Rajeev Rastogi (eds.), *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*, pp. 1655–1664. ACM, 2016. doi: 10.1145/2939672.2939838.

Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pp. 4582–4597. Association for Computational Linguistics, 2021. doi: 10.18653/v1/2021.acl-long.353.

Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. P-Tuning: Prompt tuning can be comparable to fine-tuning across scales and tasks. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Proceedings of the 60th Annual Meeting of the*

Association for Computational Linguistics (Volume 2: Short Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022, pp. 61–68. Association for Computational Linguistics, 2022. doi: 10.18653/v1/2022.acl-short.8.

Urbano Lorenzo-Seva and Jos M. F. ten Berge. Tucker's congruence coefficient as a meaningful index of factor similarity. *Methodology*, 2(2):57–64, 2006. doi: 10.1027/1614-2241.2.2.57.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

P.F. Lovibond and S.H. Lovibond. The structure of negative emotional states: Comparison of the Depression Anxiety Stress Scales (DASS) with the Beck Depression and Anxiety Inventories. *Behaviour Research and Therapy*, 33(3):335–343, 1995. doi: 10.1016/0005-7967(94)00075-U.

Ajay Mandlekar, Danfei Xu, Josiah Wong, Soroush Nasiriany, Chen Wang, Rohun Kulkarni, Li Fei-Fei, Silvio Savarese, Yuke Zhu, and Roberto Martín-Martín. What matters in learning from offline human demonstrations for robot manipulation. In *Proceedings of the 5th Conference on Robot Learning*, volume 164 of *Proceedings of Machine Learning Research*, pp. 1678–1690. PMLR, 2022. URL https://proceedings.mlr.press/v164/mandlekar22a.html.

Rod A. Martin, Patricia Puhlik-Doris, Gwen Larsen, Jeanette Gray, and Kelly Weir. Individual differences in uses of humor and their relation to psychological well-being: Development of the Humor Styles Questionnaire. *Journal of Research in Personality*, 37(1):48–75, 2003. doi: 10.1016/S0092-6566(02)00534-2.

Tatsuya Matsushima, Hiroki Furuta, Yusuke Iwasawa, and Yutaka Matsuo. Weblab xArm dataset, 2023. URL https://huggingface.co/datasets/lerobot/utokyo_xarm_pick_and_place.

Meta. Llama-3.2-3B-Instruct model card. https://huggingface.co/meta-llama/Llama-3.2-3B-Instruct, 2024.

George A. Miller. WordNet: A lexical database for English. *Communications of the ACM*, 38(11): 39–41, 1995. doi: 10.1145/219717.219748.

Ron Mokady, Amir Hertz, and Amit H. Bermano. ClipCap: CLIP prefix for image captioning. *arXiv preprint arXiv:2111.09734*, 2021. URL https://arxiv.org/abs/2111.09734.

Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer, New York, 2nd edition, 2006. doi: 10.1007/978-0-387-40065-5.

Tuomas P. Oikarinen and Tsui-Wei Weng. CLIP-Dissect: Automatic description of neuron representations in deep vision networks. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/forum?id=iPWiwWHc1V.

Abby O'Neill, Abdul Rehman, Abhiram Maddukuri, et al. Open X-Embodiment: Robotic learning datasets and RT-X models. In *IEEE International Conference on Robotics and Automation, ICRA 2024, Yokohama, Japan, May 13-17, 2024*, pp. 6892–6903. IEEE, 2024. doi: 10.1109/ICRA57147.2024.10611477.

OSP. Answers to the Consideration of Future Consequences Scale. Open-Source Psychometrics Project raw data, https://openpsychometrics.org/_rawdata/CFCS.zip, 2012a.

OSP. Answers to the Humor Styles Questionnaire. Open-Source Psychometrics Project raw data, https://openpsychometrics.org/_rawdata/HSQ.zip, 2012b.

OSP. Answers to the Kentucky Inventory of Mindfulness Skills. Open-Source Psychometrics Project raw data, https://openpsychometrics.org/_rawdata/KIMS.zip, 2012c.

OSP. Answers to the Sexual Compulsivity Scale. Open-Source Psychometrics Project raw data, https://openpsychometrics.org/_rawdata/SCS.zip, 2012d.

OSP. Answers to the Taylor Manifest Anxiety Scale. Open-Source Psychometrics Project raw data, https://openpsychometrics.org/_rawdata/TMA.zip, 2012e.

OSP. Answers to the Hypersensitive Narcissism Scale and the Dirty Dozen. Open-Source Psychometrics Project raw data, https://openpsychometrics.org/_rawdata/HSNS+DD.zip, 2013a.

OSP. Answers to the Woodworth Psychoneurotic Inventory. Open-Source Psychometrics Project raw data, https://openpsychometrics.org/_rawdata/WPI.zip, 2013b.

OSP. Answers to Cattell's 16 Personality Factors Test with items from the IPIP. Open-Source Psychometrics Project raw data, https://openpsychometrics.org/_rawdata/16PF.zip, 2014a.

OSP. Answers to the IPIP HEXACO equivalent scales. Open-Source Psychometrics Project raw data, https://openpsychometrics.org/_rawdata/HEXACO.zip, 2014b.

OSP. Answers to the Rosenberg Self-Esteem Scale. Open-Source Psychometrics Project raw data, https://openpsychometrics.org/_rawdata/RSE.zip, 2014c.

OSP. Answers to the Short Dark Triad. Open-Source Psychometrics Project raw data, https://openpsychometrics.org/_rawdata/SD3.zip, 2014d.

OSP. Development of the Nerdy Personality Attributes Scale. Open-Source Psychometrics Project, https://openpsychometrics.org/tests/NPAS/development/, 2015.

OSP. Answers to the Generic Conspiracist Beliefs Scale. Open-Source Psychometrics Project raw data, https://openpsychometrics.org/_rawdata/GCBS.zip, 2016.

OSP. Answers to the IPIP Big Five factor markers. Open-Source Psychometrics Project raw data, https://openpsychometrics.org/_rawdata/IPIP-FFM-data-8Nov2018.zip, 2018a.

OSP. Answers to the Depression Anxiety Stress Scales. Open-Source Psychometrics Project raw data, https://openpsychometrics.org/_rawdata/DASS_data_21.02.19.zip, 2018b.

OSP. Answers to the Nerdy Personality Attributes Scale. Open-Source Psychometrics Project raw data, https://openpsychometrics.org/_rawdata/NPAS-data-16December2018.zip, 2018c.

OSP. Answers to the Holland Code (RIASEC) test. Open-Source Psychometrics Project raw data, https://openpsychometrics.org/_rawdata/RIASEC_data12Dec2018.zip, 2018d.

OSP. Answers to the Machiavellianism Test, a version of the MACH-IV. Open-Source Psychometrics Project raw data, https://openpsychometrics.org/_rawdata/MACH_data.zip, 2019.

OSP. Raw data from online personality tests. https://openpsychometrics.org/_rawdata/, 2026. Accessed September 2026.

Gonçalo Santos Paulo, Alex Troy Mallen, Caden Juang, and Nora Belrose. Automatically interpreting millions of features in large language models. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*, pp. 48393–48421. PMLR, 2025. URL https://proceedings.mlr.press/v267/paulo25a.html.

Judea Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, Cambridge, 2nd edition, 2009. doi: 10.1017/CBO9780511803161.

Jonas Peters, Joris M. Mooij, Dominik Janzing, and Bernhard Schölkopf. Causal discovery with continuous additive noise models. *Journal of Machine Learning Research*, 15(58):2009–2053, 2014.

Russell A. Poldrack, Aniket Kittur, Donald J. Kalar, Eric Miller, Christian Seppa, Yolanda Gil, Douglas Stott Parker, Fred W. Sabb, and Robert M. Bilder. The Cognitive Atlas: Toward a knowledge foundation for cognitive neuroscience. *Frontiers in Neuroinformatics*, 5:17, 2011. doi: 10.3389/fninf.2011.00017.

Qwen Team. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. URL https://arxiv.org/abs/2505.09388.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

Morris Rosenberg. *Society and the Adolescent Self-Image*. Princeton University Press, Princeton, NJ, 1965. doi: 10.1515/9781400876136.

Emanuele Rossi, Henry Kenlay, Maria I. Gorinova, Benjamin Paul Chamberlain, Xiaowen Dong, and Michael M. Bronstein. On the unreasonable effectiveness of feature propagation in learning on graphs with missing node features. In Bastian Rieck and Razvan Pascanu (eds.), *Learning on Graphs Conference, LoG 2022, 9-12 December 2022, Virtual Event*, volume 198 of *Proceedings of Machine Learning Research*, pp. 11:1–11:16. PMLR, 2022. URL https://proceedings.mlr.press/v198/rossi22a.html.

Harald Schafer, Eder Santana, Andrew Haden, and Riccardo Biasini. A commute in data: The comma2k19 dataset. *arXiv preprint arXiv:1812.05752*, 2018. URL https://arxiv.org/abs/1812.05752.

Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Toward causal representation learning. *Proceedings of the IEEE*, 109(5):612–634, 2021. doi: 10.1109/JPROC.2021.3058954.

Ricardo Silva, Richard Scheines, Clark Glymour, and Peter Spirtes. Learning the structure of linear latent variable models. *Journal of Machine Learning Research*, 7:191–246, 2006. URL https://jmlr.org/papers/v7/silva06a.html.

Chandan Singh, Aliyah R. Hsu, Richard J. Antonello, Shailee Jain, Alexander G. Huth, Bin Yu, and Jianfeng Gao. Explaining black box text modules in natural language with language models. *arXiv preprint arXiv:2305.09863*, 2023. URL https://arxiv.org/abs/2305.09863.

Robyn Speer, Joshua Chin, and Catherine Havasi. ConceptNet 5.5: An open multilingual graph of general knowledge. In Satinder Singh and Shaul Markovitch (eds.), *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, pp. 4444–4451. AAAI Press, 2017. doi: 10.1609/aaai.v31i1.11164.

Peter Spirtes, Clark Glymour, and Richard Scheines. *Causation, Prediction, and Search*. Adaptive computation and machine learning. MIT Press, 2nd edition, 2000.

Alan Strathman, Faith Gleicher, David S. Boninger, and C. Scott Edwards. The consideration of future consequences: Weighing immediate and distant outcomes of behavior. *Journal of Personality and Social Psychology*, 66(4):742–752, 1994. doi: 10.1037/0022-3514.66.4.742.

Seth Sullivant, Kelli Talaska, and Jan Draisma. Trek separation for Gaussian graphical models. *The Annals of Statistics*, 38(3):1665–1685, 2010. doi: 10.1214/09-AOS760.

Gábor J. Székely, Maria L. Rizzo, and Nail K. Bakirov. Measuring and testing dependence by correlation of distances. *The Annals of Statistics*, 35(6):2769–2794, 2007. doi: 10.1214/009053607000000505.

Janet A. Taylor. A personality scale of manifest anxiety. *The Journal of Abnormal and Social Psychology*, 48(2):285–290, 1953. doi: 10.1037/h0056264.

Andrey N. Tikhonov and Vasiliy Y. Arsenin. *Solutions of Ill-posed Problems*. V. H. Winston and Sons, Washington, DC, 1977.

Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2017, Vancouver, BC, Canada, September 24-28, 2017*, pp. 23–30. IEEE, 2017. doi: 10.1109/IROS.2017.8202133.

Emanuel Todorov, Tom Erez, and Yuval Tassa. MuJoCo: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2012*, pp. 5026–5033. IEEE, 2012. doi: 10.1109/IROS.2012.6386109.

Warren S. Torgerson. Multidimensional scaling: I. Theory and method. *Psychometrika*, 17(4): 401–419, 1952. doi: 10.1007/BF02288916.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *6th International Conference on Learning Representations, ICLR 2018*, 2018. URL https://openreview.net/forum?id=rJXMpikCZ.

Ellen M. Voorhees. The TREC-8 question answering track report. In Ellen M. Voorhees and Donna K. Harman (eds.), *Proceedings of The Eighth Text REtrieval Conference, TREC 1999, Gaithersburg, Maryland, USA*, volume 500-246 of *NIST Special Publication*. National Institute of Standards and Technology (NIST), 1999. URL http://trec.nist.gov/pubs/trec8/papers/qa_report.pdf.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*, 2022. URL https://arxiv.org/abs/2212.03533.

Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Lingpeng Kong, Qi Liu, Tianyu Liu, and Zhifang Sui. Large language models are not fair evaluators. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pp. 9440–9450. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.511.

Christian Welzel. *Freedom Rising: Human Empowerment and the Quest for Emancipation*. Cambridge University Press, 2013. doi: 10.1017/CBO9781139540919.

Joe Whittaker. *Graphical Models in Applied Multivariate Statistics*. Wiley, Chichester, 1990.

Robert S. Woodworth. Woodworth Psychoneurotic Inventory. PsycTESTS, American Psychological Association, 1918.

Feng Xie, Ruichu Cai, Biwei Huang, Clark Glymour, Zhifeng Hao, and Kun Zhang. Generalized independent noise condition for estimating latent variable causal graphs. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, pp. 14891–14902, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/aa475604668730af60a0a87cc92604da-Abstract.html.

Tony Z. Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. In Kostas E. Bekris, Kris Hauser, Sylvia L. Herbert, and Jingjin Yu (eds.), *Robotics: Science and Systems XIX, Daegu, Republic of Korea, July 10-14, 2023*, 2023. doi: 10.15607/RSS.2023.XIX.016.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-judge with MT-Bench and Chatbot Arena. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, pp. 46595–46623, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/91f18a1287b398d378ef22505bf41832-Abstract-Datasets_and_Benchmarks.html.

Yujia Zheng, Ignavier Ng, and Kun Zhang. On the identifiability of nonlinear ICA: Sparsity and beyond. In Sanmi Koyejo, Shakir Mohamed, Alekh Agarwal, Danielle Belgrave, Kyunghyun Cho, and Alice Oh (eds.), *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, pp. 16411–16422, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/6801fa3fd290229efc490ee0cf1c5687-Abstract-Conference.html.

Yujia Zheng, Biwei Huang, Wei Chen, Joseph Ramsey, Mingming Gong, Ruichu Cai, Shohei Shimizu, Peter Spirtes, and Kun Zhang. Causal-learn: Causal discovery in Python. *Journal of Machine Learning Research*, 25(60):1–8, 2024. URL http://jmlr.org/papers/v25/23-0970.html.

Yujia Zheng, Zhuokai Zhao, Zijian Li, Yaqi Xie, Mingze Gao, Lizhu Zhang, and Kun Zhang. Thought communication in multiagent collaboration. In Danielle Belgrave, Cheng Zhang, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, and Nancy Chen (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, pp. 123389–123418, 2025. doi: 10.52202/085713-4113. URL http://papers.nips.cc/paper_files/paper/2025/hash/b2b502c3629beadda06311386d2c6f73-Abstract-Conference.html.

Yuke Zhu, Josiah Wong, Ajay Mandlekar, Roberto Martín-Martín, Abhishek Joshi, Kevin Lin, Abhiram Maddukuri, Soroush Nasiriany, and Yifeng Zhu. robosuite: A modular simulation framework and benchmark for robot learning. *arXiv preprint arXiv:2009.12293*, 2020. URL https://arxiv.org/abs/2009.12293.

APPENDIX CONTENTS

- **A** Notation (p. 19)
- **B** Case studies (p. 20)
- **C** Theoretical background (p. 22)
- **D** Datasets and preprocessing (p. 25)
- **E** Causal structure learning (p. 31)
- **F** Constrained embedding optimization (p. 34)
- **G** Translation via prefix adaptation (p. 36)
- **H** Evaluation protocol (p. 38)
- **I** Baselines and fidelity to the published designs (p. 39)
- **J** Supplemental experimental results (p. 41)
- **K** Settings, compute and licenses (p. 57)
- **L** Recovered structures (p. 59)

A NOTATION

Table 1: Symbols of the main text and the appendix.

symbol	meaning
$V = O \cup L$	the variables, observed and latent
f_i, ε_i	the causal mechanism of V_i in Eq. 1, and its noise term
X	the measurement matrix over the observed variables
ℓ	the true naming map, from a variable to its text
$V_{\mathrm{N}}, V_{\mathrm{U}}$	the named variables, and the unnamed variables to be named
ϕ	the frozen text encoder
e_v	the vector assigned to variable v
$e_v^\star = \phi(\ell(v))$	the vector of the name of v, an anchor when v is named
$\hat{e}_v$	the optimized vector of an unnamed variable
G, $\hat{G}$	the causal graph, and the discovered graph
$a \perp\!\!\!\perp b \mid S$	a and b are independent given S
$(A \perp\!\!\!\perp B \mid S)_G$	S d-separates A from B in G
$\Sigma_{A,B}$	the covariance matrix between the variables of A and those of B
g_v	the generating function of the vector of v from the vectors of its parents
$\mathcal{A}(G, X)$	the feasible set of assignments that satisfy Eq. 2
r_v	the residual $e_v - g_v(e_{\mathrm{pa}(v)})$
T	the operator shared across edges
f_{neg}	the frozen semantic negation map
ρ_{ab}	the distance correlation of a and b, signed by their Pearson correlation
$\tilde{\rho}_{ab}$	ρ_{ab} set to zero below the significance threshold
$e_{v \cdot S}$	the vector of v given S, e_v for empty S and the residual r_v for the parents
$\rho_{ab \cdot S}$	the distance correlation given S, of the residuals of a and b on their parents when $S = \mathrm{pa}$
ψ	the monotone map from dependence to cosine similarity, the identity here
$\mathcal{P}_{\mathrm{ind}}$	pairs d-separated by the empty set, with no common ancestor and neither an ancestor of the other
$\mathcal{P}_{\mathrm{dep}}$	the trek-connected pairs of strongest dependence
$\mathcal{P}_{\mathrm{con}}$	pairs of variables that have parents
$\lambda_{\mathrm{ind}}, \lambda_{\mathrm{dep}}, \lambda_{\mathrm{con}}, \kappa$	the weights of the three relation terms and the fraction of the marginal dependence term
Π	the soft nearest-neighbor projection onto the concept bank
c_v	the neighbor mean of an unnamed observed variable
$\mathcal{R}$	the regularizer of Eq. 13
$b_{uv}, \Delta, \eta_{uv}$	the base map of the operator on an edge, its trained correction, and the type of the edge
ω, $\mathcal{S}$	the weight network on the terms of Eq. 3, and the sign-consistency penalty
z_v, s_v	the adapter input of a variable, and its neighborhood scalars
t_v	the prompt text around v
ϕ_{LM}, $\bar{\phi}_{\mathrm{LM}}$	the token embeddings of the language model, and their mean over a text
P_v, m, d_p	the prefix of a variable, its length and the dimension of a prefix vector
q, q_i, q_γ, γ	the adapter, its five networks, its gate network and the gates
$\hat{y}$, θ	the name the language model writes, and the parameters of the adapter and the low-rank update
n, d	the number of samples, and the dimension of the embedding space
$\mathrm{pa}(v)$	the parents of v in the graph
$\mathcal{T}$, $\hat{\ell}$	the space of text, and the name written for an unnamed variable
$\hat{\mathcal{A}}(G, X)$	the estimated feasible set of Definition 5
$\mathcal{L}$, $\mathcal{L}_0$	the objective of Eq. 3, and the objective without the regularizer

B CASE STUDIES

Figure 5 compares the names that three baselines and our method wrote for the same variables. The variables are three questionnaire latent variables and three unnamed channels of one drive. Table 2 then shows one latent variable of DASS in full. The four figures that follow show our own outputs together with the context the model received. They cover two questionnaires, one vehicle drive and one real robot log.

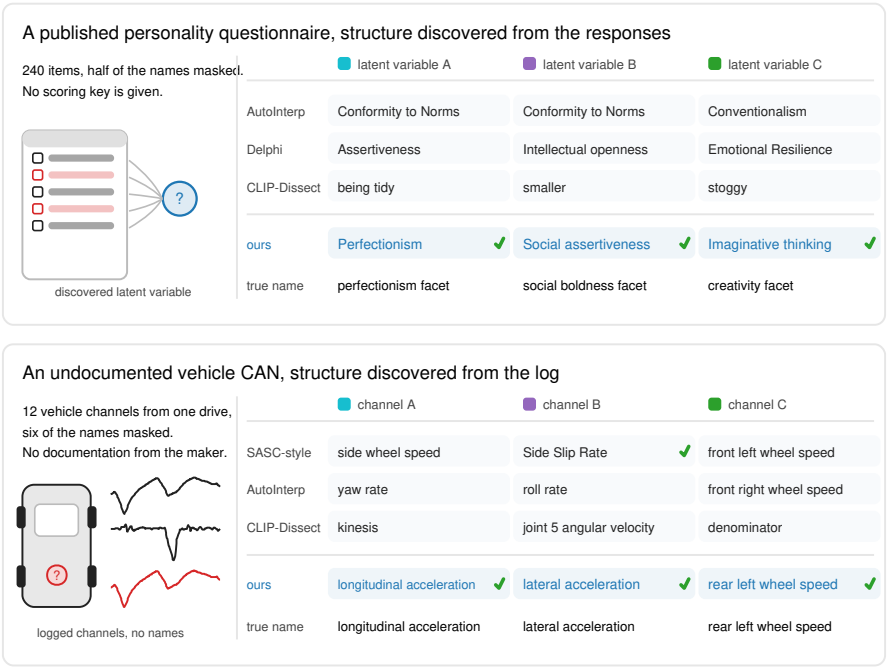

Figure 5: A questionnaire with three unnamed latent variables at 50% masking, and one drive with three of its unnamed channels at 50%. **Left**, the data. **Right**, what existing methods (above) and ours (below) wrote, with the true name underneath. A tick marks an answer the identification judge accepted.

Table 2 shows one latent variable at 50% masking. The first block holds one neighbor line per item the latent variable causes, with `(label hidden)` where the item is unnamed. The second block is the evidence line, which lists the named items that the latent variable causes. It also gives the correlations of the factor score of the latent variable with the factor scores of the other latent variables of the scale. The last two lines are the name CAUSALBRIDGE wrote and the true name.

Table 2: One latent variable end to end at 50% masking, printed with the context the model received. It causes 4 items, 2 of them named and 2 unnamed.

Causal neighbors given to the model

- EFFECT: it causes the item "I experienced breathing difficulty (eg, excessively rapid breathing, breathlessness in the absence of physical exertion)."
- EFFECT: it causes the item "(label hidden)"
- EFFECT: it causes the item "(label hidden)"
- EFFECT: it causes the item "I experienced trembling (eg, in the hands)."

Structural evidence given to the model

a factor with 4 items (2 hidden), visible items: "I experienced breathing difficulty (eg, excessively rapid breathing, breathlessness in the absence of physical exertion).", "I experienced trembling (eg, in the hands).". its factor score correlates +0.73 with the factor of "I found myself in situations that made me so anxious I was most relieved when they ended." / "I had a feeling of faintness.", +0.68 with the factor of "I found it difficult to relax." / "I found it hard to wind down.", +0.52 with the factor of "I felt down-hearted and blue." / "I was unable to become enthusiastic about anything.". cue phrases decoded from its data: feeling nervous, nervousness. in the constrained embedding its vector lies closest to "I experienced trembling (eg, in the hands)." (cos 0.94), "I felt that I was using a lot of nervous energy." (cos 0.92). its hidden items read from the data as: feeling nervous, trembling; feeling nervous, nervousness.

CAUSALBRIDGE writes Anxiety symptoms
True name the DASS clinical scale: anxiety

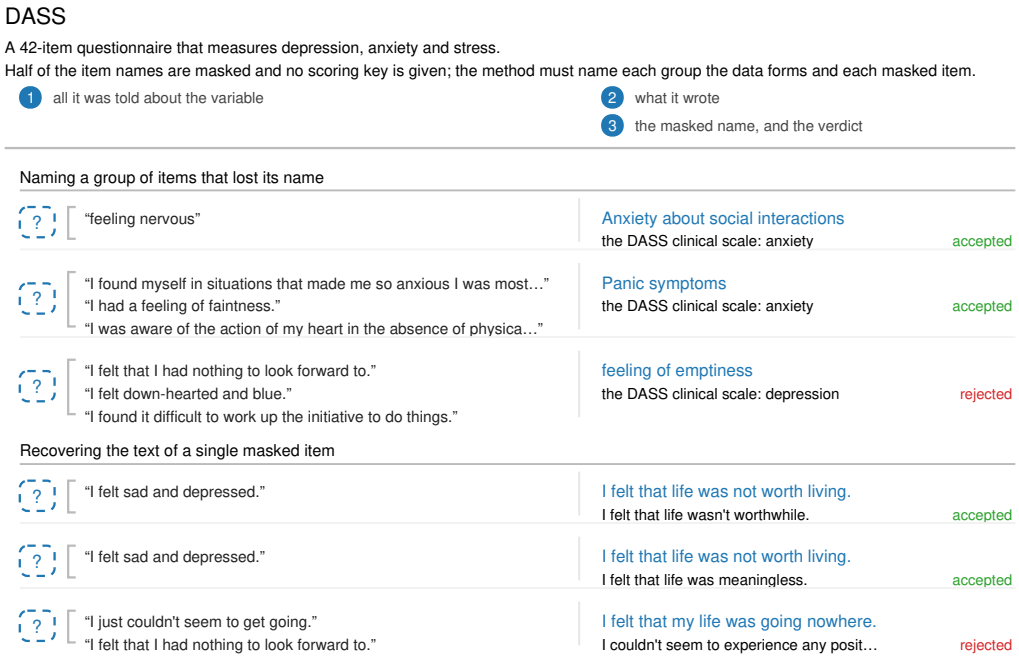

Figure 6: Cases from DASS at 50% masking. The upper block covers discovered latent variables and the lower block covers masked items. Each row shows what the model received, the name it wrote, and the reference name with the judge's verdict.

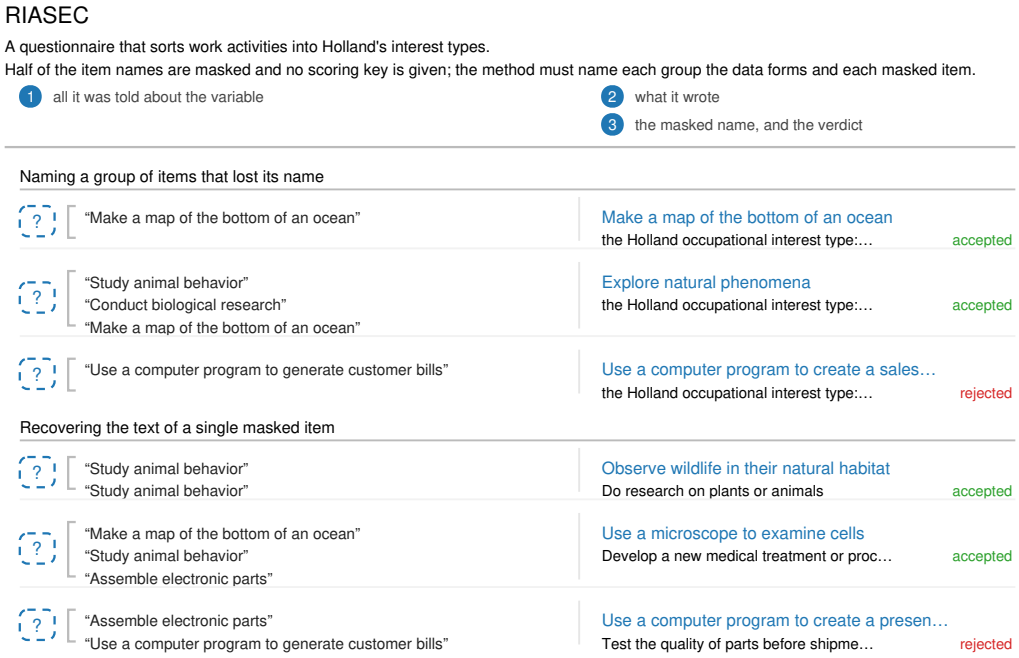

Figure 7: Cases from RIASEC at 50% masking, in the layout of Figure 6.

Figure 8 shows a failure mode that the aggregate numbers hide. On a straight drive, the speed channels have a pairwise Pearson correlation of $+1.00$, so nothing in the data separates them. Two of the three unnamed speed channels therefore receive the same name. One is then counted as an identification and two as misses.

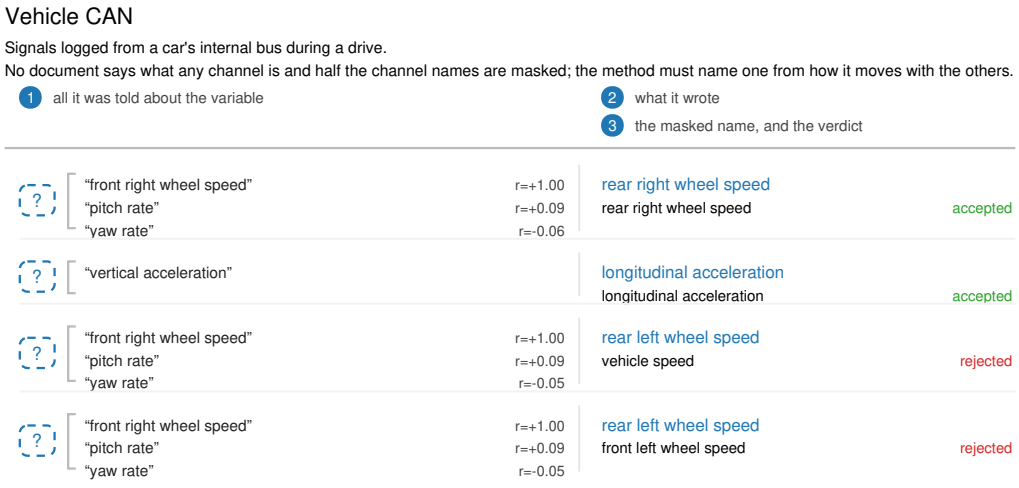

Figure 8: Cases from one vehicle drive at 50% masking. Each row shows the named neighbors of the unnamed channel in the graph, the name written, and the true name with the judge's verdict. For a channel without named neighbors, the row shows the channels it moves with.

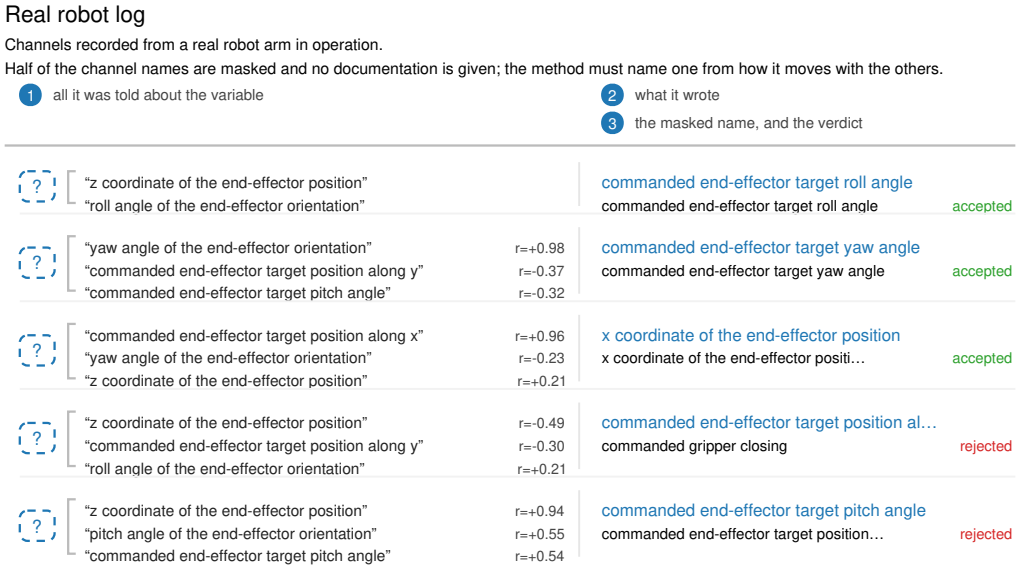

Figure 9: Cases from a real robot log at 50% masking, in the layout of Figure 8.

C Theoretical Background

Every cited statement below is restated from its source in the notation of Section 2, and its proof is in the cited work. Definition 5 and Proposition 3 are ours. Let $G = (V, E)$ be a directed acyclic graph and P a distribution over V. Let Σ be the covariance matrix of the variables and $\Sigma_{A,B}$ its submatrix with rows A and columns B. For disjoint $A, B, S \subseteq V$ we write $(A \perp\!\!\!\perp B \mid S)_G$ when S d-separates A from B in G, and $(A \perp\!\!\!\perp B \mid S)_P$ when A and B are independent given S in P.

What the graph implies about the distribution.

Definition 1 (Markov and faithfulness; Pearl, 2009; Spirtes et al., 2000)**.** *P is Markov relative to G if $(A \perp\!\!\!\perp B \mid S)_G \Rightarrow (A \perp\!\!\!\perp B \mid S)_P$ for all disjoint $A, B, S \subseteq V$, and faithful to G if the converse holds for all such triples. Under both,*

$$\{(A, B, S) : (A \perp\!\!\!\perp B \mid S)_P\} \;=\; \{(A, B, S) : (A \perp\!\!\!\perp B \mid S)_G\}. \tag{5}$$

Proposition 1 (Gaussian case; Sullivant et al., 2010, Prop. 2.2 and the remark after it)**.** *Let $P = \mathcal{N}(\mu, \Sigma)$ with Σ positive definite, and let A, B, S be disjoint. Then*

$$(A \perp\!\!\!\perp B \mid S)_P \iff \operatorname{rank} \Sigma_{A\cup S,\, B\cup S} = |S| \iff \operatorname{pcor}(a, b \mid S) = 0 \ \ \forall a \in A,\, b \in B, \tag{6}$$

where $\operatorname{pcor}(a, b \mid S)$ is the partial correlation of a and b given S, which is the Pearson correlation of their residuals after linear regression on S.

Interpretation. Under the Markov and faithfulness assumptions, the marginal terms of Eq. 3 fit the marginal independence and dependence statements that G implies. In the Gaussian case, a d-separation of G corresponds to a vanishing partial correlation. The objective measures dependence by the distance correlation ρ_{ab} (Székely et al., 2007), signed by the Pearson correlation. The distance correlation vanishes under independence without the Gaussian assumption. The conditional dependence term of Eq. 3 matches $\cos(r_a, r_b)$ to $\rho_{ab\cdot \mathrm{pa}}$. The marginal independence term matches $\cos(e_a, e_b)$ to ρ_{ab} on the pairs that are d-separated by the empty set in G. The marginal dependence term keeps $|\cos(e_a, e_b)|$ above $\kappa\,|\rho_{ab}|$ on the trek-connected pairs of strongest dependence. The target $\rho_{ab\cdot \mathrm{pa}}$ of the conditional dependence term is the dependence that remains between the residuals of a and b after regression on their own parents. When the noise terms enter additively and are mutually independent, each residual is the noise term of its variable. The residuals of a and b are then independent (Peters et al., 2014), and the target vanishes.

Rank and latent variables.

Definition 2 (Trek and t-separation; Sullivant et al., 2010, Defs. 2.6 and 2.7)**.** *A trek from u to v is a pair of directed paths (P_1, P_2) with sinks u and v and a common source, either path possibly a single vertex. For subsets A, B, C_A, C_B of V, not necessarily disjoint, (C_A, C_B) t-separates A from B if every trek from A to B contains a vertex of C_A on P_1 or a vertex of C_B on P_2.*

Theorem 1 (Trek separation; Sullivant et al., 2010, Thm. 2.8)**.** *For every covariance matrix of a linear structural equation model with graph G and mutually independent noise terms,*

$$\operatorname{rank} \Sigma_{A,B} \ \le\ \min\big\{|C_A| + |C_B| : (C_A, C_B) \text{ t-separates } A \text{ from } B\big\}, \tag{7}$$

with equality for generic parameters.

Interpretation. For generic parameters, a rank constraint $\operatorname{rank} \Sigma_{A,B} = k < \min(|A|, |B|)$ implies that a t-separating pair of total size k exists. In a measurement model, that pair is a set of k latent variables that separate A from B along every trek. The rank tests of Appendix E introduce latent variables on this evidence, at every cardinality up to a maximum.

Identifiability of the recovered latent structure.

Definition 3 (Latent linear causal model; Dong et al., 2024, Def. 1 and Asm. 1)**.** *$V = L \cup O$ with latent variables L and observed variables O, and*

$$V_i \ =\ \sum_{V_j \in \mathrm{pa}(V_i)} a_{ij} V_j + \varepsilon_i, \tag{8}$$

where a_{ij} is the causal coefficient from V_j to V_i and ε_i the noise term. The noise terms are mutually independent. P is rank faithful to G if Eq. 7 holds with equality for all $A, B \subseteq V$.

Definition 4 (Pure children and atomic cover; Dong et al., 2024, Defs. 4 and 5)**.** *The pure children of a set $\mathbf{V} \subseteq V$ are $\mathrm{PCh}(\mathbf{V}) = \{v : \mathrm{pa}(v) \neq \emptyset,\ \mathrm{pa}(v) \subseteq \mathbf{V}\}$. For a set $\mathcal{C}$ of sets let $\|\mathcal{C}\| = |\bigcup_{C\in\mathcal{C}} C|$. Let l of the variables of $\mathbf{V}$ be latent. $\mathbf{V}$ is an atomic cover if it is a single observed variable, or if the three conditions below hold. (i) There is a set $\mathcal{C}$ of atomic covers with $\bigcup_{C\in\mathcal{C}} C \subseteq \mathrm{PCh}(\mathbf{V})$ and $\|\mathcal{C}\| \ge l + 1$. (ii) There is a set $\mathcal{N}$ of covers with $(\bigcup_{N\in\mathcal{N}} N) \cap (\bigcup_{C\in\mathcal{C}} C) = \emptyset$, every element of $\bigcup_{N\in\mathcal{N}} N$ a neighbor of $\mathbf{V}$, $\mathcal{N}$ and $\mathcal{C}$ d-separated by $\mathbf{V}$, and $\|\mathcal{N}\| \ge l + 1$. (iii) $\mathbf{V}$ has no partition into two atomic covers.*

Condition 1 (Dong et al., 2024, Conds. 1 and 2)**.** *(a) Every latent variable belongs to at least one atomic cover, and for each atomic cover that contains a latent variable, none of its children is adjacent to any of its neighbors. (b) If a set C consists of colliders of two atomic covers $\mathbf{V}_1, \mathbf{V}_2$, A is the minimal set that d-separates $\mathbf{V}_1$ from $\mathbf{V}_2$, and a latent variable lies in $\mathbf{V}_1 \cup \mathbf{V}_2 \cup C \cup A$, then $|C| + |A| \ge |\mathbf{V}_1| + |\mathbf{V}_2|$.*

Theorem 2 (Identifiability of RLCD; Dong et al., 2024, Thm. 12)**.** *Let (G, P) be a latent linear causal model of Definition 3 with P rank faithful to G, and let G satisfy Condition 1. Then RLCD asymptotically identifies the Markov equivalence class of $\mathcal{O}_{\min}(\mathcal{O}_s(G))$. Here the skeleton operator $\mathcal{O}_s$ introduces edges that involve latent variables, and the minimal-graph operator $\mathcal{O}_{\min}$ absorbs redundant latent variables. The rank information over O is invariant to both operators.*

Interpretation. Theorem 2 states that RLCD returns a Markov equivalence class of a latent linear causal model. A latent variable in it is identified up to its atomic cover, which its pure children and its neighbors jointly determine (Definition 4). A discovered latent variable is therefore identified only by the items it reaches. For scoring, it is assigned to the published latent variable whose scoring key contains most of those items (Appendix D.2). The rank tests run on the correlation matrix of the standardized item responses, as in the RLCD paper.

Score-based search over observed variables.

Proposition 2 (Correctness of BOSS; Andrews et al., 2023, Prop. 2)**.** *Let P belong to a curved exponential family and be causal Markov and causal faithful to G, and let X be drawn i.i.d. from P. Then BOSS with its backward phase returns the Markov equivalence class of G for every initial permutation as the sample size grows.*

Interpretation. On questionnaires, BOSS supplies the initial graph over the items and partitions them into groups. The rank tests of RLCD then rewrite these groups, and Theorem 2 states what they identify. On robot and vehicle channels, each consecutive pair of steps of an arm, log or drive is one row. The columns of a row are the state channels at the two steps and the action at the earlier step (Appendix E). BOSS therefore returns a graph over those columns. Under its hypotheses, Proposition 2 guarantees the skeleton and the unshielded colliders of that graph. A cause precedes its effect, so an edge between the two steps runs from the earlier step. The discovery step therefore keeps the edges that run forward in time (Appendix E). The discovered graph $\hat{G}$ is the input of the constrained embedding optimization.

What the objective instantiates. Eq. 3 imposes the lines of Eq. 2 on the three pair sets $\mathcal{P}_{\mathrm{ind}}$, $\mathcal{P}_{\mathrm{dep}}$ and $\mathcal{P}_{\mathrm{con}}$. It takes ψ as the identity on $\mathcal{P}_{\mathrm{con}}$, the thresholded $\tilde{\rho}_{ab}$ of Appendix F as the target on $\mathcal{P}_{\mathrm{ind}}$, and the residual r_v of Eq. 15 as $e_{v\cdot\mathrm{pa}}$. The objective uses two conditioning sets. With S empty, the independence term covers the marginally d-separated pairs, including the causes of a common effect, while the dependence term covers the most strongly dependent trek-connected pairs, and the dependence line enters as the one-sided bound $|\cos(e_a, e_b)| \geq \kappa\,|\rho_{ab}|$ (Definition 5). With S the parents, the target of the conditional dependence term is the dependence between the residuals of a and b on their own parents. Conditioning on the parents mirrors the generating process, since removing the contribution of the parents leaves the noise term of the variable. Under a correct graph with additive and mutually independent noise, the residuals of any two variables are independent, so this target vanishes. Where the measurements show a residual dependence, the same term matches the corresponding residual cosine similarity to that dependence. The objective imposes no relation for other conditioning sets.

Definition 5 (The estimated feasible set)**.** *With $\mathcal{P}_{\mathrm{ind}}$, $\mathcal{P}_{\mathrm{dep}}$, $\mathcal{P}_{\mathrm{con}}$, r_v and κ as in Eq. 3 and $\tilde{\rho}_{ab}$ the thresholded distance correlation of Appendix F, the estimated feasible set $\hat{\mathcal{A}}(G, X)$ consists of the assignments with $e_v = e_v^\star$ on V_{N} and*

$$\begin{aligned} r_v = 0 \;\; \forall v : \mathrm{pa}(v) \neq \emptyset, \qquad & \cos(e_a, e_b) = \tilde{\rho}_{ab} \;\; \forall (a,b) \in \mathcal{P}_{\mathrm{ind}}, \\ \cos(r_a, r_b) = \rho_{ab\cdot\mathrm{pa}} \;\; \forall (a,b) \in \mathcal{P}_{\mathrm{con}}, \quad & |\cos(e_a, e_b)| \geq \kappa\,|\rho_{ab}| \;\; \forall (a,b) \in \mathcal{P}_{\mathrm{dep}}. \end{aligned} \tag{9}$$

Proposition 3. *Let $\mathcal{L}$ be the objective of Eq. 3 with ρ_{ab} on $\mathcal{P}_{\mathrm{ind}}$ taken as $\tilde{\rho}_{ab}$ and with positive term weights. Let $\mathcal{R}$ be the regularizer of Eq. 13 and $\mathcal{L}_0 = \mathcal{L} - \mathcal{R}$. Here $\cos(r_a, r_b)$ is taken on the unit-normalized residuals and is defined as 0 when a residual vanishes. For every assignment e with $e_v = e_v^\star$ on V_{N},*

$$\mathcal{L}_0(e) = 0 \iff e \in \hat{\mathcal{A}}(G, X). \tag{10}$$

The set $\hat{\mathcal{A}}(G, X)$ is nonempty only if $\rho_{ab\cdot\mathrm{pa}} = 0$ on every pair of $\mathcal{P}_{\mathrm{con}}$, since $r_v = 0$ on every child sets every residual cosine similarity to 0. The set is also nonempty only if the marginal lines can hold together with the anchors.

Proof. Each term of $\mathcal{L}_0$ is a sum or a mean of squares, or of squared hinges, over one line of Eq. 9. Each term is therefore nonnegative, and with a positive weight it vanishes exactly when that line holds. Anchoring holds by assumption. With every residual zero the residual cosine similarities are 0 by the convention above, so the conditional line holds exactly when $\rho_{ab\cdot \mathrm{pa}}$ vanishes on $\mathcal{P}_{\mathrm{con}}$. □

Interpretation. The second step is a constrained multidimensional scaling of the matrix of dependences onto the unit sphere (Torgerson, 1952; Cristianini et al., 2001), with the anchors held fixed. Each marginal cosine relation is an entry of the Gram matrix of the unit-normalized assignment. Each conditional cosine relation is an entry of the Gram matrix of the normalized residuals. A minimizer of $\mathcal{L}$ with $\mathcal{L}_0 = 0$ lies in $\hat{\mathcal{A}}(G, X)$, and $\mathcal{R}$ selects one member of it. On finite data, $\mathcal{L}_0$ does not reach zero. A minimizer of $\mathcal{L}$ is an anchored assignment that best satisfies the lines of Eq. 9 in the least squares sense of Eq. 3. Where those lines leave the vectors free, $\mathcal{R}$ selects among them. Which member of $\mathcal{A}(G, X)$ carries the semantics of a variable is left to existing knowledge, which enters as $\mathcal{R}$ and as the language model (Section 2).

D Datasets and Preprocessing

D.1 Robotics scenarios

Simulated arms. For five arms we collect 1,000 episodes of 200 steps in `robosuite` (Zhu et al., 2020) on the MuJoCo engine (Todorov et al., 2012) under the default operational-space controller (Khatib, 1987), with random actions. For the Panda we use the 1,500 machine-generated lift episodes of 150 steps released with robomimic (Mandlekar et al., 2022), whose actions come from a policy. Discovery treats these episodes as it treats the other arms. Each simulator step is one row and each scalar state variable one column. Columns that are functions of other columns are dropped before the pipeline runs. The dropped columns are of three kinds. The concatenated state vectors repeat the individual channels, and the cosine and sine of each joint angle are functions of the angle column. The object-relative position is the difference of two recorded positions. The manipulated object is excluded, because its coupling to the end-effector exists only during contact and a static graph cannot express a state-dependent edge. Each arm is then written as a lag-one design matrix, which holds every channel at the previous and the current step and the action at the previous step. Table 3 lists the six arms. The evaluation uses 1,200 steps of each arm. The true graph of a simulated arm is written from the controller and the kinematics and checked against the simulator edge by edge. Appendix E compares the discovered structure with it. Figure 11 counts the channels of every evaluated arm, log and drive by family.

Real robot logs. The real logs are three datasets read through their LeRobot parquet form (Cadene et al., 2026). A UTokyo xArm pick-and-place log (Matsushima et al., 2023) and a Stanford HYDRA log (Belkhale et al., 2023) come from the Open X-Embodiment release (O'Neill et al., 2024), and a cup-handling log comes from the ALOHA release (Zhao et al., 2023). Their channels are the recorded state and action columns of each platform, from 13 to 28 per log. The same 1,200-step design matrix is built for each log.

Vehicle drives. The CAN bus (Controller Area Network) is the network on which the controllers of a car exchange messages. The engine, brake and steering controllers are three of them. Every message carries the current value of a few signals. A logging device attached to the bus records them without any document that says what a signal is. A drive is therefore a natural case of measured but unnamed variables. The vehicle drives come from comma2k19 (Schafer et al., 2018). A drive has twelve channels, six from the CAN bus and six from the inertial sensor of the recording device. The CAN bus supplies the vehicle speed, the steering wheel angle and the four wheel speeds. The inertial sensor supplies three accelerometer axes and three gyroscope rates. Each channel is linearly resampled to a common 10 Hz grid over the segments of the drive and cut to 2,400 samples, or to 1,199 on one shorter drive. The supplied graph for the drives encodes the standard vehicle dynamics in nine edges. Steering points to yaw rate and lateral acceleration, and speed points to each wheel speed and to yaw rate. Longitudinal acceleration points to speed, and roll rate points to lateral acceleration. The first two drives in sorted order supply signal statistics and the next ten are the evaluation drives. Figure 10 shows one arm and one drive as they enter the pipeline. A channel whose lag-one autocorrelation exceeds .95 is almost equal to its own lag and would hide every other

parent in a lagged search. The discovery step therefore replaces such a channel by its first difference before the search runs (Appendix E). Figure 10 also marks those channels, and Figure 12 shows how the ten drives differ in speed and steering.

Table 3: The robot and vehicle data sets of the evaluation and of the training. Channels count the columns of the lag-one design matrix at one step, actions included. Every reported robot number is computed on the evaluated sets.

set	members	number	channels	steps	role
simulated arms	Panda (lift), Sawyer, IIWA, UR5e, Kinova3, xArm7	6	32 to 44	1,200	evaluation
real logs	UTokyo xArm, Stanford HYDRA, ALOHA cup	3	13, 14, 28	1,200	evaluation
vehicle drives	comma2k19 evaluation drives	10	12	2,400, one drive 1,199	evaluation
generated arms and vehicles	operator and weight network	20	varies	—	training
	encoder fine-tuning	300	varies	—	training
	adapter, under the true graph and a degraded edge set	$1{,}600 \times 2$	varies	—	training

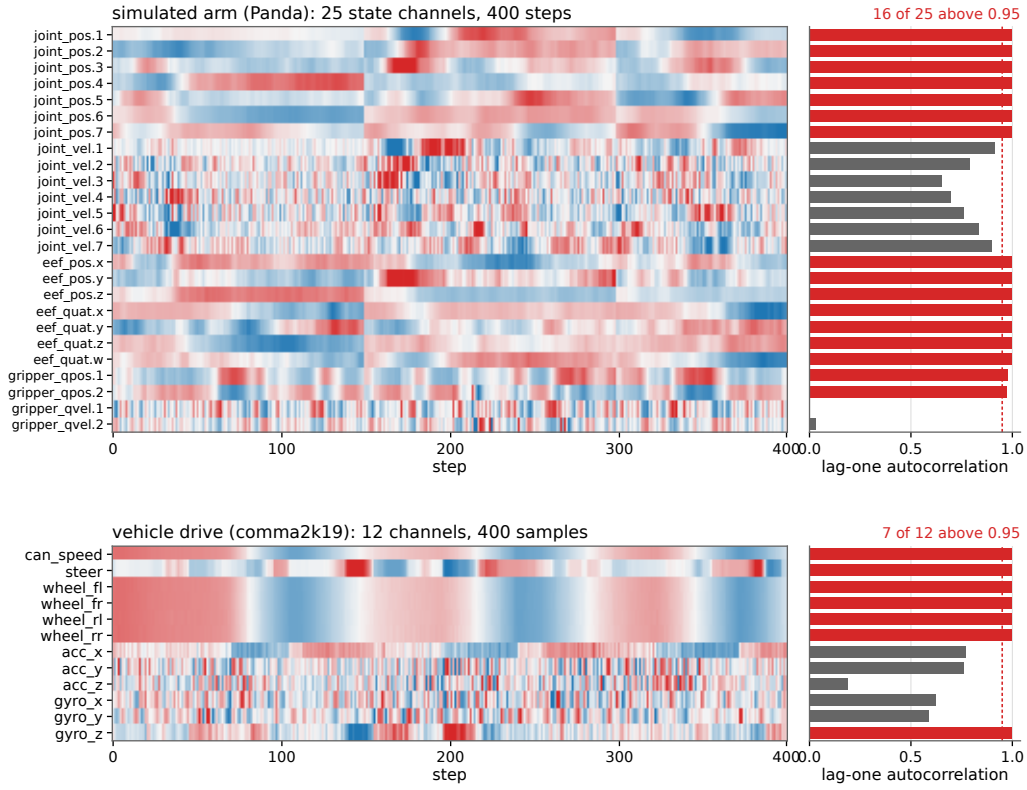

Figure 10: One simulated arm and one vehicle drive. Left, every state channel standardized over 400 consecutive steps. Right, the lag-one autocorrelation of each channel, in red above .95.

D.2 Questionnaires

Corpus and preprocessing. Table 4 lists all twenty-one scales. Seventeen are public response matrices from the Open-Source Psychometrics Project (OSP, 2026). The World Values Survey variables are the components of the indices of Welzel (2013), taken from the Survey's Wave 7 release (Haerpfer et al., 2024). The Holzinger-Swineford battery (Holzinger & Swineford, 1939) is the 1939

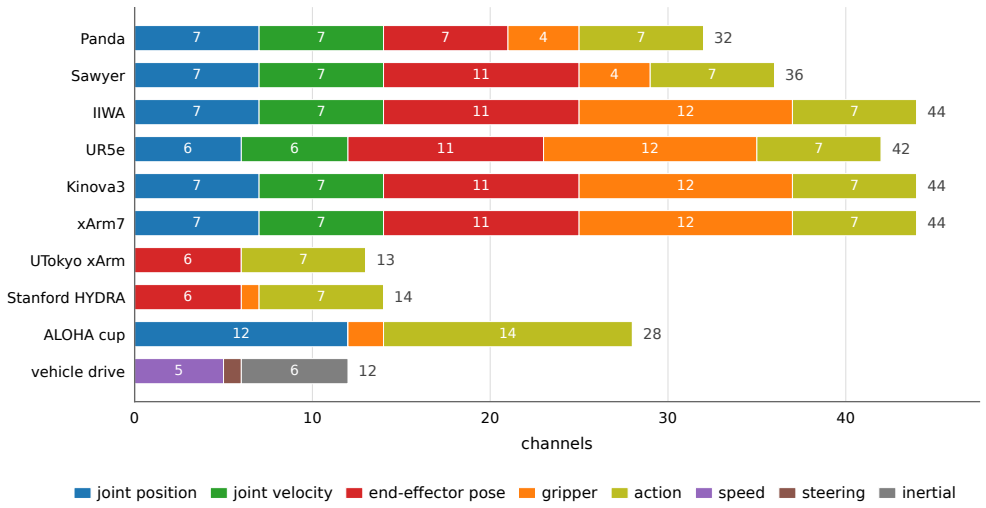

Figure 11: Channels of every evaluated arm, log and drive, by family. A simulated arm carries joint positions and velocities, the end-effector pose, the gripper and the commanded action, a real log what its platform recorded, and a vehicle drive the speed signals, the steering wheel angle and the inertial unit.

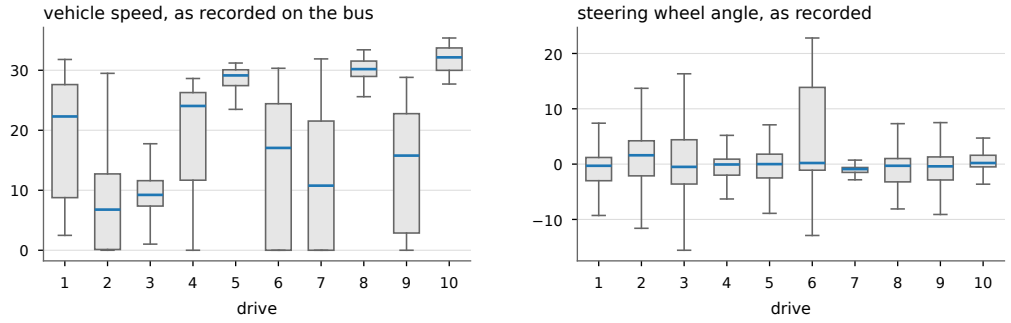

Figure 12: Vehicle speed and steering wheel angle over the ten evaluation drives, as recorded on the bus. Boxes span the middle half of the samples and whiskers reach 1.5 times that range. The drives range from stop-and-go traffic to steady highway speed.

set of cognitive tests, of which we use twenty-four. Seventeen measures of the Himi multitasking study (Himi et al., 2019a) and its TLVD benchmark subset (Du et al., 2026) come from the study's OSF archive (Himi et al., 2019b). Items are the observed variables. The published scoring key supplies the latent variables, which are the reference for the latent side. Where a scale has a nested key the nesting is kept. HEXACO has six domains over twenty-four facets, and the World Values Survey has the two overall indices of Welzel over seven of their eight sub-indices. A respondent with a missing entry on any item of the scale is dropped. The response matrix is then capped at 5,000 respondents drawn with a fixed seed, and every column is standardized. The correlation matrix that enters the discovery step is therefore complete. Figure 13 shows how the corpus is composed. The sixteen training scales span nine to 162 items and one to sixteen latent variables, and the trained parts of the optimization were fitted across this range. Figure 14 shows the five evaluation scales as correlation matrices with items ordered by latent variable. DASS, RIASEC and KIMS show clean blocks, and HEXACO shows twenty-four small blocks inside six larger ones. The block structure in that figure is what the discovery step has to find without the key. For each matrix, Figure 15 shows the absolute correlation of item pairs inside one latent variable and of item pairs across latent variables. The rank tests have to detect the contrast between these two distributions. The two medians lie farthest apart on the World Values Survey and RIASEC and closest on DASS, whose three latent variables correlate with one another. The discovered groups follow the key on four scales

and cut across it on the World Values Survey (Table 6). There a sub-index holds about three items, while the largest discovered group holds eight. The groups therefore span several sub-indices.

Reference for discovered latent variables. A discovered latent variable is given a reference by majority over the observed items beneath it. It carries no name of its own, since a rank test introduces it and only the items beneath it identify it. Every observed item that it reaches in the graph, directly or through nested latent variables, is looked up in the published scoring key. The published latent variable that owns most of those items becomes the target against which the written name is scored. Every discovered latent variable reaches at least one named item. On four scales the reference owns most of the items beneath a discovered latent variable. On the World Values Survey, whose discovered groups cut across the Welzel sub-indices, it owns about half of them (Table 6).

The rank tests introduce a latent variable from the rank deficiency of a cross-covariance block, so its identity is fixed only up to that block (Theorem 2). Factor analysis uses the same convention and matches an estimated factor to a reference factor by its loading pattern (Lorenzo-Seva & ten Berge, 2006). The majority rule is the discrete form of that match.

Table 4: Every questionnaire scale used in this paper, latent variables counted with any parent level. The source column cites the data release: the raw-data index of the Open-Source Psychometrics Project (OSP), the World Values Survey Wave 7 release, the `sem` package of R for the 1939 battery, and the Himi OSF archive cited above, shared by the Himi and TLVD rows.

scale	instrument	items	latent variables	respondents	source	role
DASS	Depression Anxiety Stress Scales (Lovibond & Lovibond, 1995)	42	3	5,000	(OSP, 2018b)	evaluation
HEXACO	IPIP-HEXACO scales (Ashton et al., 2007)	240	6 domains, 24 facets	5,000	(OSP, 2014b)	evaluation
KIMS	Kentucky Inventory of Mindfulness Skills (Baer et al., 2004)	39	4	521	(OSP, 2012c)	evaluation
RIASEC	Holland occupational interest types (Holland, 1997)	48	6	5,000	(OSP, 2018d)	evaluation
WVS	World Values Survey, Welzel indices (Welzel, 2013)	21	2 indices, 7 sub-indices	5,000	(Haerpfer et al., 2024)	evaluation
16PF	IPIP scales for Cattell's sixteen personality factors (Cattell et al., 1970; Goldberg et al., 2006)	162	16	5,000	(OSP, 2014a)	training
WPI	Woodworth Psychoneurotic Inventory (Woodworth, 1918)	116	1	3,888	(OSP, 2013b)	training
TMA	Taylor Manifest Anxiety Scale (Taylor, 1953)	50	1	4,474	(OSP, 2012e)	training
Big Five	IPIP fifty-item Big Five markers (Goldberg, 1992; Goldberg et al., 2006)	50	5	3,000	(OSP, 2018a)	training
HSQ	Humor Styles Questionnaire (Martin et al., 2003)	32	4	993	(OSP, 2012b)	training
SD3	Short Dark Triad (Jones & Paulhus, 2014)	27	3	5,000	(OSP, 2014d)	training
NPAS	Nerdy Personality Attributes Scale (OSP, 2015)	26	1	5,000	(OSP, 2018c)	training
HS	Holzinger-Swineford cognitive tests (Holzinger & Swineford, 1939)	24	5	301	(Fox et al., 2024)	training
HSNS+DD	Hypersensitive Narcissism Scale with the Dirty Dozen (Hendin & Cheek, 1997; Jonason & Webster, 2010)	22	4	5,000	(OSP, 2013a)	training
MACH-IV	Machiavellianism scale (Christie & Geis, 1970)	20	1	5,000	(OSP, 2019)	training
Himi	multitasking and executive function battery (Himi et al., 2019a)	17	6	202	Himi OSF archive	training
GCBS	Generic Conspiracist Beliefs Scale (Brotherton et al., 2013)	15	5	2,400	(OSP, 2016)	training
CFCS	Consideration of Future Consequences Scale (Strathman et al., 1994)	12	1	5,000	(OSP, 2012a)	training
SCS	Sexual Compulsivity Scale (Kalichman & Rompa, 1995)	10	1	3,243	(OSP, 2012d)	training
RSE	Rosenberg Self-Esteem Scale (Rosenberg, 1965)	10	1	5,000	(OSP, 2014c)	training
TLVD	the TLVD benchmark subset of the Himi study (Du et al., 2026)	9	4	202	Himi OSF archive	training

D.3 Selection of the evaluation data

Robot scenarios. The three scenarios are chosen by how much of the platform is documented, which is what the claim of the paper depends on. A simulator comes with the true graph of its physics, and a real platform comes with a declared channel layout. A vehicle CAN comes with no documentation from the maker of the car. Within each scenario the platforms are those whose channel layout is known. For the drives the layout comes from the comma2k19 release. Together they cover the kinds of channel system the method has to handle. The simulator supplies six single-arm platforms under one controller, with 32 to 44 channels each (Table 3). The real logs come from two releases. Open X-Embodiment, which is the shared corpus of robot demonstrations cited in the introduction, supplies two logs of single arms recorded at the end-effector. The ALOHA release supplies one log of a bimanual arm recorded at every joint. The three logs have 13 to 28 channels. comma2k19 supplies ten drives of one car, from stop-and-go traffic to steady highway speed (Figure 12). Figure 11 shows the channel families each platform carries, which is what differs between platforms.

Questionnaires. A scale enters the evaluation only if its scoring key is published and every latent variable in that key has a published name, so that every target has a reference. None of the five is a training scale. Within that rule the five are chosen to span the forms a questionnaire can take. They cover a clinical scale, a personality inventory, a mindfulness scale, a vocational interest inventory and a values survey. They run from 21 to 240 items and from 3 to 30 latent variables counted with their parent level (Table 4 and Figure 13). HEXACO and the World Values Survey carry a nested key, which allows a test of a latent structure of more than one level. The five also differ in how visible the latent structure is in the responses (Figures 14 and 15). DASS shows clean blocks, and the groups of the World Values Survey cut across the key.

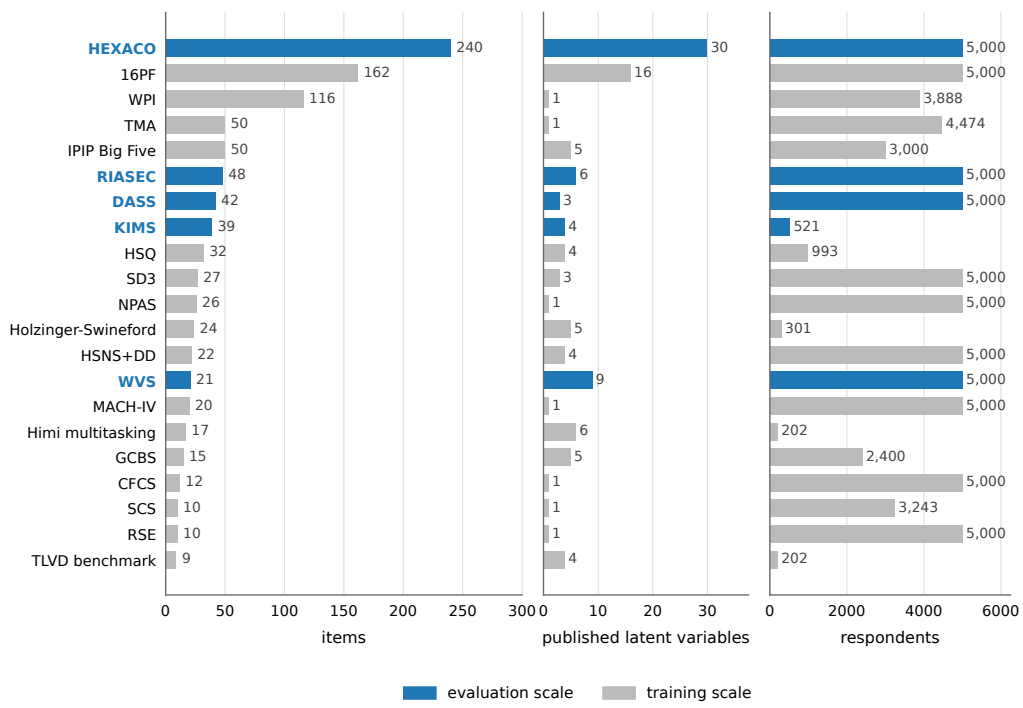

Figure 13: Composition of the questionnaire corpus. Bars give the items, the published latent variables and the respondents per scale, with the five evaluation scales in color. HEXACO counts its six domains and twenty-four facets together.

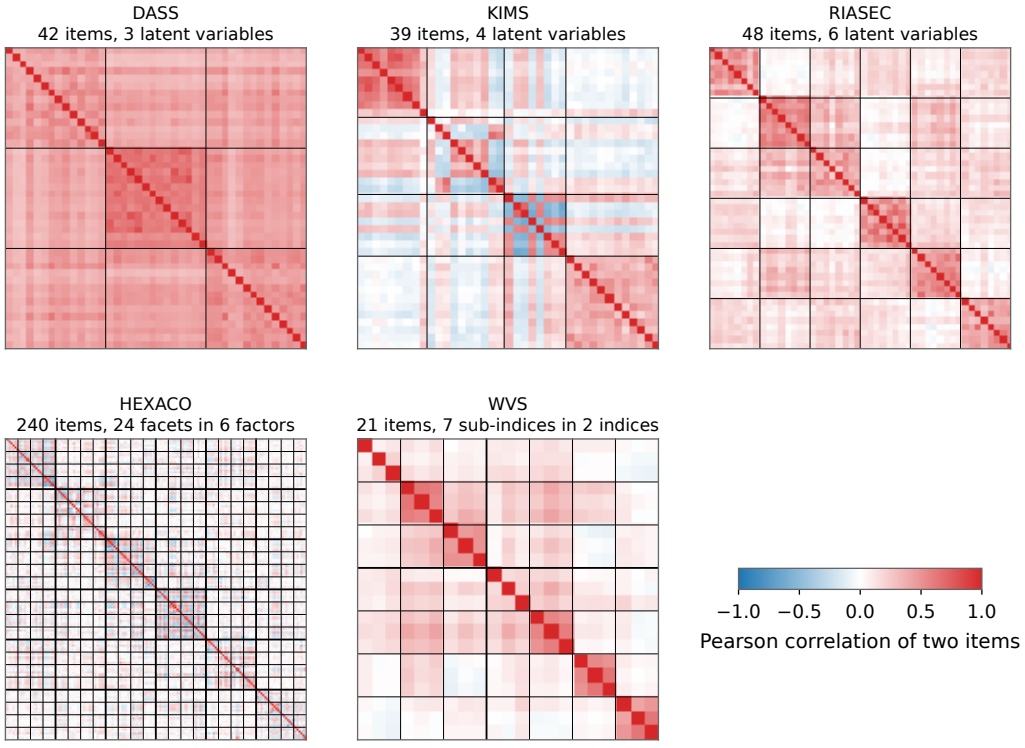

Figure 14: The five evaluation scales as item correlation matrices. Items are ordered by the published latent variable, and the borders of the latent variables are drawn. Red is positive and blue negative.

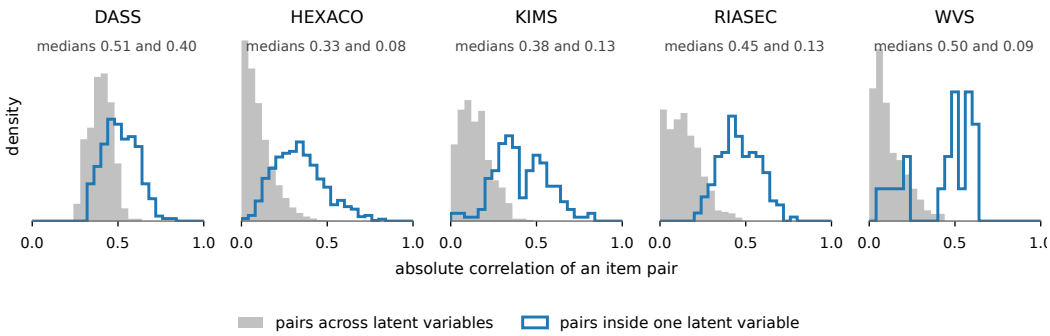

Figure 15: Absolute correlation of item pairs inside one published latent variable against pairs across latent variables, per evaluation scale, with the two medians printed in each panel.

D.4 Masking and columns

Masking. Names are masked at 20, 50, 75 and 90%. On a questionnaire scale five folds are drawn at each level, each with its own set of masked names. On an arm, log or drive one set is drawn at each level, so the variation reported for a robot column of Figure 4 is over its arms, logs or drives. The same set is reused for every method and every structure source. Only the name of an unnamed variable is masked. Its data stay available, since the task is to name a variable that is measured but undocumented. Every latent variable is a target at every level, and the level sets how many of its items keep a name. Figure 16 shows the numbers of masked items and named anchors on each evaluation scale.

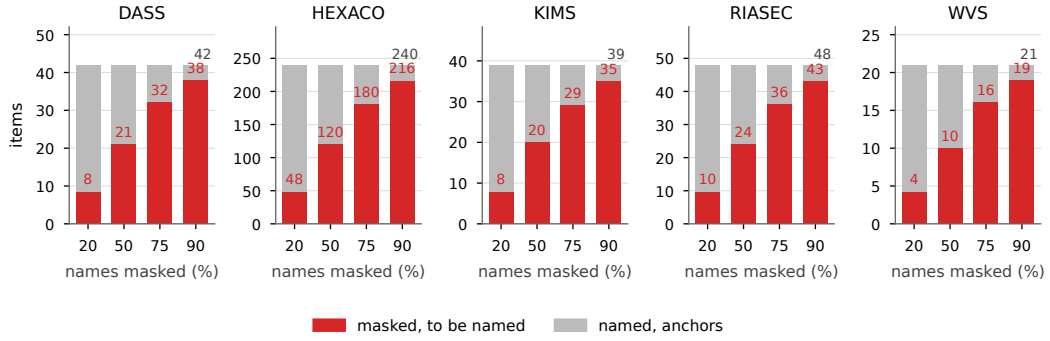

Figure 16: What one fold leaves named on each evaluation scale. Bars are the items of the scale. The masked part is what the fold asks the method to name, and the named part is what remains as anchors. The pink count above each masked part is the number of masked items, and the gray count is the total number of items in the scale.

D.5 Training data

The trained parts of CausalBridge belong to the optimization or to the adapter, and each has training data of its own. Table 5 lists every trained part against its data, and none of it overlaps the evaluation scales, arms, logs or drives.

The optimization. The operator, the weight network and the encoder fine-tuning of Appendix F are fitted on the sixteen training scales of Table 4 with their published keys. These public scales span nine to 162 items and one to sixteen latent variables. On the robot side the operator and the weight network are fitted on twenty generated arms and vehicles. The encoder fine-tuning is warm-started from the questionnaire one and fitted on 300 generated arms and vehicles, while it keeps unchanged the vectors of 20,000 English words and phrases sampled from the concept bank. This bank combines the ConceptNet Numberbatch vocabulary, WordNet and the Cognitive Atlas

(Speer et al., 2017; Miller, 1995; Poldrack et al., 2011). The semantic negation map is trained on WordNet antonym pairs and on the pole pairs of the sixteen training scales defined in Appendix F.

The adapter. The questionnaire adapter is trained on 600 synthetic scales. The local language model Qwen3-8B (Qwen Team, 2025) writes latent variable names for forty domains of psychology and eight first-person items for each, two or three of them reverse keyed. A scale is assembled from one to five latent variables of one domain. Likert responses for 600 respondents are then simulated from a factor model with loadings, reverse keying and correlations between the latent variables. A generated item or latent variable is dropped when its encoder cosine to any of the 1,121 real item texts and latent variable descriptions reaches .92. This filter removes 5,666 of 12,376 items and 108 of 1,547 latent variables. The 600 solved scales yield 33,073 training examples. Sets of masked names are drawn at the same four levels, half under the true graph and half under a degraded edge set. The adapter is thus trained on the kind of graph error that the discovery step makes. This design applies domain randomization (Tobin et al., 2017) to the structure. Of the 28,014 base examples, the prompt text is removed in 5,590 and the evidence line in 5,719. An unrelated variable's vector replaces z_v in 3,420 examples, and a named sibling's vector replaces it in 1,565. A further 5,059 examples repeat an item whose sibling is also unnamed, with the evidence line removed. These variants train the adapter to use both the prefix and the text. The robot adapter is trained on 1,600 generated arms and vehicles, each entered twice. One entry uses its true graph, and the other uses an edge set degraded to the quality of a discovered graph. The entries yield 391,352 examples in all, of which the examples of 96 entries are held out for early stopping.

Generated arms and vehicles. A generated arm or vehicle is a causal channel system that comes with its true graph, its measurements and a name for every channel. It takes one of the four shapes of the evaluation platforms. These are an arm, a log that records only the end-effector, a bimanual joint log and a vehicle CAN. An arm has five to eight joints, an end-effector pose block, a gripper and a commanded action. Each joint has an angle, which follows a commanded value with a lag, and a velocity. The edges run from the action to the joints, between the angle and the velocity of a joint, and from the joints to the end-effector pose and the gripper. A vehicle has the twelve channels of the vehicle CAN, with the steering exogenous. The wheel speeds and the vehicle speed lie on the drivetrain, and the inertial channels are responses. The signals are simulated from these dynamics with autoregressive noise over 600 to 1,200 steps, and the names come from template banks. About 15% of the channels have their sign flipped, so that the optimization is trained on inverted channels. Two real logs and the first two comma2k19 drives supply signal statistics for the generator and are not evaluated. The two logs are an ALOHA coffee task and a Berkeley AutoLab UR5 log (Chen et al., 2023).

Table 5: Every trained part of CAUSALBRIDGE and the data it was fitted on.

trained part	training data	size
optimization, questionnaires (operator, weight network, encoder fine-tuning)	the sixteen public scales with their published keys (Table 4)	16 scales
optimization, robots (operator, weight network)	generated arms and vehicles	20
encoder fine-tuning, robots	generated arms and vehicles, warm-started from the questionnaire one, 20,000 general words held fixed	300
semantic negation map	WordNet antonym pairs and pole pairs from the sixteen training scales	—
adapter and low-rank update, questionnaires	synthetic scales, after the leakage filter	600 scales, 33,073 examples
adapter and low-rank update, robots	generated arms and vehicles, each under its true graph and under a degraded edge set	$1{,}600 \times 2$, 391,352 examples

E CAUSAL STRUCTURE LEARNING

Questionnaires. The discovery method is RLCD as released in `causal-learn` (Zheng et al., 2024), with its first stage exchanged. BOSS on the correlation matrix replaces greedy equivalence search (Chickering, 2002) and partitions the items into the groups the rank tests then work within. The penalty discount of the score is raised where a group would otherwise be too large for those tests. Their cost grows exponentially with the number of latent variables in a group. On the 42-item DASS the original first stage had not finished after 35 minutes, while the pipeline with the

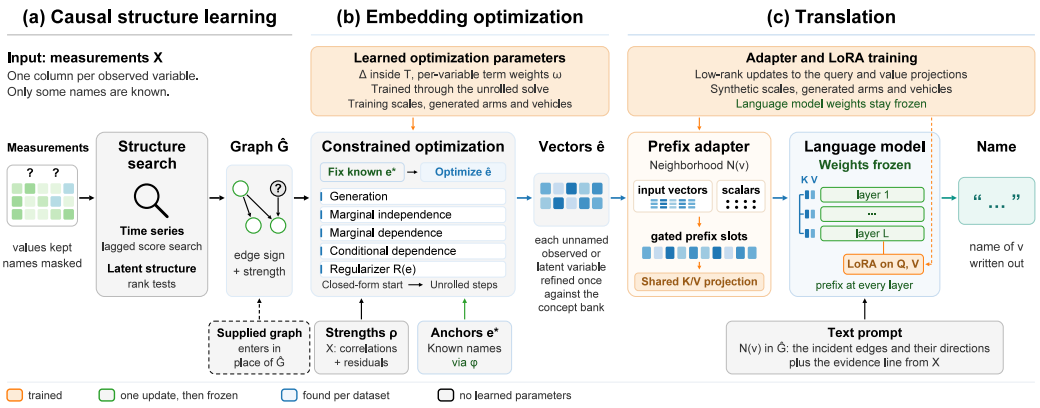

Figure 17: The objects CAUSALBRIDGE passes from measurements to a name. Structure learning returns the graph. The optimization assigns a vector to every unnamed variable under the relations of the graph, given the anchors. The adapter passes the vector to a frozen language model as a prefix, with the neighborhood written as text. Inputs enter from below the row and training from above it.

replacement produced the graph used here in under five. Rank tests then introduce latent variables. The search keeps an active set of covers, which starts as the single observed variables and gains every cover it finds. Within a group, the search looks for $k+1$ variables made of observed variables $\mathcal{X}$ and atomic covers $\mathcal{C}$ of the active set. The covariance of these variables with $\mathcal{X}$ and the rest of the group has rank k,

$$\operatorname{rank} \Sigma_{\mathcal{C}\cup\mathcal{X},\, \mathcal{X}\cup R} = k, \qquad |\mathcal{C}| + |\mathcal{X}| = k + 1, \tag{11}$$

where R holds the other variables of the group without the measured descendants of $\mathcal{C}$. Such a rank constraint means that k variables separate the two sides along every trek (Theorem 1). The search therefore introduces an atomic cover made of $\mathcal{X}$ and $k - |\mathcal{X}|$ latent variables, with $\mathcal{C}$ among its pure children (Dong et al., 2024, Theorem 8). The latent variables are new unless found before. The rank of a covariance submatrix is tested through its canonical correlations with a Wilks statistic (Dong et al., 2024, Appendix A.19), and a collider check prevents a collider from being mistaken for a latent parent. The search starts at $k = 1$, returns to $k = 1$ after every cover it finds and moves to $k + 1$ otherwise, up to a maximum cardinality. A cover that is found joins the variables of the next tests, so that a latent variable may have latent parents.

Execution. The rank tests run batched on one GPU in a single serial process, which keeps the search deterministic. Their accept and reject decisions agree with the released CPU test on all 1,000 test cases, and on a fifteen-variable scale the two return the same graph. HEXACO has 240 items over 5,000 respondents. Its structure is discovered in 20 seconds, since the first stage partitions the items into small groups before the rank tests run. A synthetic scale of 150 variables generated from 25 latent variables over 8,000 rows takes 107 seconds. Exact searches stay practical to about 150 active variables at $k \leq 3$, and a larger scale relies on the first stage keeping each group small. Every evaluation scale takes seconds to a few minutes (Appendix K).

Robot and vehicle channels. The columns are every channel at the previous and the current time step and the action at the previous time step. A channel whose lag-one autocorrelation is near one is replaced by its first difference. At high sampling rates, such a near-integrated channel would otherwise conceal every other parent. On the real logs and the vehicle drives a channel is replaced when its lag-one autocorrelation exceeds .95, and on the simulated arms every state channel is replaced. BOSS runs over all columns on the correlation matrix. With u_{t-1} and v_t two channels at the previous and the current time step, the retained edge set is

$$E \subseteq \{u_{t-1} \to v_t : u, v \in V\}, \tag{12}$$

and each child is then regressed on its retained parents by least squares, which gives every edge a signed coefficient. The lagged graph is then reduced to a summary graph over the channels, so every channel enters the next step as one variable. An edge $u_{t-1} \to v_t$ becomes $u \to v$, and a self-edge $u_{t-1} \to u_t$ is dropped. Two passes on partial correlations then refine the edge set. The first pass

runs on the simulated arms, whose actions are inputs. It adds an edge from an input to a channel when their partial correlation is high, given the channel's own lag and the other inputs. The second pass drops an edge whose partial correlation, given the other parents and the inputs, is low. Table 33 gives the two thresholds.

Recovered questionnaire structures. The discovered structure is finer than the published scoring key on four of the five scales and cuts across it on the World Values Survey (Table 6). This mapping fixes the scoring target of every discovered latent variable (Appendix D.2). A discovered latent variable is assigned to the published latent variable whose scoring key contains most of its items. Purity is the share of the items of a discovered latent variable that belong to its reference, averaged over the discovered latent variables of the scale. DASS has three published latent variables and nine discovered ones. Six of the nine lie inside one published latent variable, and the other three mix anxiety with stress. The method therefore splits each published latent variable into smaller groups of items and mixes only anxiety and stress. RIASEC and KIMS show the same pattern. HEXACO is the scale where the rank tests are most conservative, with twenty-five latent variables over 64 of the 240 items. The remaining items enter the next two steps with their BOSS edges and no latent parent. On the World Values Survey the groups cut across the key the most, with a purity of .53 against .80 to .95 on the other four scales. Appendix L shows the discovered graphs of DASS and the World Values Survey.

Table 6: The structure discovered on each evaluation scale against the published scoring key. Published latent variables are the groups of the key at the item level, and items reached counts the items under at least one discovered latent variable.

scale	published latent variables	discovered latent variables	latent to latent edges	items reached	purity
DASS	3	9	2	40 / 42	0.86
HEXACO	24	25	5	64 / 240	0.80
KIMS	4	11	5	29 / 39	0.88
RIASEC	6	13	8	28 / 48	0.95
WVS	7	12	14	21 / 21	0.53
all five	44	70	34	182 / 390	—

Table 7: The discovered structure of every arm, log and drive beside the supplied graph. Simulated arms are counted at the lagged level, real logs and vehicle drives at the channel level. Shared counts the edges the two graphs have in common.

arm, log or drive	supplied	found	shared
Panda	113	148	29
Sawyer	108	100	27
IIWA	122	106	31
UR5e	106	117	34
Kinova3	121	109	24
xArm7	119	92	28
UTokyo xArm	6	5	1
Stanford HYDRA	7	5	2
ALOHA cup	14	37	10
vehicle CAN, 10 drives	9	1 to 6	0 to 2

Recovered robot and vehicle structures. On the simulated arms the discovered graph has about as many edges as the true graph and shares about a quarter of the true graph's edges. The true graph is the simulator's own physics, so the discovered graph can be compared with it. Table 7 counts the edges of both graphs per arm in the time-lagged form of Eq. 12. Appendix L shows one arm, one log and one drive as a summary graph over the channels. The discovered graph recovers the edges from the actions to the joints and the edges from one step to the next among the joints. The edges it adds join channels that are correlated. The forward kinematics act within a time step, so the retention rule of Eq. 12 drops their edges. The edges from a velocity to the position it integrates are the ones the search misses most, since those channels enter the search as first differences. A graph of this quality is the one RQ1 compares with the supplied graph. On the arms and the real logs the

method names channels less accurately with it than with the supplied graph, and on the drives more accurately (Appendix J.2).

On the xArm and HYDRA logs and on the vehicle drives, the discovered graph shares few edges with the supplied graph. The vehicle drives are the informative case. The supplied graph has an edge from the vehicle speed to each of the four wheel speeds and nine edges in all. The search finds one to six edges per drive, and most of them lie among the accelerations and the rates. The yaw rate enters the lateral acceleration on five drives and the vertical acceleration enters the longitudinal acceleration on five. The search shares at most two edges with the supplied graph on any drive, and it returns none of the four edges from the vehicle speed to the wheel speeds. In RQ1 the method names the channels of the vehicle CAN more accurately with this sparse graph than with the supplied graph (Appendix J.2).

A physical law as an optional prior. The discovery step can also admit the edges within a time step that the physics of the platform orients. On an arm these run from the joints to the end-effector pose and the gripper. On a drive they run from the steering wheel angle and the wheel speeds to the vehicle speed, and from those to the accelerations and the rates. Such a law contains no information about the name of a channel, so it can be supplied wherever a platform makes it available. Appendix J.2 reports how this prior changes the names.

F Constrained Embedding Optimization

Space and variables. Embeddings lie in the space of `e5-large-v2` (Wang et al., 2022). The encoder fine-tuning trains small low-rank matrices in the encoder on the sixteen training scales of Appendix D. Its loss uses the same dependence and independence relations as the objective. A dictionary of English words and phrases from the resources in Appendix D.5 is held fixed during this fine-tuning, so that the space keeps its semantics. The trained matrices are then frozen, and scoring uses the unmodified encoder. The vectors to be optimized are one per unnamed variable. The vectors of the named variables are the encodings of their names and enter as constants, which enforces the anchoring line of Eq. 2 exactly.

Objective. The objective is Eq. 3 of Section 3.2. $\mathcal{P}_{\mathrm{ind}}$ is the set of pairs that are d-separated by the empty set in G. In such a pair the two variables have no common ancestor, and neither is an ancestor of the other. $\mathcal{P}_{\mathrm{con}}$ is the set of pairs of variables that have parents, and $\mathcal{P}_{\mathrm{dep}}$ the set of trek-connected pairs of strongest dependence. The distance correlation ρ_{ab} is computed in the measurements and signed by the Pearson correlation of the two variables. In the marginal independence term it is set to zero where the absolute Pearson correlation falls below the significance threshold $2/\sqrt{n}$. This thresholded value appears as $\tilde{\rho}_{ab}$ in Eq. 9. A pair that is independent both in the graph and in the data therefore becomes orthogonal. A measured dependence that the graph does not imply is kept. The target $\rho_{ab\cdot\mathrm{pa}}$ is the distance correlation of the residuals of the two variables after a gradient-boosted regression (Friedman, 2001) on their own parents, signed by the Pearson correlation of the residuals. The marginal dependence term keeps the absolute cosine similarity of a strongly dependent pair above the fraction $\kappa < 1$ of its unsigned distance correlation, a lower bound that leaves the vectors the freedom to fit the other constraints. A latent variable enters every correlation of the objective through its score on the first principal component of its observed descendants (Jolliffe, 2002). The first four terms vanish exactly on the estimated feasible set (Proposition 3).

Optimization. We initialize the embeddings in closed form by ridge-regularized least squares (Tikhonov & Arsenin, 1977) with the linear approximation $T(e_u; \rho_{uv}) = \rho_{uv} e_u$. We then run unrolled Adam steps (Kingma & Ba, 2015; Domke, 2012). Unrolling keeps the optimization differentiable, so the shared operator and the per-variable term weights can be fitted through it. At inference the same loop optimizes the embeddings with the trained operator and term weights held fixed.

Refinement against the concept bank. The optimized vectors are then refined once against the concept bank by coordinate ascent. Each vector of an unnamed variable takes its fifty nearest bank phrases as candidates, together with the vector itself. A candidate is scored by its cosine similarities to the current vectors of the other variables, each weighted by the measured correlation with that

variable. The mean pairwise cosine similarity of the bank is subtracted from this score. At most three sweeps of coordinate ascent pick the best candidate for every unnamed variable, with the vectors of the named variables held fixed. Where a phrase is picked, the returned vector is the normalized mean of the optimized vector and that phrase. The operator and the term weights are fitted without this step.

Regularizer. The term $\mathcal{R}$ of Eq. 3 is

$$\mathcal{R}(e) = \lambda_{\text{bank}} \sum_{v \in V_{\text{U}}} \big(1 - \cos(e_v, \Pi(e_v))\big) + \lambda_{\text{nbr}} \sum_{v \in O \cap V_{\text{U}}} \big(1 - \cos(e_v, c_v)\big) + \frac{\lambda_{\text{norm}}}{|V_{\text{U}}|} \sum_{v \in V_{\text{U}}} \big(\|e_v\| - 1\big)^2, \tag{13}$$

where $\Pi(e_v)$ is the soft nearest-neighbor projection of e_v onto the concept bank, which is the softmax-weighted mean of its nearest bank entries. The concept bank is the English phrase bank described in Appendix D.5. The neighbor mean c_v is the correlation-weighted mean of the vectors of the few named variables that an unnamed observed variable correlates with most. On a questionnaire a negatively correlated named variable enters through the semantic negation map f_{neg} defined below, and on a robot it enters by its magnitude alone. The last term holds the vectors at unit norm.

Standard constructions in the objective. The conditional dependence term compares the residuals of the two vectors on their own parents, formed through r_v of Eq. 15, as a partial correlation compares residuals (Whittaker, 1990). Every relation enters as a quadratic penalty (Nocedal & Wright, 2006). The feasible set has more than one member, so the regularizer selects one of them. This is the role that regularization plays in an ill-posed problem, as in the ridge initialization above.

The trained components of the optimization. Three small networks are fitted once. The operator and the weight network are fitted on the sixteen training scales for questionnaires and on generated arms and vehicles for robots. The semantic negation map is fitted on the antonym and pole pairs described below. The parameters of the operator network Δ and of the weight network ω are fitted through the unrolled optimization with

$$\min_{\Delta, \omega} \frac{1}{|O \cap V_{\text{U}}|} \sum_{v \in O \cap V_{\text{U}}} \big(1 - \cos(\hat{e}_v, e_v^\star)\big) + \frac{1}{2\,|L|} \sum_{v \in L} \big(1 - \cos(\hat{e}_v, e_v^\star)\big) + 0.1\,\mathcal{S}(\Delta), \tag{14}$$

where $\hat{e}$ is the output of the unrolled optimization on a training scale, arm or vehicle with its names masked as at evaluation, $e_v^\star$ is the encoding of the masked true name, and $\mathcal{S}$ is the sign-consistency penalty of the operator described below.

The *semantic negation map* f_{neg} is a small network in the encoder space. It exists because a reverse-keyed item is generated from the opposite pole of its latent parent's meaning, and the embedding of the antonym differs from the negated vector $-e_u$. It is trained on WordNet antonym pairs in both directions and on pole pairs from the training scales. A pole pair is the mean of a latent variable's positive item texts against the mean of its negative ones. The loss is a cosine mapping term and an involution term for the condition $f_{\text{neg}}(f_{\text{neg}}(x)) \approx x$.

The *operator* T of the generating function $g_v(e_{\text{pa}(v)}) = \sum_{u \in \text{pa}(v)} T(e_u; \rho_{uv})$ of Section 3.2 is shared across edges,

$$r_v = e_v - \sum_{u \in \text{pa}(v)} T(e_u; \rho_{uv}), \qquad T(e_u; \rho_{uv}) = |\rho_{uv}| \,\big(b_{uv}(e_u) + \Delta(e_u, \rho_{uv}, \eta_{uv})\big), \tag{15}$$

with $b_{uv}(e_u) = e_u$ on an edge with $\rho_{uv} > 0$ and $b_{uv}(e_u) = f_{\text{neg}}(e_u)$ on an edge with $\rho_{uv} < 0$, and η_{uv} the type of the edge, which records whether it joins two latent variables. Δ is a small network conditioned on ρ_{uv} and η_{uv}. Its output starts at zero, so the operator initially scales the parent's vector or its image under the semantic negation map by $|\rho_{uv}|$. The correction is learned during training. T is additive over the parents, so a child receives nothing from a non-parent and the sparsity of the Jacobian follows the graph without a penalty. Δ can change the sign of a contribution, so Eq. 14 includes the sign-consistency penalty $\mathcal{S}$. This penalty aligns the contribution of a positive edge with the parent's vector and that of a negative edge with its image under the semantic negation map.

The *weight network* ω is a small network that emits per-variable weights for the terms of Eq. 3, which the equation omits. With these weights the balance between terms can differ between a variable with many named neighbors and one with few.

On the robot channels the same objective runs, with the parts that carry vocabulary replaced by their robot counterparts. The encoder fine-tuning uses a second set of low-rank matrices, warm-started from the questionnaire ones and fitted on generated arms and vehicles. The concept bank is a bank of templated robot and vehicle phrases. The operator and the weight network are the pair fitted on the generated arms and vehicles (Table 5).

G TRANSLATION VIA PREFIX ADAPTATION

Why the structure enters as a prefix. The prefix is the one input that can carry what the structure determines. For an unnamed variable the structure determines its relations to the named variables. The vector $\hat{e}_v$ holds these relations, to every named variable at once, with a strength and a sign. A prompt lists the names of a few named neighbors, which for an item are at most six siblings. When those neighbors are unnamed too or differ from the target by one index, their names no longer distinguish the variable. Its relations still distinguish it, so $\hat{e}_v$ enters the language model directly.

The prefix also carries the vector to where the name is generated. Every attention layer attends to the prefix through the KV cache, so the model attends to the vector as it generates the name. A soft prompt at the input layer (Mokady et al., 2021) is read once and reaches the later layers only through the hidden states. The language model stays frozen, so its representation of words is unchanged. The vector $\hat{e}_v$ already lies in the space of the frozen text encoder, which also holds the names. The adapter therefore learns one map per data kind and is trained once for each backbone.

The experiments support each point. The variant without the prefix still receives $\hat{e}_v$ in words, since its evidence line keeps the nearest phrases decoded from that vector. Under the discovered graph, its NRR@1 still falls below that of the variant without the evidence line. This holds on the vehicle CAN at every level and on every robot column from 75% masking on (Appendix J.4). Under the discovered graph on the vehicle CAN at 20% masking, the three backbones alone reach a type accuracy of .050, .450 and .000 in the order of Table 30, and with CAUSALBRIDGE .900, .950 and 1.000 (Appendix J.5). One recipe serves three backbones, and ten undocumented drives are named in about seven minutes (Section 4).

Inputs. The adapter input z_v of Section 3.3 is the concatenation of five vector slots $x_1, \ldots, x_5$, normalized when nonzero, and a vector s_v of a few scalars. The scalars describe the position of the variable in the graph. For an observed variable, the first three vectors are its vector $\hat{e}_v$, the vector of its latent parent and the component of the first orthogonal to the second. The fourth is the correlation-weighted mean of the vectors of its most correlated named variables. The fifth is that mean with negatively correlated members passed through the semantic negation map. An observed variable with no latent parent in the graph supplies its own vector in the second position. The discovered graph leaves part of the items of a scale without a latent parent (Table 6). For a latent variable, the loading-weighted mean of the vectors of its children takes the place of the variable. Its vector $\hat{e}_v$ takes the place of the latent parent, and the two means are taken over its children. The scalars record the position of the variable among its neighbors. They include its correlation to its most correlated named variables and its signed loading on the latent parent. They also include the share of its siblings that are unnamed and the cosine similarity between its vector and the neighbor mean. A robot channel keeps the five vectors, with the fifth set to zero since a channel carries no keying. Its scalars describe its parents and children, its lagged correlations and its role as an actuator or a sensor.

Architecture. Each input vector, with the scalars appended, passes through its own two-layer network q_i into two prefix vectors, so the prefix length is $m = 10$. A learned position embedding is added. One sigmoid gate γ_i per input vector scales the two prefix vectors of that input. A gate network q_γ computes the gate from the scalars and from which of the five vectors are present, and the gate is initialized near one. For each layer j one pair of learned matrices W_j^K and W_j^V, shared by every variable, turns the prefix into the KV cache of that layer. The adapter and the low-rank update

of the model together hold about 51 million parameters for a model of four billion. In symbols,

$$
\begin{aligned}
z_v &= [\, x_1;\ \ldots;\ x_5;\ s_v \,], \qquad \gamma = \sigma\big(q_\gamma(s_v, \|x_1\|, \ldots, \|x_5\|)\big), \\
P_v &= q(z_v) = \big[\, \gamma_1\, q_1(x_1, s_v);\ \ldots;\ \gamma_5\, q_5(x_5, s_v) \,\big] \in \mathbb{R}^{m \times d_p}, \\
\mathbf{K}_j &= P_v W_j^K, \qquad \mathbf{V}_j = P_v W_j^V \qquad \text{for every layer } j,
\end{aligned}
\tag{16}
$$

where each $q_i(x_i, s_v)$ is a $2 \times d_p$ block, and the gates γ raise the weight of the prefix where the neighborhood has few named variables.

Training. The language model stays frozen. A low-rank update is inserted into the query and value projections of every layer, and only that update and the adapter are optimized. The loss is

$$
\mathcal{L}_{\text{adapter}} = \mathcal{L}_{\text{NLL}} + \lambda_{\text{sim}} \left(1 - \cos(\bar{\phi}_{\text{LM}}(\hat{y}), \bar{\phi}_{\text{LM}}(\ell(v)))\right) + \lambda_{\text{ctr}}\, \mathcal{L}_{\text{ctr}} + \lambda_{\text{sib}}\, \mathcal{L}_{\text{sib}}, \tag{17}
$$

where $\mathcal{L}_{\text{NLL}}$ is the negative log-likelihood of the true name at its token positions. The second term is the semantic similarity loss of Section 3.3. Here $\bar{\phi}_{\text{LM}}(\hat{y})$ is the mean over those positions of the expected embedding of the predicted token distribution $\hat{y}$. The expected embedding is taken under the token embedding table ϕ_{LM} of the model. The term pulls $\bar{\phi}_{\text{LM}}(\hat{y})$ toward the mean embedding of the tokens of the true name in the same table. The contrastive loss $\mathcal{L}_{\text{ctr}}$ is a softmax cross-entropy over the cosine similarities between that expected embedding and the embeddings of the true name and of a few sibling names. The sibling loss $\mathcal{L}_{\text{sib}} = -\log(1 - p_{\text{sib}})$ penalizes the probability mass p_{sib} that a token position of the name places on tokens unique to a sibling name. The last two push the model toward a name that separates the variable from its siblings.

One adapter is trained on synthetic scales for the questionnaires and one on generated arms and vehicles for the robots, both with the same recipe. The examples are corrupted on the fixed schedule of Appendix D, so that the adapter learns to use both the prefix and the text. Latent targets are drawn more often than item targets, and training stops early on a validation split of the synthetic scales and of the generated arms and vehicles. Only the shape of the KV cache enters the adapter, so the recipe is the same for every backbone.

Prompt. The prompt text t_v is a fixed instruction, the incident edges of the variable and one line of structural evidence, which we call the evidence line. The instruction states what the prefix carries and asks for one short name. Each variant receives an instruction whose wording is true of its own input. The edge block is rendered from the graph with one line per incident edge.

```
CAUSAL NEIGHBORS:
- CAUSE: an unnamed latent factor also causes:  "⟨named
sibling⟩"; ...
- CAUSE: "⟨named parent⟩" influences it
- EFFECT: it influences "⟨named child⟩"
- EFFECT: it causes the item "⟨named child, or (label
hidden)⟩"
```

An item shows at most six named siblings and a latent variable at most twenty named children. An unnamed neighbor of a latent variable appears as the literal string `(label hidden)`, so the prompt shows how much of the group is named. The identifier of the target never appears. The evidence line reports the strongest signed correlations with named variables and the membership and keying of the variable within its group. It also reports the nearest named neighbors of the variable in the embedding space with their cosine similarities and its partial correlations given the group. Last, it gives the sibling that shares the most variance with the variable beyond the group. The robot form reports lead and lag relations with signed correlations. The evidence line also carries the nearest bank phrases decoded from the variable's own vector $\hat{e}_v$ and from the vectors of its unnamed neighbors. Where a neighborhood carries no name at all, those phrases are the only information left.

Decoding. Generation is greedy and capped at a short budget, and the first line of the output is taken as the name. One variable therefore costs one prompt and one greedy decoding call. An empty or unparseable output is a failure and ranks last.

H EVALUATION PROTOCOL

Scoring space. Every method is scored from the text it writes, in one frozen public embedding space shared by every method. We use `intfloat/e5-large-v2` at a pinned revision and rank by cosine similarity. The encoder's `query:` prefix is applied to both the written name and the candidates.

Candidate pool. The pool is every name of the same arm, log, drive or scale. For a robot channel it is every channel name of that arm, log or drive. For an item it is every item text of the scale, and for a latent variable every latent variable name of its key. Because the known names stay in the pool, a method that copies a known name scores zero on NRR@1 by construction. Under the discovered graph, the targets are the discovered latent variables, each assigned to a latent variable of the key by the rule of Appendix D.2. The pool stays the full list of the key. A tie takes the mean rank of the tied positions, and an empty output ranks last.

Metrics. NRR@1 stands for the recall at rank 1 of name retrieval. It is the rate at which the true name is ranked first among the candidates. MRR is the mean reciprocal rank of the true name. The identification metric asks a judge to pick the target out of five candidates and credits the case only when the judge picks it in both presentation orders. Naming quality, reported for latent variables, adds two requirements to identification. One requires the output to have the form of a name, and one rejects an output that repeats an item text verbatim. Every cell is the mean over the cases of its scales, arms, logs or drives, so a scale with more unnamed items weighs more. Every questionnaire cell is a mean over folds. Three auxiliary metrics are reported in Appendix J. Sibling discrimination accuracy (SDA) is computed over the variables that have a sibling. It counts the case when the written name lies closer in the scoring space to the true name than to every sibling of the variable. The siblings are the published siblings of an item or the channels of the same type on an arm, log or drive. The metric therefore measures whether a name that reaches the right latent variable or channel type also reaches the right item or channel. Robot channels report two further metrics. Type accuracy assigns the written name a channel type and counts the case when that type is the type of the true name. The type of a true name follows from its documented wording. A channel that a platform names after an arm joint, such as the waist or the elbow, is a joint channel. A written name that equals a documented channel name takes the type of that name. Any other written name is embedded in the scoring space and takes the nearest of 21 fixed channel-type descriptions. These descriptions range from the angle of one arm joint to the rotation speed of one wheel. Coordinate accuracy counts the case when the digit, axis letter or side word in the written name equals the one in the true name, over the names that carry one.

The identification judge. The judge is Gemma 3 12B in its instruction-tuned release (Gemma Team, 2025), served locally at temperature 0. For one case it receives the written name and five candidates, the target among them. The four distractors for an item are two published siblings of the item, drawn from the same latent variable, and two other items of the same scale. The distractors for a latent variable are up to four sibling latent variables of the same scale. A scale with fewer siblings fills the remaining slots with latent variables of other scales. Under the discovered graph the siblings are the latent variables of the key that the discovered groups reach. For a robot channel, two distractors are the channel names of the same arm, log or drive that share the most words with the true name. The other two are further channel names of that arm, log or drive. A name that fits only the latent variable or the channel type therefore competes with the siblings. On items a description that names only the latent variable is right about one time in three. Every case is judged twice, once in a seeded candidate order and once in the reverse order (Zheng et al., 2023; Wang et al., 2024). A case counts as correct only when the judge picks the target in both orders. This rule controls for the position bias of a language-model judge. A case is kept only when both responses parse.

The two requirements of naming quality. Naming quality is identification with two further requirements. The form check asks the same judge whether the output names a concept or quotes a specific statement, and only a name passes. The verbatim check runs offline and overrides the judge. The check lowercases the output, collapses whitespace and removes a final period. An output that then equals an item text of the same scale is rejected whatever the judge said. Together the two

checks remove the degenerate solution of repeating a neighbor. Identification alone can credit such a repeat when the neighbor belongs to the same latent variable.

Variation across folds. Every questionnaire number is a mean over the five folds of its scale. Every robot number is the mean over the cases of the arms, logs or drives of its column, each with one set of unnamed variables per masking level. The spread between folds is wide and grows as names are removed, since a high masking level leaves fewer anchors and each one weighs more. On DASS latent variables under the discovered graph, identification is $.689 \pm .183$ at 20% masking and $.222 \pm .208$ at 90%. The $\pm$ term is the standard deviation across the five folds of nine discovered targets each. On RIASEC the same quantities are $.800 \pm .088$ and $.369 \pm .084$ over thirteen discovered targets per fold. Differences between methods within one fold are therefore better evidence than differences between cells, which is why the comparisons in this paper are same-fold and same-target. The one exception is latent variables with no structure supplied, where CAUSALBRIDGE names the latent variables it discovers (Appendix J.2). The band around our curves in every result figure is one standard error over replicates. It equals the sample standard deviation across the replicates divided by the square root of their number. On a questionnaire column a replicate is one fold, with the five scales pooled by sample inside it, so the band rests on five values. On a robot column a replicate is one arm, log or drive, so the band rests on the six simulated arms, the three real logs or the ten drives. The value of one arm, log or drive at 20% masking rests on a few unnamed channels, which is why the robot bands are widest there. The bands are clipped to the unit interval. Appendix J reports the paired differences to the strongest baseline on these replicates.

I BASELINES AND FIDELITY TO THE PUBLISHED DESIGNS

The comparison covers the three ways a variable has been named from its data. A method selects a concept from a fixed bank, asks a language model, or regresses into the embedding space of the names. With a structure supplied the comparison adds two methods that consume the graph.

- **CLIP-Dissect-e5** (Oikarinen & Weng, 2023) selects the concept from a fixed bank whose activation pattern best matches the variable.
- **Automated Interpretability** (Bills et al., 2023) prompts a language model to explain the variable from a sample of its cases and then scores the explanation by simulation.
- **Delphi** (Paulo et al., 2025) explains the variable from a sample of its activating cases and adds detection scoring passes.
- **SASC-style** (Singh et al., 2023) prompts a language model for a short explanation of the variable's response profile.
- **DeViSE-style** (Frome et al., 2013) regresses the variable's correlation profile into the embedding space of the names and decodes the predicted vector to its three nearest entries in the concept bank of Appendix F. With fewer than two named variables there is no regression to fit and the method returns no name, which NRR@1 and MRR score as a miss.
- **Base language model reading the graph** receives the same prompt as CAUSALBRIDGE, which holds the edges and the evidence line. It runs with no prefix and no low-rank update, so it isolates what the prefix and the low-rank update contribute.
- **GraphMAE** (Hou et al., 2022) reconstructs the masked node representation by masked graph autoencoding with a GCN encoder (Kipf & Welling, 2017) and decodes the reconstruction to the nearest known name. On a latent variable it decodes to the nearest named item text. A masked name is therefore never among its candidates.

With no structure supplied, every baseline names latent variables behind a shared factor-analysis front end, since naming a latent variable requires a grouping and the front end supplies it. With a structure supplied, each baseline takes its latent grouping from that graph, except SASC-style and DeViSE-style, which keep the factor-analysis grouping. The front end is given the list of latent variables of the key and matches each to the factor whose item set overlaps it most. For a parent-level latent variable of a nested key the items are those of its child latent variables. A baseline therefore names every latent variable of the key, and one factor can serve two of them. CAUSALBRIDGE names the latent variables it discovers, referred to the key as Appendix D.2 describes.

Name-copy references. Three references build a name from the known names alone, and Appendix J reports them apart from the roster above. Correlation kNN returns the name of the most correlated named variable. Feature Propagation (Rossi et al., 2022) diffuses the vectors of the known names over a correlation graph that the port builds. It decodes the result to the nearest known name on an arm, log or drive and to the nearest phrases of the concept bank on a questionnaire. EFA loading naming returns the text of the item that loads highest on the factor.

Common protocol. Every baseline that calls a language model calls the same pinned Qwen3-4B snapshot, served locally at seed 0 in bfloat16. This is the default backbone of our own method. The one exception is SASC-style, which samples its candidate explanations at temperature .7 and scores them deterministically. Every output of our method and of the baselines is ranked in the same frozen encoder and judged by the same judge in the same dual-order protocol. The cases of a baseline copy the fold, the set of masked names and the target list from ours. The exception is latent variables with no structure supplied, where a baseline names the latent variables of the key. Every method sees the known names of the fold only. A missing or refused output is a failure on NRR@1 and MRR and is left out of the identification rate. Methods with no structure supplied receive no edges and no grouping from the key. The factor analysis supplies the grouping, and the key's item sets serve only to assign each factor its scoring target.

Fidelity to the published designs. Each reimplementation is a port to a new domain.

- **Automated Interpretability** keeps the explain-then-simulate structure, standardized activations, simulation on a disjoint held-out set and correlation scoring. The examples are respondent profiles or robot windows, where the original uses token windows. They are drawn across the activation range, where the original explains from top-activating snippets.
- **Delphi** uses the contrastive explainer of the library, which explains top-activating examples against hard negatives. The hard negatives here are the response-space nearest neighbors of the strong examples. The port keeps detection scoring on balanced held-out profiles. The port scores three candidate explanations jointly in one detector request, with disjoint validation and test stages.
- **CLIP-Dissect-e5** keeps SoftWPMI and the rank-reorder scoring at the source defaults. Probe images become profile texts built from the known names of the fold, and CLIP (Radford et al., 2021) becomes the frozen e5 encoder, which the table name records. The two similarity functions are combined by rank averaging, since their scales differ.
- **SASC-style** keeps the two phases, in which it proposes short candidate explanations and scores each against data. Candidates come from anchor correlations, where the original uses n-gram summaries. The score is sign agreement between predicted and measured correlations, since the original scorer runs the module on new inputs and an unnamed survey item has no new respondents. This is the largest structural change among the language model baselines, and the table name records it.
- **GraphMAE** keeps masked feature reconstruction with learnable mask tokens, re-mask decoding and the paper's scaled cosine error. The encoder is a GCN, where the paper defaults to GAT (Veličković et al., 2018). A latent variable has no node text to reconstruct, so its prediction is the decoder output of its neighbor-conditioned state.
- **DeViSE-style** keeps only the mapping of a variable into the embedding space of the names and the decoding by nearest neighbors. Ridge regression on correlation features replaces the deep visual encoder and its hinge ranking loss, so the training signal, the model class and the loss all differ. The table name records the change.
- The **factor-analysis front end** is our own component. It uses parallel analysis (Horn, 1965) for the number of factors, varimax rotation (Kaiser, 1958) and argmax loading membership.

The scored output of Automated Interpretability and Delphi is a short name of one to four content words, which our interface requires. Their one-sentence explanation is stored, since ranking a sentence against a set of names would measure something else.

Table 8: No structure supplied, CAUSALBRIDGE on the graph it discovers. Bold is the best value in a column within a set of variables and underline the second best. A dash is a method that returned no name at that level. Appendix J carries every metric each set of variables reports.

		NRR@1				MRR				identification			
variables	method	20	50	75	90	20	50	75	90	20	50	75	90
simulated arms	CAUSALBRIDGE	**.083**	**.140**	**.137**	**.046**	**.224**	**.286**	**.278**	**.194**	**.106**	**.203**	**.208**	**.116**
	CLIP-Dissect-e5	.083	.058	.027	.027	.185	.186	.116	.120	.062	.050	.033	.060
	AutoInterp	.000	.008	.011	.005	.078	.095	.095	.086	.000	.033	.039	.033
	Delphi	.000	.025	.016	.005	.079	.120	.119	.083	.062	.034	.072	.028
	SASC-style	.021	.058	.027	.018	.112	.162	.130	.118	.042	.074	.060	.028
	DeViSE-style	.021	.033	.022	.032	.143	.136	.131	.116	.000	.025	.016	.005
real logs	CAUSALBRIDGE	**.417**	**.296**	**.195**	**.120**	**.485**	**.385**	**.314**	**.248**	**.417**	**.296**	**.244**	**.180**
	CLIP-Dissect-e5	.167	.074	.049	.080	.305	.258	.205	.226	.000	.037	.100	.080
	AutoInterp	.083	.037	.049	.020	.265	.200	.178	.152	.250	.111	.098	.041
	Delphi	.000	.037	.073	.120	.163	.212	.215	.228	.250	.120	.024	.100
	SASC-style	.000	.111	.024	.060	.145	.223	.138	.184	.083	.115	.073	.080
	DeViSE-style	.083	.074	.024	.000	.216	.188	.178	.084	.083	.074	.098	.120
vehicle CAN	CAUSALBRIDGE	**.800**	**.367**	**.244**	**.218**	**.859**	**.570**	**.412**	**.397**	**.800**	**.400**	**.300**	**.282**
	CLIP-Dissect-e5	.000	.067	.078	.091	.135	.229	.298	.286	.000	.050	.089	.118
	AutoInterp	.000	.000	.000	.000	.184	.245	.171	.191	.050	.017	.000	.009
	Delphi	.000	.017	.067	.118	.195	.292	.241	.281	.000	.067	.033	.055
	SASC-style	.000	.167	.044	.109	.276	.366	.223	.268	.000	.250	.146	.191
	DeViSE-style	.000	.000	.000	.000	.180	.173	.141	.083	.050	.067	.022	–
questionnaire items	CAUSALBRIDGE	**.121**	**.098**	**.059**	**.028**	**.253**	**.216**	**.164**	**.105**	**.356**	**.364**	.283	.217
	CLIP-Dissect-e5	.028	.023	.022	.014	.096	.095	.099	.071	.163	.179	.183	.153
	AutoInterp	.010	.012	.008	.007	.058	.055	.054	.054	.085	.103	.101	.099
	Delphi	.008	.010	.013	.011	.060	.055	.060	.063	.141	.122	.133	.113
	SASC-style	.079	.062	.035	.021	.211	.175	.128	.087	.356	.342	**.303**	**.219**
	DeViSE-style	.033	.023	.014	.008	.131	.110	.087	.064	.238	.227	.193	.164
latent variables	CAUSALBRIDGE	**.540**	**.486**	**.386**	**.240**	**.660**	**.623**	**.560**	**.424**	**.531**	**.511**	**.441**	**.254**
	CLIP-Dissect-e5	.073	.119	.127	.142	.257	.288	.298	.292	.128	.120	.151	.121
	AutoInterp	.096	.100	.108	.088	.258	.259	.270	.252	.160	.140	.170	.139
	Delphi	.123	.108	.085	.069	.266	.269	.251	.235	.138	.108	.100	.115
	SASC-style	.369	.315	.204	.115	.523	.479	.393	.291	.502	.424	.319	.204
	DeViSE-style	.138	.138	.108	.112	.291	.281	.260	.266	.151	.154	.123	.112

J SUPPLEMENTAL EXPERIMENTAL RESULTS

This appendix supplements the first three research questions of Section 4 in their order. Appendices J.1 to J.3 belong to RQ1. They give the lead of our method on every metric, the comparison of the discovered graph with the supplied one, and the same lead replicate by replicate and scale by scale. Appendix J.4 belongs to RQ2 and takes the components in turn. Appendix J.5 belongs to RQ3. The complete tables close the appendix.

J.1 RQ1: THE LEAD WITH NO STRUCTURE SUPPLIED

The finding of RQ1 is that CAUSALBRIDGE leads on every column and that the lead is largest where the names give a language prior the least information. Tables 9 to 11 give every metric behind that claim beside the three metrics of Table 8. These are the sibling discrimination of items and channels, the type and coordinate accuracy of channels and the naming quality of latent variables. At 50 and 75% masking on the three robot columns, the NRR@1 of our method is 2.2 to 5.1 times that of the strongest baseline. Its identification rate is 1.6 to 2.9 times and its MRR 1.4 to 2.1 times that of the strongest baseline.

A channel label gives a language prior almost no semantic information. On the three robot columns the baselines remain low at every masking level. The neighbors of a channel are labels that describe no signal, and a sampling method receives only numbers. On the vehicle CAN they even rise a little as names are masked, Delphi from .000 to .118. A sampling method writes the same few names for every channel, and such a name earns credit only once that channel is unnamed. At 90% masking SASC-style writes rear right wheel speed for 55 of the 110 targets and front right wheel speed for 19. At that level one to three names remain per real log and one per drive, so the anchors are nearly gone and the lead closes. On the real logs CAUSALBRIDGE and Delphi both retrieve .120.

A channel name has two parts, and CAUSALBRIDGE *leads on both.* The type of a channel says what is measured, such as the angle of a joint. Its coordinate says which one, by an index, an axis

Table 9: Robot channels with no structure supplied, CAUSALBRIDGE on the graph it discovers. The table gives every metric a robot channel reports, which are the three of Table 8, sibling discrimination, type accuracy and coordinate accuracy (Appendix H).

variables	method	NRR@1 20	NRR@1 50	NRR@1 75	NRR@1 90	MRR 20	MRR 50	MRR 75	MRR 90	SDA 20	SDA 50	SDA 75	SDA 90
simulated arms	CAUSALBRIDGE	**.083**	**.140**	**.137**	**.046**	**.224**	**.286**	**.278**	**.194**	.128	.190	.153	.164
	CLIP-Dissect-e5	.083	.058	.027	.027	.185	.186	.116	.120	.234	.155	.142	.169
	AutoInterp	.000	.008	.011	.005	.078	.095	.095	.086	**.255**	.207	.165	.155
	Delphi	.000	.025	.016	.005	.079	.120	.119	.083	.043	**.241**	**.193**	**.197**
	SASC-style	.021	.058	.027	.018	.112	.162	.130	.118	.170	.147	.165	.183
	DeViSE-style	.021	.033	.022	.032	.143	.136	.131	.116	.170	.198	.170	.178
real logs	CAUSALBRIDGE	**.417**	**.296**	**.195**	**.120**	**.485**	**.385**	**.314**	**.248**	**.455**	**.360**	**.359**	.255
	CLIP-Dissect-e5	.167	.074	.049	.080	.305	.258	.205	.226	.273	.160	.128	.234
	AutoInterp	.083	.037	.049	.020	.265	.200	.178	.152	.364	.240	.205	.170
	Delphi	.000	.037	.073	.120	.163	.212	.215	.228	.182	.200	.179	**.298**
	SASC-style	.000	.111	.024	.060	.145	.223	.138	.184	.091	.240	.205	.234
	DeViSE-style	.083	.074	.024	.000	.216	.188	.178	.084	.273	.200	.179	.064
vehicle CAN	CAUSALBRIDGE	**.800**	**.367**	**.244**	**.218**	**.859**	**.570**	**.412**	**.397**	**.800**	.367	**.500**	**.367**
	CLIP-Dissect-e5	.000	.067	.078	.091	.135	.229	.298	.286	.300	.167	.150	.233
	AutoInterp	.000	.000	.000	.000	.184	.245	.171	.191	.100	.100	.100	.000
	Delphi	.000	.017	.067	.118	.195	.292	.241	.281	.100	.133	.300	.000
	SASC-style	.000	.167	.044	.109	.276	.366	.223	.268	.000	**.433**	.500	.333
	DeViSE-style	.000	.000	.000	.000	.180	.173	.141	.083	.000	.333	.000	.000

variables	method	identification 20	identification 50	identification 75	identification 90	type acc 20	type acc 50	type acc 75	type acc 90	coord acc 20	coord acc 50	coord acc 75	coord acc 90
simulated arms	CAUSALBRIDGE	**.106**	**.203**	**.208**	**.116**	**.396**	**.537**	**.577**	**.420**	**.106**	**.181**	**.159**	**.145**
	CLIP-Dissect-e5	.062	.050	.033	.060	.271	.215	.137	.105	.085	.026	.023	.009
	AutoInterp	.000	.033	.039	.033	.000	.107	.126	.114	.000	.009	.034	.000
	Delphi	.062	.034	.072	.028	.104	.141	.170	.132	.021	.052	.051	.014
	SASC-style	.042	.074	.060	.028	.208	.281	.203	.173	.043	.060	.040	.014
	DeViSE-style	.000	.025	.016	.005	.312	.273	.275	.173	.000	.000	.000	.000
real logs	CAUSALBRIDGE	**.417**	**.296**	**.244**	**.180**	**.667**	**.518**	**.512**	**.460**	**.750**	**.545**	**.606**	**.579**
	CLIP-Dissect-e5	.000	.037	.100	.080	.083	.000	.073	.060	.000	.045	.061	.026
	AutoInterp	.250	.111	.098	.041	.250	.185	.244	.260	.250	.182	.212	.368
	Delphi	.250	.120	.024	.100	.167	.074	.098	.240	.000	.136	.151	.026
	SASC-style	.083	.115	.073	.080	.083	.222	.146	.180	.250	.409	.333	.395
	DeViSE-style	.083	.074	.098	.120	.083	.111	.195	.000	.500	.227	.303	.316
vehicle CAN	CAUSALBRIDGE	**.800**	**.400**	**.300**	**.282**	**.900**	**.683**	**.356**	**.391**	**.800**	**.367**	**.500**	**.367**
	CLIP-Dissect-e5	.000	.050	.089	.118	.000	.050	.033	.027	.000	.000	.000	.000
	AutoInterp	.050	.017	.000	.009	.000	.400	.200	.273	.000	.000	.000	.000
	Delphi	.000	.067	.033	.055	.000	.183	.156	.273	.000	.000	.000	.000
	SASC-style	.000	.250	.146	.191	.200	.383	.067	.300	.000	.200	.000	.333
	DeViSE-style	.050	.067	.022	–	.000	.000	.000	.000	.000	.000	.000	.000

Table 10: Questionnaire items with no structure supplied, CAUSALBRIDGE on the graph it discovers, with sibling discrimination (Appendix H).

variables	method	NRR@1 20	NRR@1 50	NRR@1 75	NRR@1 90	MRR 20	MRR 50	MRR 75	MRR 90	SDA 20	SDA 50	SDA 75	SDA 90	identification 20	identification 50	identification 75	identification 90
questionnaire items	CAUSALBRIDGE	**.121**	**.098**	**.059**	**.028**	**.253**	**.216**	**.164**	**.105**	**.226**	**.189**	**.164**	**.147**	**.356**	**.364**	.283	.217
	CLIP-Dissect-e5	.028	.023	.022	.014	.096	.095	.099	.071	.097	.112	.112	.137	.163	.179	.183	.153
	AutoInterp	.010	.012	.008	.007	.058	.055	.054	.054	.118	.124	.109	.109	.085	.103	.101	.099
	Delphi	.008	.010	.013	.011	.060	.055	.060	.063	.108	.113	.121	.110	.141	.122	.133	.113
	SASC-style	.079	.062	.035	.021	.211	.175	.128	.087	.169	.164	.164	.136	.356	.342	**.303**	**.219**
	DeViSE-style	.033	.023	.014	.008	.131	.110	.087	.064	.154	.162	.155	.133	.238	.227	.193	.164

Table 11: Latent variables, same setting as Table 10, with naming quality. The baselines name the factors that a shared factor analysis extracts. CAUSALBRIDGE names the latent variables its own discovery step finds.

variables	method	NRR@1 20	NRR@1 50	NRR@1 75	NRR@1 90	MRR 20	MRR 50	MRR 75	MRR 90	identification 20	identification 50	identification 75	identification 90	naming quality 20	naming quality 50	naming quality 75	naming quality 90
latent variables	CAUSALBRIDGE	**.540**	**.486**	**.386**	**.240**	**.660**	**.623**	**.560**	**.424**	**.531**	**.511**	**.441**	**.254**	.446	**.426**	**.390**	**.217**
	CLIP-Dissect-e5	.073	.119	.127	.142	.257	.288	.298	.292	.128	.120	.151	.121	.116	.112	.135	.105
	AutoInterp	.096	.100	.108	.088	.258	.259	.270	.252	.160	.140	.170	.139	.160	.140	.170	.139
	Delphi	.123	.108	.085	.069	.266	.269	.251	.235	.138	.108	.100	.115	.138	.108	.100	.115
	SASC-style	.369	.315	.204	.115	.523	.479	.393	.291	.502	.424	.319	.204	**.502**	.424	.319	.204
	DeViSE-style	.138	.138	.108	.112	.291	.281	.260	.266	.151	.154	.123	.112	.062	.065	.069	.054

or a side. Type accuracy scores the first part and coordinate accuracy the second (Appendix H and Figure 18). The type is the part the structure determines most. Our method leads type accuracy on all three robot columns at every masking level. On the simulated arms its type accuracy is 1.3 to 2.4 times that of the strongest baseline, .537 against .281 at 50% masking and .577 against .275 at 75%. On the real logs it is 1.8 to 2.7 times that of the strongest baseline. On the simulated arms its type accuracy rises from .396 to .577 between 20 and 75% masking, so the type stays recoverable as names are masked. The coordinate is the harder part, and on the simulated arms it is where the gap to the baselines opens as names are masked. At 90% masking the coordinate accuracy of our method on the simulated arms is .145 against .014 for the strongest baseline. On the real logs it leads at every level, .579 against .395 at 90%. On the vehicle CAN it leads at every level, .367 against .200 at 50%. Once the names of the siblings are masked, the position in the graph is what still distinguishes joint 3 from joint 5.

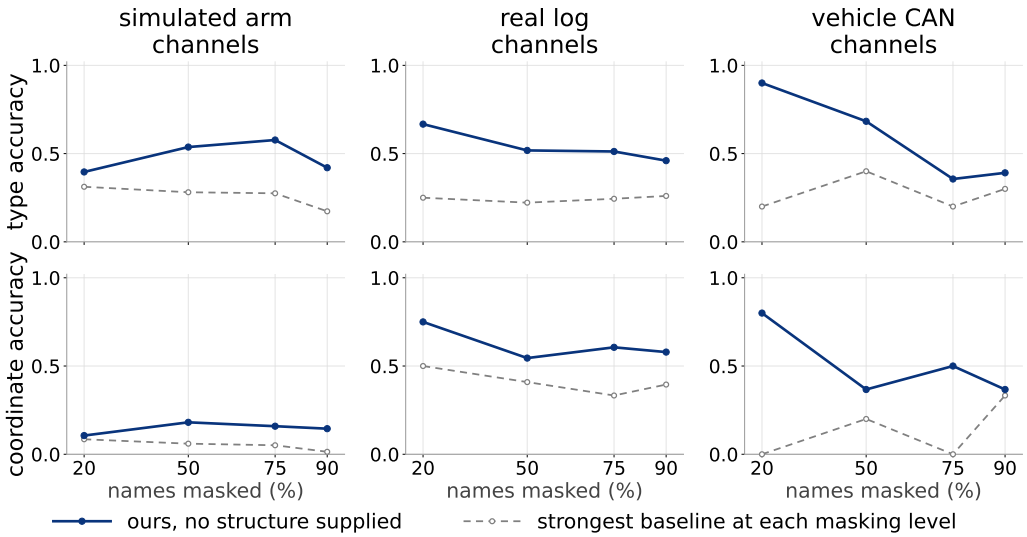

Figure 18: The two parts of a channel name on the three robot columns, with no structure supplied. Top, type accuracy scores what is measured. Bottom, coordinate accuracy scores which one of that type. The solid line is CAUSALBRIDGE and the dashed line the strongest baseline at each masking level.

Through the structure, CAUSALBRIDGE *names the latent variables that no activation profile can name.* A latent variable has no data column and no text of its own. The four methods that read an activation profile therefore stay between .07 and .14 in NRR@1 at every masking level. SASC-style reads the item texts and falls from .369 to .115 as they are masked. Our method uses the relations of the latent variable to its items and falls from .540 to .240. The ratio between the two therefore grows from 1.5 to 1.9 through 75% masking. Naming quality rejects a quoted statement and a copied item text. Under naming quality our method ranks first from 50% masking on, .390 against .319 for SASC-style at 75%. On questionnaire items, a named sibling indicates the topic of the unnamed item. There the ratio of CAUSALBRIDGE to SASC-style in NRR@1 is 1.5, 1.6, 1.7 and 1.3 at the four levels.

Two panels depart from this pattern, and the named siblings are the reason in both. On the simulated arms NRR@1 peaks at 50% masking, since at light masking CAUSALBRIDGE copies the name of a named sibling that differs from the target by one index. A copied sibling name lies nearer to that sibling than to the target. Sibling discrimination on the simulated arms is therefore the one auxiliary metric on which some baselines lie above our method. On questionnaire items the identification curve of SASC-style overlaps that of our method. The judge picks one of five sentences from the same scale, so it credits any name that reaches the right latent variable. Any method that reads the neighboring sentences reaches that latent variable (Table 18). NRR@1 requires the exact item, and there our method leads at every level.

J.2 RQ1: THE DISCOVERED GRAPH AND THE SUPPLIED GRAPH

The second finding of RQ1 is that CAUSALBRIDGE discovers the structure it needs and that its lead is wider when none is supplied. Tables 12 to 14 give every method under the discovered graph and under the supplied graph. They carry the numbers behind the lower row of Figure 4.

Table 12: Robot channels under the supplied graph and under the graph CAUSALBRIDGE discovers, every method given the same graph. A dagger marks a method whose output does not change with the structure it is supplied, and a dash a method that returned no name for any channel of that set at that level.

			NRR@1				MRR				SDA			
structure	variables	method	20	50	75	90	20	50	75	90	20	50	75	90
supplied graph	simulated arms	CAUSALBRIDGE	**.146**	**.198**	**.181**	**.055**	**.317**	**.341**	**.354**	**.225**	.170	.259	**.222**	.117
		CLIP-Dissect-e5†	.083	.058	.027	.027	.185	.186	.116	.120	.234	.155	.142	.169
		AutoInterp†	.000	.008	.011	.005	.078	.095	.095	.086	**.255**	.207	.165	.155
		Delphi†	.000	.025	.016	.005	.079	.120	.119	.083	.043	.241	.193	.197
		SASC-style†	.021	.058	.027	.018	.112	.162	.130	.118	.170	.147	.165	.183
		DeViSE-style†	.021	.033	.022	.032	.143	.136	.131	.116	.170	.198	.170	.178
		base model given graph	.021	.083	.033	.023	.164	.193	.135	.148	**.255**	**.293**	.205	**.211**
		GraphMAE	.000	.000	.000	.000	.162	.134	.086	.082	.128	.147	.097	.174
supplied graph	real logs	CAUSALBRIDGE	**.500**	**.333**	**.244**	**.160**	**.566**	**.451**	**.369**	**.276**	.545	**.440**	**.333**	**.298**
		CLIP-Dissect-e5†	.167	.074	.049	.080	.305	.258	.205	.226	.273	.160	.128	.234
		AutoInterp†	.083	.037	.049	.020	.265	.200	.178	.152	.364	.240	.205	.170
		Delphi†	.000	.037	.073	.120	.163	.212	.215	.228	.182	.200	.179	.298
		SASC-style†	.000	.111	.024	.060	.145	.223	.138	.184	.091	.240	.205	.234
		DeViSE-style†	.083	.074	.024	.000	.216	.188	.178	.084	.273	.200	.179	.064
		base model given graph	.167	.185	.146	.020	.434	.387	.278	.176	**.727**	.440	.333	.213
		GraphMAE	.000	.000	.000	.000	.205	.150	.131	.136	.000	.200	.103	.149
supplied graph	vehicle CAN	CAUSALBRIDGE	**.750**	**.300**	**.156**	**.145**	**.775**	**.553**	**.317**	**.378**	**1.000**	.300	.400	.133
		CLIP-Dissect-e5†	.000	.067	.078	.091	.135	.229	.298	.286	.300	.167	.150	.233
		AutoInterp†	.000	.000	.000	.000	.184	.245	.171	.191	.100	.100	.100	.000
		Delphi†	.000	.017	.067	.118	.195	.292	.241	.281	.100	.133	.300	.000
		SASC-style†	.000	.167	.044	.109	.276	.366	.223	.268	.000	**.433**	**.500**	**.333**
		DeViSE-style†	.000	.000	.000	.000	.180	.173	.141	.083	.000	.333	.000	.000
		base model given graph	.000	.083	.067	.091	.269	.322	.269	.297	.000	.100	**.500**	.000
		GraphMAE	.000	.000	.000	.000	.250	.199	.156	.191	.000	.333	.000	.000
discovered graph	simulated arms	CAUSALBRIDGE	**.083**	**.140**	**.137**	**.046**	**.224**	**.286**	**.278**	**.194**	.128	.190	.153	.164
		CLIP-Dissect-e5†	.083	.058	.027	.027	.185	.186	.116	.120	.234	.155	.142	.169
		AutoInterp†	.000	.008	.011	.005	.078	.095	.095	.086	**.255**	.207	.165	.155
		Delphi†	.000	.025	.016	.005	.079	.120	.119	.083	.043	**.241**	**.193**	.197
		SASC-style†	.021	.058	.027	.018	.112	.162	.130	.118	.170	.147	.165	.183
		DeViSE-style†	.021	.033	.022	.032	.143	.136	.131	.116	.170	.198	.170	.178
		base model given graph	.000	.050	.027	.023	.147	.176	.128	.131	.191	.103	.182	**.202**
		GraphMAE	.000	.000	.000	.000	.154	.097	.082	.080	.149	.198	.148	.160
discovered graph	real logs	CAUSALBRIDGE	**.417**	**.296**	**.195**	**.120**	**.485**	**.385**	**.314**	**.248**	**.455**	**.360**	**.359**	.255
		CLIP-Dissect-e5†	.167	.074	.049	.080	.305	.258	.205	.226	.273	.160	.128	.234
		AutoInterp†	.083	.037	.049	.020	.265	.200	.178	.152	.364	.240	.205	.170
		Delphi†	.000	.037	.073	.120	.163	.212	.215	.228	.182	.200	.179	**.298**
		SASC-style†	.000	.111	.024	.060	.145	.223	.138	.184	.091	.240	.205	.234
		DeViSE-style†	.083	.074	.024	.000	.216	.188	.178	.084	.273	.200	.179	.064
		base model given graph	.000	.148	.122	.020	.297	.334	.284	.181	.455	.320	.282	.128
		GraphMAE	.000	.000	.000	.000	.156	.127	.135	.134	.000	.120	.103	.128
discovered graph	vehicle CAN	CAUSALBRIDGE	**.800**	**.367**	**.244**	**.218**	**.859**	**.570**	**.412**	**.397**	**.800**	.367	**.500**	**.367**
		CLIP-Dissect-e5†	.000	.067	.078	.091	.135	.229	.298	.286	.300	.167	.150	.233
		AutoInterp†	.000	.000	.000	.000	.184	.245	.171	.191	.100	.100	.100	.000
		Delphi†	.000	.017	.067	.118	.195	.292	.241	.281	.100	.133	.300	.000
		SASC-style†	.000	.167	.044	.109	.276	.366	.223	.268	.000	**.433**	.500	.333
		DeViSE-style†	.000	.000	.000	.000	.180	.173	.141	.083	.000	.333	.000	.000
		base model given graph	.100	.050	.022	.073	.343	.279	.236	.265	.000	.100	.500	.000
		GraphMAE	.000	.000	.000	.000	.147	.179	.169	.191	.000	.267	.200	.000

CAUSALBRIDGE *does as well on the graph it discovers as on the supplied graph.* The dashed line of the lower row is our method on its discovered graph, and it follows the solid line in every panel. On the vehicle CAN it is ahead at every masking level. On the simulated arms and the real logs the supplied graph describes the mechanism that produced the measurements, and it is ahead there. On latent variables the two graphs define different targets, and the two lines sit at a similar level.

Table 12: Continued with the same rows, for identification, type accuracy and coordinate accuracy.

			identification				type acc				coord acc			
structure	variables	method	20	50	75	90	20	50	75	90	20	50	75	90
supplied graph	simulated arms	CAUSALBRIDGE	**.167**	**.252**	**.242**	**.106**	**.542**	**.612**	**.703**	**.525**	**.149**	**.233**	**.216**	**.132**
		CLIP-Dissect-e5†	.062	.050	.033	.060	.271	.215	.137	.105	.085	.026	.023	.009
		AutoInterp†	.000	.033	.039	.033	.000	.107	.126	.114	.000	.009	.034	.000
		Delphi†	.062	.034	.072	.028	.104	.141	.170	.132	.021	.052	.051	.014
		SASC-style†	.042	.074	.060	.028	.208	.281	.203	.173	.043	.060	.040	.014
		DeViSE-style†	.000	.025	.016	.005	.312	.273	.275	.173	.000	.000	.000	.000
		base model given graph	.109	.190	.078	.084	.375	.281	.214	.288	.043	.172	.074	.061
		GraphMAE	.064	.076	.041	.048	.375	.273	.115	.101	.021	.060	.040	.085
supplied graph	real logs	CAUSALBRIDGE	**.500**	**.407**	**.268**	**.208**	**.750**	**.593**	**.610**	**.500**	**1.000**	**.682**	.455	**.553**
		CLIP-Dissect-e5†	.000	.037	.100	.080	.083	.000	.073	.060	.000	.045	.061	.026
		AutoInterp†	.250	.111	.098	.041	.250	.185	.244	.260	.250	.182	.212	.368
		Delphi†	.250	.120	.024	.100	.167	.074	.098	.240	.000	.136	.151	.026
		SASC-style†	.083	.115	.073	.080	.083	.222	.146	.180	.250	.409	.333	.395
		DeViSE-style†	.083	.074	.098	.120	.083	.111	.195	.000	.500	.227	.303	.316
		base model given graph	.333	.333	.128	.120	.167	.222	.195	.260	.875	.591	**.485**	.500
		GraphMAE	.083	.000	.000	.061	.583	.222	.220	.380	.250	.500	.242	.447
supplied graph	vehicle CAN	CAUSALBRIDGE	**.750**	**.317**	**.216**	**.200**	**.750**	**.650**	**.289**	**.382**	**1.000**	**.300**	**.400**	.133
		CLIP-Dissect-e5†	.000	.050	.089	.118	.000	.050	.033	.027	.000	.000	.000	.000
		AutoInterp†	.050	.017	.000	.009	.000	.400	.200	.273	.000	.000	.000	.000
		Delphi†	.000	.067	.033	.055	.000	.183	.156	.273	.000	.000	.000	.000
		SASC-style†	.000	.250	.146	.191	.200	.383	.067	.300	.000	.200	.000	**.333**
		DeViSE-style†	.050	.067	.022	–	.000	.000	.000	.000	.000	.000	.000	.000
		base model given graph	.105	.133	.091	.074	.250	.483	.000	.264	.000	.000	.000	.000
		GraphMAE	.000	.017	.000	.009	.500	.000	.222	.273	.000	.000	.000	.000
discovered graph	simulated arms	CAUSALBRIDGE	**.106**	**.203**	**.208**	**.116**	.396	**.537**	**.577**	**.420**	**.106**	**.181**	**.159**	**.145**
		CLIP-Dissect-e5†	.062	.050	.033	.060	.271	.215	.137	.105	.085	.026	.023	.009
		AutoInterp†	.000	.033	.039	.033	.000	.107	.126	.114	.000	.009	.034	.000
		Delphi†	.062	.034	.072	.028	.104	.141	.170	.132	.021	.052	.051	.014
		SASC-style†	.042	.074	.060	.028	.208	.281	.203	.173	.043	.060	.040	.014
		DeViSE-style†	.000	.025	.016	.005	.312	.273	.275	.173	.000	.000	.000	.000
		base model given graph	.021	.076	.096	.089	**.417**	.364	.225	.265	.021	.035	.040	.052
		GraphMAE	.045	.078	.045	.029	.312	.132	.099	.101	.064	.095	.085	.066
discovered graph	real logs	CAUSALBRIDGE	**.417**	.296	**.244**	**.180**	**.667**	**.518**	**.512**	**.460**	**.750**	**.545**	**.606**	**.579**
		CLIP-Dissect-e5†	.000	.037	.100	.080	.083	.000	.073	.060	.000	.045	.061	.026
		AutoInterp†	.250	.111	.098	.041	.250	.185	.244	.260	.250	.182	.212	.368
		Delphi†	.250	.120	.024	.100	.167	.074	.098	.240	.000	.136	.151	.026
		SASC-style†	.083	.115	.073	.080	.083	.222	.146	.180	.250	.409	.333	.395
		DeViSE-style†	.083	.074	.098	.120	.083	.111	.195	.000	.500	.227	.303	.316
		base model given graph	.083	**.370**	.171	.020	.083	.259	.293	.280	.625	.500	.485	.474
		GraphMAE	.000	.000	.122	.040	.250	.259	.220	.380	.250	.227	.303	.421
discovered graph	vehicle CAN	CAUSALBRIDGE	**.800**	**.400**	**.300**	**.282**	**.900**	**.683**	**.356**	**.391**	**.800**	**.367**	**.500**	**.367**
		CLIP-Dissect-e5†	.000	.050	.089	.118	.000	.050	.033	.027	.000	.000	.000	.000
		AutoInterp†	.050	.017	.000	.009	.000	.400	.200	.273	.000	.000	.000	.000
		Delphi†	.000	.067	.033	.055	.000	.183	.156	.273	.000	.000	.000	.000
		SASC-style†	.000	.250	.146	.191	.200	.383	.067	.300	.000	.200	.000	.333
		DeViSE-style†	.050	.067	.022	–	.000	.000	.000	.000	.000	.000	.000	.000
		base model given graph	.050	.133	.023	.135	.050	.367	.000	.009	.000	.000	.000	.000
		GraphMAE	.000	.067	.000	.009	.100	.150	.122	.273	.000	.000	.000	.000

Type accuracy agrees with NRR@1 on all three columns. On the vehicle CAN the discovered graph is ahead at every masking level, .900 and .683 against .750 and .650 at 20 and 50%. The supplied graph is ahead at every level on the simulated arms and on the real logs.

On latent variables the two lines of the lower row are two tasks with different targets. Under the supplied graph the targets are the 52 latent variables of the five keys counted with their parent level. Under the discovered graph they are the 70 latent variables our method discovers, each referred to the key by the rule of Appendix D.2. A discovered latent variable holds 4.8 items on average, counting the items beneath its latent children, against 12.5 for a latent variable of the key. At 75% masking it therefore keeps 1.2 named items against 3.1. Table 20 gives both sources scale by scale.

A supplied graph lifts the baselines that can use it. The five methods that name a variable from its data alone do not change with the structure they are given, which the dagger in Tables 12 and 13 marks. On latent variables the same holds for the two that keep the factor-analysis grouping. The baselines that use the graph gain from it, and the gain depends on what the graph adds to their input.

Table 13: Questionnaire items under both structure sources, every method given the same graph. A dagger marks a method whose output does not change with the structure it is supplied.

		NRR@1				MRR				SDA				identification			
structure	method	20	50	75	90	20	50	75	90	20	50	75	90	20	50	75	90
supplied graph	CAUSALBRIDGE	**.126**	**.097**	**.064**	**.038**	**.254**	**.220**	**.179**	**.121**	**.200**	**.176**	.147	.129	.351	.314	**.304**	.227
	CLIP-Dissect-e5†	.028	.023	.022	.014	.096	.095	.099	.071	.097	.112	.112	**.137**	.163	.179	.183	.153
	AutoInterp†	.010	.012	.008	.007	.058	.055	.054	.054	.118	.124	.109	.109	.085	.103	.101	.099
	Delphi†	.008	.010	.013	.011	.060	.055	.060	.063	.108	.113	.121	.110	.141	.122	.133	.113
	SASC-style†	.079	.062	.035	.021	.211	.175	.128	.087	.169	.164	**.164**	.136	**.356**	**.342**	.303	.219
	DeViSE-style†	.033	.023	.014	.008	.131	.110	.087	.064	.154	.162	.155	.133	.238	.227	.193	.164
	base model given graph	.074	.051	.051	.017	.219	.182	.166	.100	.126	.102	.119	.109	.338	.309	.280	**.228**
	GraphMAE	.000	.000	.000	.000	.086	.080	.075	.059	.077	.080	.075	.096	.144	.139	.118	.108
discovered graph	CAUSALBRIDGE	**.121**	**.098**	**.059**	**.028**	**.253**	**.216**	**.164**	**.105**	**.226**	**.189**	**.164**	**.147**	.356	.364	.283	.217
	CLIP-Dissect-e5†	.028	.023	.022	.014	.096	.095	.099	.071	.097	.112	.112	.137	.163	.179	.183	.153
	AutoInterp†	.010	.012	.008	.007	.058	.055	.054	.054	.118	.124	.109	.109	.085	.103	.101	.099
	Delphi†	.008	.010	.013	.011	.060	.055	.060	.063	.108	.113	.121	.110	.141	.122	.133	.113
	SASC-style†	.079	.062	.035	.021	.211	.175	.128	.087	.169	.164	.164	.136	.356	.342	.303	.219
	DeViSE-style†	.033	.023	.014	.008	.131	.110	.087	.064	.154	.162	.155	.133	.238	.227	.193	.164
	base model given graph	.085	.054	.046	.017	.227	.188	.162	.094	.174	.163	.154	.125	**.364**	**.384**	**.309**	**.229**
	GraphMAE	.000	.000	.000	.000	.069	.067	.062	.054	.103	.103	.096	.109	.159	.135	.123	.115

Table 14: Latent variables under both structure sources, every method given the same graph.

		NRR@1				MRR				identification				naming quality			
structure	method	20	50	75	90	20	50	75	90	20	50	75	90	20	50	75	90
supplied graph	CAUSALBRIDGE	**.485**	.415	**.415**	**.238**	**.624**	**.560**	**.551**	**.400**	**.627**	**.537**	**.535**	**.351**	**.596**	**.518**	**.500**	**.332**
	CLIP-Dissect-e5	.158	.146	.135	.181	.333	.314	.315	.337	.243	.203	.188	.186	.243	.203	.188	.186
	AutoInterp	.108	.108	.112	.096	.262	.267	.275	.260	.139	.202	.135	.142	.139	.202	.135	.142
	Delphi	.088	.123	.135	.088	.250	.287	.283	.257	.169	.174	.192	.173	.169	.174	.192	.173
	SASC-style†	.369	.315	.204	.115	.523	.479	.393	.291	.502	.424	.319	.204	.502	.424	.319	.204
	DeViSE-style†	.138	.138	.108	.112	.291	.281	.260	.266	.151	.154	.123	.112	.062	.065	.069	.054
	base model given graph	.477	**.419**	.331	.212	.614	.560	.480	.376	.512	.492	.454	.285	.512	.492	.450	.277
	GraphMAE	.242	.181	.158	.115	.400	.347	.320	.276	.350	.315	.208	.124	.000	.000	.000	.000
discovered graph	CAUSALBRIDGE	**.540**	**.486**	.386	**.240**	**.660**	**.623**	.560	**.424**	**.531**	**.511**	.441	**.254**	.446	.426	.390	.217
	CLIP-Dissect-e5	.137	.203	.191	.146	.347	.403	.383	.351	.160	.188	.177	.095	.160	.188	.177	.095
	AutoInterp	.131	.126	.163	.126	.334	.335	.367	.330	.154	.151	.157	.143	.154	.151	.157	.143
	Delphi	.177	.137	.174	.126	.372	.344	.366	.345	.191	.157	.200	.166	.183	.157	.197	.166
	SASC-style†	.369	.315	.204	.115	.523	.479	.393	.291	.502	.424	.319	.204	.502	.424	.319	.204
	DeViSE-style†	.138	.138	.108	.112	.291	.281	.260	.266	.151	.154	.123	.112	.062	.065	.069	.054
	base model given graph	.394	.389	**.403**	.186	.576	.568	**.579**	.401	.531	.480	**.446**	.226	**.520**	**.480**	**.437**	**.223**
	GraphMAE	.203	.197	.200	.180	.383	.380	.389	.385	.294	.280	.251	.163	.000	.000	.000	.000

On the real logs, the base model that reads the graph as text reaches .185 and .146 in NRR@1 at 50 and 75% masking. The strongest baseline with no graph reaches .111 and .073. On the simulated arms its identification rate is above every such baseline at all four levels, .190 against .074 at 50%. On both the gain in NRR@1 is gone at 90% masking, where few neighbors keep a name. On questionnaire items its MRR is above every baseline with no graph at all four levels, .219 against .211 at 20%. The gain is small, since an item's named neighbors are sentences that those baselines read too. On latent variables five baselines use the graph. CLIP-Dissect-e5, AutoInterp and Delphi take their grouping from it. CLIP-Dissect-e5 climbs from .073 to .158 in NRR@1 at 20% masking and from .128 to .243 in identification. GraphMAE reaches .242 and the base model .477, against .369 for the strongest baseline with no structure supplied. The base model gains since a latent variable of the key enters the text as its list of named items. GraphMAE never places a true name first on a channel or an item, since it returns a known name there. On the vehicle CAN the base model stays below the strongest baseline with no graph from 50% masking on, .083 against .167 at 50%. The documented edges follow the kinematics, while the recorded speed is computed from the wheel speeds.

Only CAUSALBRIDGE *can discover a graph, so its lead over the strongest baseline is wider with no structure supplied.* With no structure supplied, none of the baselines has a graph to use. A baseline takes from a graph what text can carry. The optimization and the prefix turn the structure into a vector, so our method keeps the lead under both sources. On the vehicle CAN, the lead in NRR@1 at the four masking levels is .800, .200, .167 and .100 with no structure supplied. Under the supplied graph it is .750, .133, .078 and .027. From 50% masking on the ratio is 1.8 to 3.1 against 1.2 to 2.0. On the real logs the lead is 2.7 times at 50 and 75% masking against 1.8 and 1.7 times. The same

holds on latent variables at every masking level, where the targets of our method are the two sets described above. There the difference between the two rows is largest through 50% masking, .163 and .175 against .098 and .072 at 75 and 90%. Many anchors let the graph constrain an unnamed variable tightly, and the difference closes where few names are left in either row. On the vehicle CAN the difference stays between .050 and .088 at every level.

A physical law supplied as a prior raises NRR@1 at 20 and 50% masking on the simulated arms and on the vehicle CAN. Table 15 compares the full method on the graphs discovered with the prior of Appendix E and on the graphs discovered from the time series alone. On the simulated arms the prior raises NRR@1 from .083 to .104 at 20% masking and from .140 to .157 at 50%. On the vehicle CAN it raises NRR@1 from .800 to .950 at 20% and from .367 to .483 at 50%.

Table 15: The main method on the graph discovered from the time series alone, the setting of every other table, against the same method when the discovery step also admits the instantaneous edges that the physics of the platform orients, on an arm from the joints to the end-effector pose and the gripper, and on a drive from the steering wheel angle and the wheel speeds to the vehicle speed and from those to the accelerations and the rates. No structure is supplied in either row. Columns are the four masking levels.

		NRR@1				MRR				identification				type acc				coord acc			
variables	discovery	20	50	75	90	20	50	75	90	20	50	75	90	20	50	75	90	20	50	75	90
simulated arms	time only	.083	.140	.137	.046	.224	.286	.278	.194	.106	.203	.208	.116	.396	.537	.577	.420	.106	.181	.159	.145
	with the physical law	.104	.157	.132	.068	.286	.316	.299	.213	.167	.229	.187	.144	.583	.587	.654	.466	.106	.198	.153	.174
vehicle CAN	time only	.800	.367	.244	.218	.859	.570	.412	.397	.800	.400	.300	.282	.900	.683	.356	.391	.800	.367	.500	.367
	with the physical law	.950	.483	.222	.209	.955	.632	.379	.394	.950	.500	.278	.282	.950	.800	.333	.427	1.000	.333	.500	.200

J.3 RQ1: THE LEAD REPLICATE BY REPLICATE, AGAINST COPIED NAMES AND SCALE BY SCALE

Three checks test whether the lead of RQ1 depends on a few replicates, on the way a name is scored or on the pooling.

Paired differences to the strongest baseline. *The lead holds on the same replicates.* Tables 16 and 17 put every cell of the main comparison on the same replicates as its strongest baseline and report the mean paired difference with its standard error over replicates. The replicates are folds on the questionnaire columns and arms, logs or drives on the robot columns. The strongest baseline of a cell is the baseline with the highest pooled value in that cell of Tables 9 to 11, or of Tables 12 to 14 under the supplied graph. Without a supplied structure, CAUSALBRIDGE is ahead in 58 of the 64 cells and ahead by more than two standard errors in 38. It is behind by more than two standard errors in no cell. The lead lies in the retrieval metrics NRR@1 and MRR, where a name has to reach the right variable. There the difference is positive in 38 of the 40 cells and exceeds two standard errors in 29. On the judged metrics, our method is ahead by more than two standard errors in 9 of the 24 cells and level with the strongest baseline within two standard errors in 14. The judge credits any name that reaches the right latent variable or channel type. Under the supplied graph the three counts are 60, 33 and 1 of 64. The one cell behind by more than two standard errors is item identification at 50% masking. There SASC-style reads the neighboring sentences, and the judge credits the latent variable they name. The cell for real-log identification at 90% rests on a single replicate and has no standard error. A count below the number of replicates means that the baseline returned no name on the other replicates.

Name-copy references. *A copied name reaches at most the right latent variable or channel type.* Tables 18 and 19 score the three references of Appendix I. Where the output is a known name, on every robot column and for Correlation kNN everywhere, NRR@1 never places the true name first. Identification still credits such an output, since the judge often confuses a returned neighbor of the same type with the target among five candidates. This is why the task is scored on NRR@1 and naming quality beside identification. On questionnaire items, where Feature Propagation decodes to the concept bank, it reaches at most .067 on NRR@1. On latent variables, the two references that run behind the factor-analysis front end approach CAUSALBRIDGE in NRR@1. In the scoring space, the text of an item lies near the name of its own latent variable. EFA loading naming has .500 against .540 at 20% masking and .254 against .240 at 90%. Naming quality separates both from our

Table 16: Paired differences to the strongest baseline, no structure supplied. Each cell is our value minus the strongest baseline's, as the mean over replicates, its standard error, the number of replicates in which our method is ahead, and the baseline, C CLIP-Dissect-e5, A AutoInterp, D Delphi, S SASC-style, V DeViSE-style. Bold is a difference above two standard errors.

variables	metric	20	50	75	90
simulated arms	NRR@1	−.019 ± .067 (2/6) C	+.079 ± .040 (4/6) S	**+.109 ± .022** (6/6) S	+.014 ± .017 (3/6) V
	MRR	+.026 ± .058 (5/6) C	**+.097 ± .030** (6/6) C	**+.145 ± .026** (6/6) V	**+.069 ± .017** (6/6) C
	identification	+.042 ± .024 (3/6) D	**+.130 ± .022** (6/6) S	**+.135 ± .031** (6/6) D	**+.054 ± .023** (4/6) C
real logs	NRR@1	+.167 ± .167 (1/3) C	+.238 ± .133 (3/3) S	**+.132 ± .034** (3/3) D	−.002 ± .046 (1/3) D
	MRR	+.090 ± .182 (1/3) C	+.177 ± .090 (3/3) C	**+.104 ± .008** (3/3) D	+.021 ± .062 (2/3) D
	identification	+.111 ± .111 (1/3) D	+.234 ± .135 (3/3) D	+.163 ± .121 (2/3) C	+.000 ± – (0/1) V
vehicle CAN	NRR@1	**+.800 ± .111** (9/10) S	**+.200 ± .033** (9/10) S	**+.167 ± .025** (10/10) C	**+.100 ± .042** (5/10) D
	MRR	**+.583 ± .089** (10/10) S	**+.204 ± .036** (10/10) S	**+.114 ± .024** (10/10) C	**+.111 ± .041** (7/10) C
	identification	**+.750 ± .112** (9/10) V	**+.150 ± .046** (6/10) S	**+.154 ± .067** (7/10) S	**+.091 ± .038** (6/10) S
questionnaire items	NRR@1	**+.041 ± .016** (4/5) S	**+.037 ± .009** (5/5) S	**+.025 ± .010** (5/5) S	+.008 ± .005 (3/5) S
	MRR	**+.041 ± .019** (4/5) S	**+.041 ± .005** (5/5) S	**+.037 ± .009** (5/5) S	**+.018 ± .003** (5/5) S
	identification	+.001 ± .034 (2/5) S	+.022 ± .011 (4/5) S	−.020 ± .017 (1/5) S	−.003 ± .007 (2/5) S
latent variables	NRR@1	**+.171 ± .025** (5/5) S	**+.170 ± .019** (5/5) S	**+.182 ± .028** (5/5) S	**+.098 ± .040** (4/5) C
	MRR	**+.137 ± .018** (5/5) S	**+.144 ± .016** (5/5) S	**+.168 ± .024** (5/5) S	**+.132 ± .029** (5/5) C
	identification	+.029 ± .020 (4/5) S	**+.087 ± .018** (5/5) S	**+.122 ± .040** (4/5) S	+.050 ± .039 (4/5) S
	naming quality	−.056 ± .031 (1/5) S	+.002 ± .012 (2/5) S	+.070 ± .039 (4/5) S	+.013 ± .035 (3/5) S

Table 17: Paired differences to the strongest baseline under the supplied graph, which every method that accepts a graph as input receives, with the structure-free rows as reference. Each cell is our value minus the strongest baseline's, as the mean over replicates, its standard error, the number of replicates in which our method is ahead, and the baseline, C CLIP-Dissect-e5, A AutoInterp, D Delphi, S SASC-style, V DeViSE-style, B base model given graph, G GraphMAE. Bold is a difference above two standard errors.

variables	metric	20	50	75	90
simulated arms	NRR@1	+.037 ± .084 (3/6) C	**+.121 ± .033** (6/6) B	**+.149 ± .016** (6/6) B	+.027 ± .021 (3/6) V
	MRR	+.107 ± .093 (5/6) C	**+.151 ± .029** (6/6) B	**+.220 ± .020** (6/6) B	**+.083 ± .031** (6/6) B
	identification	+.041 ± .055 (3/6) B	+.058 ± .038 (4/6) B	**+.164 ± .024** (6/6) B	+.022 ± .019 (3/6) B
real logs	NRR@1	+.278 ± .147 (2/3) B	+.190 ± .156 (2/3) B	**+.098 ± .002** (3/3) B	+.041 ± .024 (2/3) D
	MRR	+.115 ± .133 (2/3) B	+.098 ± .133 (2/3) B	**+.095 ± .007** (3/3) B	**+.057 ± .018** (3/3) D
	identification	+.167 ± .096 (2/3) B	+.032 ± .159 (2/3) B	**+.143 ± .030** (3/3) B	+.115 ± .071 (3/3) B
vehicle CAN	NRR@1	**+.750 ± .083** (10/10) B	**+.133 ± .042** (8/10) S	+.078 ± .044 (6/10) C	+.027 ± .030 (4/10) D
	MRR	**+.499 ± .078** (10/10) S	**+.188 ± .033** (9/10) S	+.019 ± .042 (5/10) C	**+.080 ± .018** (10/10) B
	identification	**+.650 ± .130** (8/10) B	+.067 ± .057 (4/10) S	+.069 ± .065 (5/10) S	+.009 ± .025 (4/10) S
questionnaire items	NRR@1	**+.046 ± .014** (5/5) S	**+.036 ± .014** (4/5) S	**+.013 ± .006** (3/5) B	**+.017 ± .007** (5/5) S
	MRR	**+.035 ± .013** (5/5) B	**+.038 ± .012** (4/5) B	**+.012 ± .005** (5/5) B	**+.021 ± .008** (5/5) B
	identification	−.005 ± .019 (2/5) S	**−.028 ± .009** (0/5) S	+.001 ± .011 (2/5) S	−.001 ± .018 (2/5) B
latent variables	NRR@1	+.008 ± .019 (2/5) B	−.004 ± .034 (1/5) B	**+.085 ± .013** (5/5) B	**+.027 ± .005** (5/5) B
	MRR	+.009 ± .016 (2/5) B	+.000 ± .028 (2/5) B	**+.071 ± .009** (5/5) B	**+.025 ± .003** (5/5) B
	identification	**+.115 ± .029** (5/5) B	+.045 ± .026 (4/5) B	**+.081 ± .029** (4/5) B	**+.067 ± .018** (5/5) B
	naming quality	**+.085 ± .023** (5/5) B	+.025 ± .022 (4/5) B	+.050 ± .028 (4/5) B	**+.055 ± .013** (5/5) B

method. EFA loading naming returns an item of the scale, and the verbatim check rejects a copied item text in every case. Its identification rate of .608 at 20% masking therefore becomes a naming quality of .000. Feature Propagation decodes the mean of a factor's items to the concept bank. Its naming quality lies below that of our method at every masking level, .408 against .446 at 20%.

Scale by scale, arm by arm, log by log. *The lead holds scale by scale.* In NRR@1 on items CAUSALBRIDGE is first in 15 of the 20 combinations of scale and masking level. The main text pools the five evaluation scales by sample and the arms, logs or drives of a robot column by sample. Tables 21 to 25 give every method on the items and the latent variables of each scale. They report the three metrics of Table 8 at every masking level and naming quality on latent variables. Table 26 gives identification on each simulated arm and real log. Pooled by sample, the cells of the five scale tables reproduce the main tables. The scales differ in how much information their items give a frozen model. HEXACO has short everyday statements, so it is the easiest scale for identification. Table 20 gives our method on each scale under both structure sources. The key groups the items by the theory of the scale, and the measurements group them by how respondents answer. The two

Table 18: Name-copy references with no structure supplied. Correlation kNN returns the name of the most correlated named variable, and Feature Propagation decodes the diffused known names to the nearest known name on robot channels and to the concept bank on questionnaires.

		NRR@1				MRR				identification			
variables	method	20	50	75	90	20	50	75	90	20	50	75	90
simulated arms	Correlation kNN	.000	.000	.000	.000	.181	.173	.159	.117	.106	.067	.068	.051
	Feature Propagation	.000	.000	.000	.000	.189	.157	.168	.118	.128	.067	.074	.052
real logs	Correlation kNN	.000	.000	.000	.000	.262	.218	.179	.142	.083	.222	.098	.040
	Feature Propagation	.000	.000	.000	.000	.210	.209	.174	.136	.167	.259	.122	.020
vehicle CAN	Correlation kNN	.000	.000	.000	.000	.228	.251	.181	.191	.000	.083	.012	.000
	Feature Propagation	.000	.000	.000	.000	.174	.248	.159	.191	.000	.067	.023	.000
questionnaire items	Correlation kNN	.000	.000	.000	.000	.261	.227	.169	.115	.433	.390	.325	.263
	Feature Propagation	.067	.045	.031	.013	.222	.199	.159	.103	.272	.281	.244	.174

Table 19: The two references on latent variables. EFA loading naming returns the text of the item that loads highest on the factor, and Feature Propagation decodes the mean member embedding of the factor to the concept bank.

	NRR@1				MRR				identification				naming quality			
method	20	50	75	90	20	50	75	90	20	50	75	90	20	50	75	90
EFA loading naming	.500	.458	.381	.254	.643	.611	.543	.431	.608	.600	.550	.402	.000	.000	.000	.000
Feature Propagation + EFA	.454	.400	.358	.235	.616	.570	.519	.413	.631	.585	.519	.342	.408	.373	.308	.196

sources therefore define different sets of latent variables on every scale. On the World Values Survey, the discovered latent variables differ most from the key (Appendix E).

Table 20: Latent variables scale by scale, our method under both structure sources.

		NRR@1				MRR				identification				naming quality			
structure	scale	20	50	75	90	20	50	75	90	20	50	75	90	20	50	75	90
discovered graph	DASS	.756	.800	.622	.400	.859	.881	.789	.648	.689	.689	.511	.222	.667	.622	.511	.156
	HEXACO	.408	.320	.304	.104	.516	.462	.461	.247	.512	.496	.504	.248	.384	.432	.456	.192
	KIMS	.691	.655	.418	.400	.814	.803	.633	.605	.382	.382	.273	.164	.382	.345	.255	.164
	RIASEC	.692	.585	.508	.262	.793	.726	.655	.473	.800	.662	.562	.369	.615	.446	.406	.323
	WVS	.350	.333	.217	.233	.524	.488	.427	.409	.300	.367	.283	.250	.283	.317	.267	.250
supplied graph	DASS	.733	.667	.733	.533	.867	.822	.867	.733	.667	.467	.467	.200	.667	.467	.467	.200
	HEXACO	.387	.300	.373	.140	.508	.442	.495	.298	.560	.500	.587	.407	.547	.500	.587	.393
	KIMS	.600	.750	.450	.550	.792	.858	.671	.721	.500	.500	.450	.300	.500	.500	.450	.300
	RIASEC	.667	.633	.600	.400	.822	.786	.750	.573	.933	.828	.633	.433	.867	.690	.367	.367
	WVS	.556	.422	.311	.222	.720	.584	.450	.374	.689	.512	.356	.182	.600	.488	.333	.159

Table 21: DASS, no structure supplied. Items above and latent variables below. Bold is the best value per column and underline the second best.

	NRR@1				MRR				identification			
items	20	50	75	90	20	50	75	90	20	50	75	90
CAUSALBRIDGE	**.071**	**.067**	**.056**	**.021**	.192	**.184**	**.173**	.101	.286	**.219**	**.176**	.105
CLIP-Dissect-e5	.000	.010	.013	.011	.078	.093	.115	.094	.029	.107	.088	.079
AutoInterp	.024	.019	.019	.016	.103	.095	.103	**.104**	.048	.105	.113	**.111**
Delphi	.000	.029	.013	.005	.092	.093	.093	.089	.143	.086	.157	.074
SASC-style	.071	.048	.025	.016	**.218**	.162	.130	.091	**.310**	.143	.125	.053
DeViSE-style	.000	.019	.006	.011	.129	.133	.088	.085	.048	.143	.056	.095

	NRR@1				MRR				identification				naming quality			
latent variables	20	50	75	90	20	50	75	90	20	50	75	90	20	50	75	90
CAUSALBRIDGE	.756	**.800**	**.622**	**.400**	.859	**.881**	**.789**	.648	**.689**	**.689**	**.511**	.222	**.667**	.622	**.511**	.156
CLIP-Dissect-e5	.267	.467	.267	.400	.611	.678	.567	**.667**	.200	.067	.133	.067	.133	.067	.067	.067
AutoInterp	.333	.400	.400	.200	.611	.644	.644	.544	.267	.267	.333	.133	.267	.267	.333	.133
Delphi	.333	.267	.067	.267	.622	.578	.456	.578	.000	.133	.067	.133	.000	.133	.067	.133
SASC-style	**.933**	.733	.267	.333	**.967**	.856	.556	.611	.600	.667	.133	.200	.600	**.667**	.133	**.200**
DeViSE-style	.467	.533	.200	.400	.678	.722	.533	**.667**	.267	.267	.200	**.267**	.000	.067	.067	.000

Table 22: HEXACO, no structure supplied. Items above and latent variables below. Bold is the best value per column and underline the second best.

	NRR@1				MRR				identification			
items	20	50	75	90	20	50	75	90	20	50	75	90
CAUSALBRIDGE	**.138**	**.113**	**.059**	**.026**	**.249**	**.209**	**.141**	**.080**	**.442**	**.438**	.348	.279
CLIP-Dissect-e5	.025	.022	.017	.013	.068	.074	.068	.051	.204	.209	.204	.190
AutoInterp	.004	.005	.004	.000	.027	.027	.027	.020	.083	.114	.103	.108
Delphi	.000	.003	.004	.004	.023	.026	.027	.030	.167	.149	.146	.133
SASC-style	.087	.060	.039	.016	.199	.160	.117	.063	.421	.438	**.402**	**.290**
DeViSE-style	.033	.022	.014	.003	.104	.082	.072	.043	.312	.290	.259	.209

	NRR@1				MRR				identification				naming quality			
latent variables	20	50	75	90	20	50	75	90	20	50	75	90	20	50	75	90
CAUSALBRIDGE	**.408**	**.320**	**.304**	**.104**	**.516**	**.462**	**.461**	**.247**	**.512**	**.496**	**.504**	**.248**	.384	**.432**	**.456**	.192
CLIP-Dissect-e5	.027	.073	.093	.087	.157	.190	.221	.184	.160	.140	.187	.127	.153	.127	.167	.107
AutoInterp	.053	.027	.027	.053	.158	.128	.140	.154	.133	.127	.147	.160	.133	.127	.147	.160
Delphi	.020	.067	.027	.033	.110	.166	.143	.126	.113	.107	.081	.133	.113	.107	.081	.133
SASC-style	.267	.247	.160	.067	.413	.391	.322	.196	.460	.413	.360	.220	**.460**	.413	.360	**.220**
DeViSE-style	.067	.053	.047	.033	.165	.143	.145	.134	.173	.160	.133	.107	.047	.040	.060	.047

Table 23: KIMS, no structure supplied. Items above and latent variables below. Bold is the best value per column and underline the second best.

	NRR@1				MRR				identification			
items	20	50	75	90	20	50	75	90	20	50	75	90
CAUSALBRIDGE	**.179**	.050	**.069**	**.074**	**.370**	.237	**.236**	**.198**	.179	**.310**	**.214**	.143
CLIP-Dissect-e5	.026	.010	.041	.017	.147	.127	.165	.118	.184	.140	.148	.093
AutoInterp	.026	.010	.014	.011	.096	.092	.099	.100	.077	.051	.083	.069
Delphi	.026	.040	.041	.034	.117	.126	.136	.124	.154	.090	.083	.069
SASC-style	.077	**.120**	.055	.057	.273	**.270**	.203	.175	**.333**	.242	.193	**.190**
DeViSE-style	.051	.060	.034	.029	.170	.174	.132	.114	.231	.121	.131	.137

	NRR@1				MRR				identification				naming quality			
latent variables	20	50	75	90	20	50	75	90	20	50	75	90	20	50	75	90
CAUSALBRIDGE	**.691**	**.655**	.418	**.400**	**.814**	**.803**	.633	**.605**	.382	**.382**	.273	**.164**	.382	.345	.255	**.164**
CLIP-Dissect-e5	.300	.250	.250	.350	.562	.517	.525	.588	.050	.050	.050	.150	.000	.050	.050	.150
AutoInterp	.100	.350	.200	.200	.421	.588	.487	.467	.211	.158	.158	.050	.211	.158	.158	.050
Delphi	.500	.300	.300	.150	.671	.537	.550	.487	.250	.050	.150	.150	.250	.050	.150	.150
SASC-style	.300	.500	**.500**	.200	.575	.688	**.700**	.521	**.500**	.350	**.350**	.150	**.500**	**.350**	**.350**	.150
DeViSE-style	.250	.400	.350	.400	.558	.633	.592	.604	.150	.150	.050	.050	.150	.150	.050	.000

Table 24: RIASEC, no structure supplied. Items above and latent variables below. Bold is the best value per column and underline the second best.

	NRR@1				MRR				identification			
items	20	50	75	90	20	50	75	90	20	50	75	90
CAUSALBRIDGE	**.062**	.042	**.050**	.009	**.220**	**.198**	**.194**	**.124**	**.271**	**.258**	**.200**	**.116**
CLIP-Dissect-e5	.042	**.050**	.028	.014	.148	.153	.147	.074	.083	.167	.194	.079
AutoInterp	.021	.025	.011	**.023**	.108	.085	.075	.102	.125	.117	.133	.088
Delphi	.021	.008	.022	**.023**	.122	.085	.089	.105	.042	.100	.106	.102
SASC-style	.062	**.050**	.022	.014	.201	.179	.119	.102	.167	.217	.172	.112
DeViSE-style	.021	.008	.006	.009	.180	.150	.108	.079	.125	.158	.100	.079

	NRR@1				MRR				identification				naming quality			
latent variables	20	50	75	90	20	50	75	90	20	50	75	90	20	50	75	90
CAUSALBRIDGE	**.692**	**.585**	**.508**	**.262**	**.793**	**.726**	**.655**	**.473**	.800	.662	**.562**	**.369**	.615	.446	**.406**	**.323**
CLIP-Dissect-e5	.100	.133	.200	.233	.362	.377	.416	.453	.069	.100	.167	.167	.069	.100	.167	.133
AutoInterp	.167	.133	.233	.133	.392	.401	.474	.383	.233	.167	.333	.133	.233	.167	.333	.133
Delphi	.200	.067	.233	.100	.442	.350	.446	.354	.267	.067	.200	.033	.267	.067	.200	.033
SASC-style	.633	.433	.233	.133	.776	.628	.487	.387	**.867**	**.667**	.267	.200	**.867**	**.667**	.267	.200
DeViSE-style	.233	.200	.200	.167	.438	.423	.412	.422	.172	.200	.100	.167	.172	.133	.100	.133

J.4 RQ2: COMPONENT BY COMPONENT

The finding of RQ2 is that every component contributes and that the prefix contributes the most. Figure 19 (top) shows the variants with one component removed at a time under the discovered graph.

Table 25: WVS, no structure supplied. Items above and latent variables below. Bold is the best value per column and underline the second best.

	NRR@1				MRR				identification			
items	20	50	75	90	20	50	75	90	20	50	75	90
CAUSALBRIDGE	.048	**.220**	**.075**	.032	**.269**	**.372**	**.210**	**.184**	.048	**.140**	.075	.095
CLIP-Dissect-e5	**.095**	.020	.050	.032	.234	.140	.188	.165	.048	.060	**.138**	**.129**
AutoInterp	.000	.060	.013	.032	.139	.171	.127	.144	.095	.040	.025	.063
Delphi	.048	.000	.037	**.042**	.165	.116	.162	.171	.048	.000	.087	.063
SASC-style	.048	.020	.000	.032	.246	.181	.127	.164	**.190**	.100	.037	.042
DeViSE-style	**.095**	.000	.000	.021	.254	.174	.124	.131	.048	.020	.050	.032

	NRR@1				MRR				identification				naming quality			
latent variables	20	50	75	90	20	50	75	90	20	50	75	90	20	50	75	90
CAUSALBRIDGE	.350	**.333**	**.217**	**.233**	.524	**.488**	**.427**	**.409**	.300	**.367**	**.283**	**.250**	.283	**.317**	**.267**	**.250**
CLIP-Dissect-e5	.044	.089	.089	.089	.270	.327	.288	.291	.068	.114	.068	.071	.068	.114	.068	.071
AutoInterp	.111	.111	.156	.089	.310	.328	.343	.295	.143	.116	.089	.114	.143	.116	.089	.114
Delphi	.178	.133	.089	.067	.368	.335	.281	.290	.133	.156	.089	.089	.133	.156	.089	.089
SASC-style	**.378**	.244	.178	.156	**.550**	.451	.377	.332	**.350**	.238	.267	.178	**.350**	.238	.267	.178
DeViSE-style	.156	.133	.111	.111	.369	.345	.302	.316	.022	.067	.111	.067	.022	.067	.089	.067

Table 26: Identification arm by arm and log by log, our method under both structure sources, with the number of judged cases in parentheses. Vehicle drives are pooled in Table 12.

	discovered graph				supplied graph			
arm or log	20	50	75	90	20	50	75	90
Panda	.167 (6)	.250 (16)	.227 (22)	.138 (29)	.000 (6)	.188 (16)	.208 (24)	.207 (29)
Sawyer	.143 (7)	.167 (18)	.154 (26)	.125 (32)	.286 (7)	.278 (18)	.333 (27)	.161 (31)
IIWA	.000 (9)	.238 (21)	.152 (33)	.125 (40)	.222 (9)	.300 (20)	.242 (33)	.075 (40)
UR5e	.125 (8)	.190 (21)	.258 (31)	.079 (38)	.125 (8)	.190 (21)	.188 (32)	.079 (38)
Kinova3	.111 (9)	.227 (22)	.242 (33)	.128 (39)	.111 (9)	.273 (22)	.182 (33)	.075 (40)
xArm7	.125 (8)	.150 (20)	.212 (33)	.105 (38)	.222 (9)	.273 (22)	.303 (33)	.075 (40)
UTokyo xArm	.000 (3)	.667 (6)	.400 (10)	.250 (12)	.333 (3)	.833 (6)	.400 (10)	.333 (12)
Stanford HYDRA	.333 (3)	.286 (7)	.100 (10)	.231 (13)	.000 (3)	.000 (7)	.100 (10)	.167 (12)
ALOHA cup	.667 (6)	.143 (14)	.238 (21)	.120 (25)	.833 (6)	.429 (14)	.286 (21)	.167 (24)

The prefix carries the structure. Removing the prefix costs the most. Without it, the NRR@1 curve falls below the curve without the evidence line on the vehicle CAN at every level and on every robot column from 75% masking on. The variant without the prefix still receives the vector $\hat{e}_v$ in words, since its evidence line keeps the nearest phrases decoded from that vector (Appendix G). These phrases keep the rough meaning of the vector and lose how the variable relates to each named variable, so a few decoded phrases cannot replace the vector. On the vehicle CAN removing the prefix loses 50 to 88% of the NRR@1 of the full method. On the simulated arms and the real logs it loses the most at deep masking, 68% at 75% on the arms and 83% at 90% on the logs. On questionnaire items it loses 25 to 50% of the NRR@1 of the full method, while identification does not fall. A recognized concept therefore no longer reaches the right variable.

The evidence line supplies the vocabulary of the name. On the vehicle CAN the variant without the evidence line lies between the full method and the variant without the prefix at every masking level. On the identification of items it is the lowest curve, 30 to 38% below the full method at every masking level. The model phrases the vector $\hat{e}_v$ better when the evidence line also gives the named variables most correlated with the unnamed variable. On the simulated arms and the real logs, removing the evidence line costs more than removing the prefix through 50% masking. On the real logs at 20% masking, the variant without the evidence line reaches .167 and the variant without the prefix .333, with the full method at .417. The evidence line lists the named channels most correlated with an unnamed channel, and their names carry the platform's own vocabulary. From 75% on removing the prefix costs more.

The prefix and the evidence line converge on latent variables and at 90% masking. On latent variables the prefix and the evidence line both describe the variable through the same items. The four identification curves therefore lie within .07 of one another at every level, and the four naming quality values within .10 (Table 29). The full method leads NRR@1 and MRR at 20 and 50% masking, and the NRR@1 curves cross at 75%. At 90% masking the variant without the evidence line meets

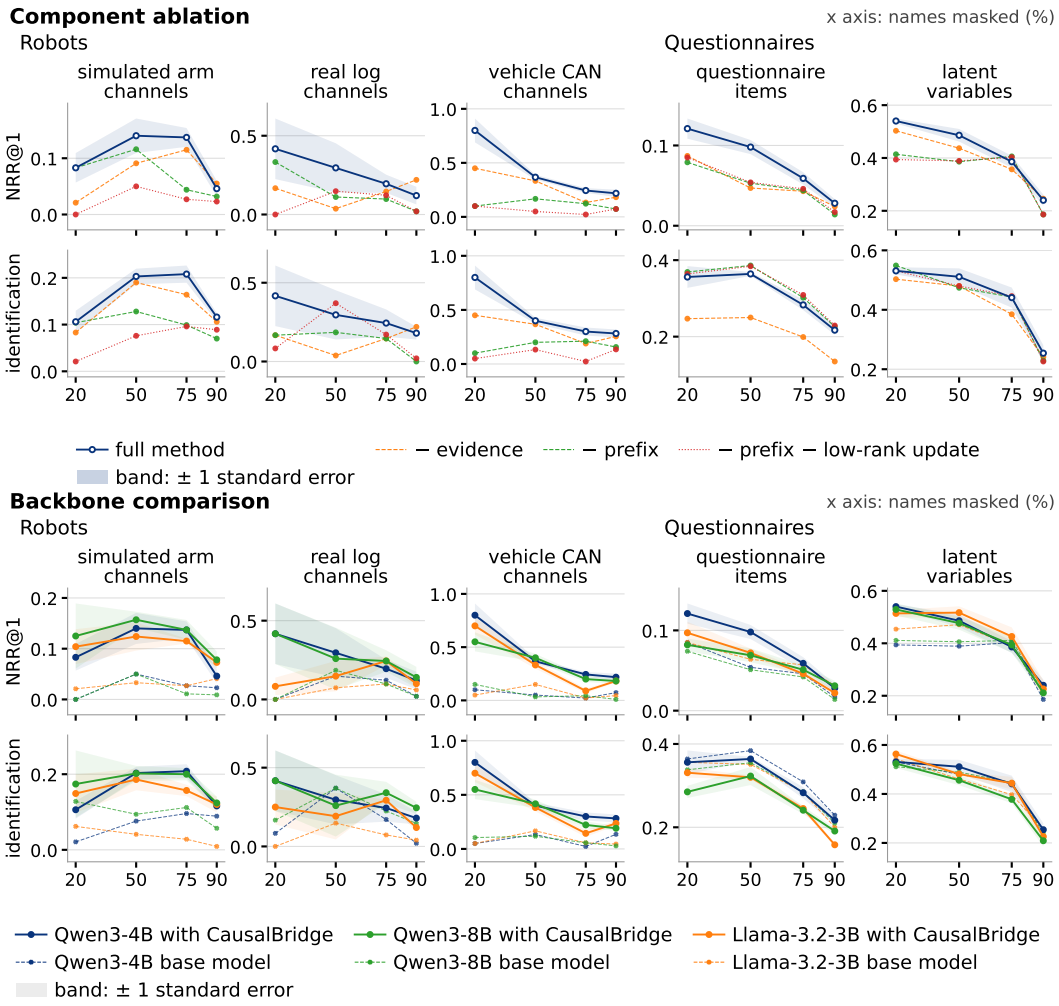

Figure 19: One component removed at a time (top) and one recipe on three backbones (bottom), under the discovered graph. Rows are NRR@1 and identification. In the bottom block, a solid line is a backbone with CAUSALBRIDGE and the dashed line of the same color is that backbone alone.

or passes the full method on the simulated arms, the real logs and the latent variables as well. The evidence line lists the most correlated named variables, and at that level almost none are left to list.

The prefix determines the type of a channel on every robot column, and the evidence line helps determine its coordinate. Type and coordinate accuracy (Appendix H) separate the contributions of the prefix and the evidence line on all three robot columns (Table 27). The type depends on the prefix everywhere. At 50% masking, type accuracy falls on every column without the prefix. It falls from .537 to .347 on the simulated arms, from .518 to .259 on the real logs and from .683 to .300 on the vehicle CAN. The vector $\hat{e}_v$ therefore carries the type of a channel. Without the evidence line the type accuracy stays within .07 of the full method on the simulated arms and the vehicle CAN. On the real logs it falls at 20 and 50% masking, .370 against .518 at 50%, since the names in the evidence line carry the platform's own vocabulary. The coordinate depends on the evidence line where the names of the most correlated named channels carry it. On the vehicle CAN the four wheel speeds differ only by a side word. Without the evidence line, coordinate accuracy falls from .800 to .000 at 20% masking and from .500 to .050 at 75%. At 50 and 90% it stays within .04. On the real logs it falls at every level, from .750 to .625 at 20% masking and from .579 to .474 at 90%. On the simulated arms the coordinate depends on the prefix from 50% masking on, .181 against .078 at 50% and .145 against .042 at 90% without it. The index of a joint follows from its position among the other joints, and no neighbor's name states it. The evidence line is worth .04 to .05 there through 75%.

The low-rank update adds little on its own. Without the prefix, the model with the low-rank update coincides with the base model alone on both questionnaire columns. It keeps part of the gain on the simulated arms, on the real logs at 20% masking and on the vehicle CAN at 50 and 75%.

Under a supplied graph, the prefix is worth less where the structure can be written into the prompt. A latent variable of the key can be written into the text as its item list, while a discovered latent variable exists only as a position in the embedding space. On latent variables the prefix is therefore worth .13 and .10 in NRR@1 at 20 and 50% masking under the discovered graph and almost nothing under the supplied graph (Table 29). On the simulated arms at light masking the text alone even does better, since under the supplied graph the prompt lists a joint's named neighbor with its index.

J.5 RQ3: ACROSS BACKBONES

The finding of RQ3 is that one recipe carries across backbones. Tables 30 to 32 carry the numbers of Figure 19 (bottom), which shows one recipe on Qwen3-4B, Qwen3-8B and Llama-3.2-3B, the order in which every triple of numbers below is listed.

One recipe serves every backbone. With CAUSALBRIDGE, every backbone lies far above itself alone in NRR@1 on the channels, whose labels give a language prior little semantic information. On the vehicle CAN at 20% masking the three backbones rise from .100, .150 and .050 alone to .800, .550 and .700 with our method. On the real logs they rise from .000 alone to .417, .417 and .083. Type accuracy shows the same gain. Alone, the backbones name the type of .050, .450 and .000 of the unnamed channels of the vehicle CAN at 20% masking, and with our method .900, .950 and 1.000. From 50% masking on, the type accuracy of each backbone with our method lies above that backbone alone on every robot column. The gain shrinks where the text already carries more of the name, to a few hundredths on questionnaire items. The item text already indicates the latent variable, and the prefix adds the exact item.

The backbones differ in their starting level and in the columns where they perform best. The two Qwen models lie within .10 of each other on the simulated arms and the real logs at every level and on the vehicle CAN from 50% masking on. The 3B Llama starts lowest on the real logs and reaches the other two by 75% masking. On latent variables, where the text carries part of the structure, it is the best of the three at 50 and 75% masking. At 90% masking the three lie within .10 of one another in NRR@1 on every column. At deep masking the structure therefore sets the level, and the choice of backbone has little effect. The same pattern holds under the supplied graph, where the gain on the vehicle CAN at 20% masking is .750, .400 and .600 for the three backbones.

J.6 TABLES

Component and backbone tables. These tables carry the numbers behind the component ablation and the backbone comparison of the main text, under both structure sources, with every metric each set of variables reports. Bold marks the best value in a column within a set of variables and underline the second best. A zero that a method returns by construction is never marked.

Table 27: Robot channels, one component removed at a time, under both structure sources.

structure	variables	variant	NRR@1 20	NRR@1 50	NRR@1 75	NRR@1 90	MRR 20	MRR 50	MRR 75	MRR 90	SDA 20	SDA 50	SDA 75	SDA 90
supplied graph	simulated arms	CAUSALBRIDGE	.146	**.198**	**.181**	**.055**	.317	.341	**.354**	**.225**	.170	.259	.222	.117
		− evidence	.104	.140	.099	.055	.240	**.350**	.277	.216	.149	.164	.131	.150
		− prefix	**.333**	.198	.088	.018	**.402**	.288	.206	.122	**.447**	**.293**	**.250**	.183
		− prefix − low-rank update	.021	.083	.033	.023	.164	.193	.135	.148	.255	**.293**	.205	**.211**
supplied graph	real logs	CAUSALBRIDGE	**.500**	**.333**	.244	.160	.566	**.451**	.369	.276	.545	**.440**	.333	.298
		− evidence	.500	.259	**.268**	**.200**	**.567**	.369	**.385**	**.351**	.545	.360	**.359**	**.319**
		− prefix	.250	.148	.171	.060	.427	.308	.310	.194	.545	.400	.308	.277
		− prefix − low-rank update	.167	.185	.146	.020	.434	.387	.278	.176	**.727**	.440	.333	.213
supplied graph	vehicle CAN	CAUSALBRIDGE	**.750**	**.300**	**.156**	**.145**	**.775**	**.553**	**.317**	**.378**	**1.000**	.300	.400	**.133**
		− evidence	.250	.250	.044	.009	.442	.449	.259	.295	.000	**.333**	.000	.000
		− prefix	.000	.100	.111	.064	.226	.327	.312	.281	.000	.300	**.500**	.000
		− prefix − low-rank update	.000	.083	.067	.091	.269	.322	.269	.297	.000	.100	**.500**	.000
discovered graph	simulated arms	CAUSALBRIDGE	**.083**	**.140**	**.137**	.046	**.224**	**.286**	**.278**	**.194**	.128	.190	.153	.164
		− evidence	.021	.091	.115	**.055**	.156	.256	.251	.189	.085	.147	.136	.150
		− prefix	.083	.116	.044	.032	.220	.223	.148	.125	**.277**	**.233**	**.199**	.169
		− prefix − low-rank update	.000	.050	.027	.023	.147	.176	.128	.131	.191	.103	.182	**.202**
discovered graph	real logs	CAUSALBRIDGE	**.417**	**.296**	**.195**	.120	.485	**.385**	**.314**	.248	.455	**.360**	**.359**	.255
		− evidence	.167	.037	.146	**.220**	.286	.190	.296	**.347**	.182	.120	.179	**.319**
		− prefix	.333	.111	.098	.020	**.533**	.265	.229	.154	**.636**	.320	.282	.149
		− prefix − low-rank update	.000	.148	.122	.020	.297	.334	.284	.181	.455	.320	.282	.128
discovered graph	vehicle CAN	CAUSALBRIDGE	**.800**	**.367**	**.244**	**.218**	**.859**	**.570**	**.412**	**.397**	**.800**	**.367**	**.500**	**.367**
		− evidence	.450	.333	.133	.182	.633	.528	.365	.370	.000	.333	.050	.333
		− prefix	.100	.167	.122	.073	.320	.395	.324	.297	.100	.200	.500	.000
		− prefix − low-rank update	.100	.050	.022	.073	.343	.279	.236	.265	.000	.100	.500	.000

structure	variables	variant	identification 20	identification 50	identification 75	identification 90	type acc 20	type acc 50	type acc 75	type acc 90	coord acc 20	coord acc 50	coord acc 75	coord acc 90
supplied graph	simulated arms	CAUSALBRIDGE	.167	**.252**	**.242**	**.106**	**.542**	.612	**.703**	**.525**	.149	**.233**	**.216**	.132
		− evidence	.130	.215	.155	.105	.438	**.719**	.626	.484	.106	.190	.125	**.136**
		− prefix	**.354**	.229	.162	.069	.458	.347	.297	.173	**.340**	.224	.153	.052
		− prefix − low-rank update	.109	.190	.078	.084	.375	.281	.214	.288	.043	.172	.074	.061
supplied graph	real logs	CAUSALBRIDGE	**.500**	**.407**	.268	.208	**.750**	**.593**	.610	.500	**1.000**	**.682**	.455	**.553**
		− evidence	.500	.259	**.293**	**.260**	.667	.556	**.658**	**.640**	.875	.636	.455	.447
		− prefix	.273	.231	.250	.149	.333	.333	.366	.240	.750	.500	**.606**	.526
		− prefix − low-rank update	.333	.333	.128	.120	.167	.222	.195	.260	.875	.591	.485	.500
supplied graph	vehicle CAN	CAUSALBRIDGE	**.750**	**.317**	**.216**	**.200**	**.750**	**.650**	**.289**	**.382**	**1.000**	**.300**	**.400**	**.133**
		− evidence	.250	.317	.115	.100	.750	.567	.267	.273	.000	.267	.000	.000
		− prefix	.000	.100	.213	.136	.200	.217	.111	.291	.000	.000	.000	.000
		− prefix − low-rank update	.105	.133	.091	.074	.250	.483	.000	.264	.000	.000	.000	.000
discovered graph	simulated arms	CAUSALBRIDGE	**.106**	**.203**	**.208**	**.116**	.396	.537	**.577**	**.420**	.106	**.181**	**.159**	**.145**
		− evidence	.083	.190	.164	.106	.375	**.579**	.522	.415	.064	.129	.119	.145
		− prefix	.104	.128	.099	.070	**.438**	.347	.286	.256	**.128**	.078	.091	.042
		− prefix − low-rank update	.021	.076	.096	.089	.417	.364	.225	.265	.021	.035	.040	.052
discovered graph	real logs	CAUSALBRIDGE	**.417**	.296	**.244**	.180	**.667**	**.518**	.512	.460	.750	**.545**	**.606**	**.579**
		− evidence	.167	.038	.146	**.220**	.583	.370	**.585**	**.600**	.625	.409	.485	.474
		− prefix	.167	.185	.146	.000	.333	.259	.342	.220	**1.000**	.455	.606	.447
		− prefix − low-rank update	.083	**.370**	.171	.020	.083	.259	.293	.280	.625	.500	.485	.474
discovered graph	vehicle CAN	CAUSALBRIDGE	**.800**	**.400**	**.300**	**.282**	.900	**.683**	**.356**	**.391**	**.800**	**.367**	**.500**	**.367**
		− evidence	.450	.367	.189	.255	**.950**	.667	.344	.364	.000	.333	.050	.333
		− prefix	.100	.200	.211	.157	.150	.300	.133	.318	.100	.000	.000	.000
		− prefix − low-rank update	.050	.133	.023	.135	.050	.367	.000	.009	.000	.000	.000	.000

Table 28: Questionnaire items, one component removed at a time, under both structure sources.

structure	variant	NRR@1 20	NRR@1 50	NRR@1 75	NRR@1 90	MRR 20	MRR 50	MRR 75	MRR 90	SDA 20	SDA 50	SDA 75	SDA 90	identification 20	identification 50	identification 75	identification 90
supplied graph	CAUSALBRIDGE	**.126**	**.097**	**.064**	**.038**	**.254**	**.220**	**.179**	**.121**	**.200**	**.176**	**.147**	**.129**	**.351**	**.314**	**.304**	.227
	− evidence	.067	.056	.042	.025	.177	.161	.130	.092	.092	.092	.081	.087	.190	.218	.174	.131
	− prefix	.074	.053	.050	.018	.217	.184	.165	.101	.123	.103	.122	.109	.344	.314	.286	.227
	− prefix − low-rank update	.074	.051	.051	.017	.219	.182	.166	.100	.126	.102	.119	.109	.338	.309	.280	**.228**
discovered graph	CAUSALBRIDGE	**.121**	**.098**	**.059**	**.028**	**.253**	**.216**	**.164**	**.105**	**.226**	**.189**	**.164**	**.147**	.356	.364	.283	.217
	− evidence	.087	.047	.043	.024	.196	.143	.117	.084	.172	.156	.141	.123	.247	.250	.199	.135
	− prefix	.079	.053	.044	.014	.229	.190	.158	.091	.169	.157	.152	.123	**.369**	**.386**	.302	.226
	− prefix − low-rank update	.085	.054	.046	.017	.227	.188	.162	.094	.174	.163	.154	.125	.364	.384	**.309**	**.229**

Table 29: Latent variables, one component removed at a time, under both structure sources.

structure	variant	NRR@1 20	NRR@1 50	NRR@1 75	NRR@1 90	MRR 20	MRR 50	MRR 75	MRR 90	identification 20	identification 50	identification 75	identification 90	naming quality 20	naming quality 50	naming quality 75	naming quality 90
supplied graph	CAUSALBRIDGE	**.485**	.415	**.415**	.238	**.624**	.560	**.551**	**.400**	**.627**	**.537**	**.535**	**.351**	**.596**	**.518**	**.500**	**.332**
	− evidence	.446	**.450**	.381	**.262**	.579	**.566**	.507	.396	.539	.479	.496	.290	.535	.475	.485	.278
	− prefix	.462	.404	.319	.208	.600	.550	.475	.371	.496	.465	.465	.281	.496	.465	.462	.273
	− prefix − low-rank update	.477	.419	.331	.212	.614	.560	.480	.376	.512	.492	.454	.285	.512	.492	.450	.277
discovered graph	CAUSALBRIDGE	**.540**	**.486**	.386	.240	**.660**	**.623**	.560	.424	.531	**.511**	.441	**.254**	.446	.426	.390	.217
	− evidence	.503	.437	.357	**.246**	.637	.582	.530	**.428**	.503	.480	.385	.241	.491	.471	.382	**.229**
	− prefix	.414	.386	**.406**	.186	.590	.565	**.581**	.404	**.549**	.474	.443	.229	**.537**	.474	**.437**	.226
	− prefix − low-rank update	.394	.389	.403	.186	.576	.568	.579	.401	.531	.480	**.446**	.226	.520	**.480**	**.437**	.223

Table 30: Robot channels on three language models, each with CAUSALBRIDGE and as the base model, under both structure sources.

structure	variables	model	NRR@1 20	NRR@1 50	NRR@1 75	NRR@1 90	MRR 20	MRR 50	MRR 75	MRR 90	SDA 20	SDA 50	SDA 75	SDA 90
supplied graph	simulated arms	Qwen3-4B	.146	**.198**	**.181**	.055	.317	**.341**	**.354**	.225	.170	.259	**.222**	.117
		Qwen3-4B base model	.021	.083	.033	.023	.164	.193	.135	.148	.255	**.293**	.205	.211
		Qwen3-8B	.167	.182	.170	**.119**	.326	.329	.348	**.265**	.213	.259	.216	.197
		Qwen3-8B base model	.083	.058	.027	.046	.221	.168	.141	.165	.255	.233	.210	.207
		Llama-3.2-3B	**.271**	.157	.154	.105	**.393**	.300	.304	.251	**.319**	.259	.182	**.225**
		Llama-3.2-3B base model	.021	.017	.044	.032	.137	.113	.141	.126	.213	.207	.148	.188
supplied graph	real logs	Qwen3-4B	**.500**	**.333**	.244	.160	**.566**	.451	.369	.276	.545	.440	.333	.298
		Qwen3-4B base model	.167	.185	.146	.020	.434	.387	.278	.176	**.727**	.440	.333	.213
		Qwen3-8B	.250	**.333**	**.317**	**.200**	.395	**.466**	**.456**	**.326**	.364	.320	.333	**.319**
		Qwen3-8B base model	.083	.074	.049	.020	.308	.286	.207	.145	.545	**.520**	.256	.234
		Llama-3.2-3B	.333	.259	.244	.120	.430	.367	.349	.253	.364	.440	**.385**	**.319**
		Llama-3.2-3B base model	.000	.074	.098	.080	.177	.266	.228	.195	.000	.200	.179	.213
supplied graph	vehicle CAN	Qwen3-4B	**.750**	.300	.156	.145	.775	.553	.317	.378	**1.000**	.300	.400	.133
		Qwen3-4B base model	.000	.083	.067	.091	.269	.322	.269	.297	.000	.100	**.500**	.000
		Qwen3-8B	.450	**.417**	**.256**	.154	.585	**.584**	.401	.327	.100	**.333**	**.500**	.300
		Qwen3-8B base model	.050	.000	.044	.027	.332	.249	.253	.253	.000	.000	**.500**	.000
		Llama-3.2-3B	.700	.300	.244	**.182**	**.776**	.528	**.418**	**.398**	.600	**.333**	**.500**	**.333**
		Llama-3.2-3B base model	.100	.167	.078	.145	.281	.383	.247	.331	.000	.133	.000	.267
discovered graph	simulated arms	Qwen3-4B	.083	.140	**.137**	.046	.224	.286	.278	.194	.128	.190	.153	.164
		Qwen3-4B base model	.000	.050	.027	.023	.147	.176	.128	.131	.191	.103	.182	.202
		Qwen3-8B	**.125**	**.157**	**.137**	**.078**	.255	**.288**	**.279**	**.198**	.191	.198	.176	**.207**
		Qwen3-8B base model	.000	.050	.011	.009	.155	.184	.121	.113	.085	.121	**.188**	.164
		Llama-3.2-3B	.104	.124	.115	.073	**.256**	.261	.251	.193	.170	**.216**	.142	.197
		Llama-3.2-3B base model	.021	.033	.027	.041	.162	.135	.126	.137	**.213**	.190	.165	.169
discovered graph	real logs	Qwen3-4B	**.417**	**.296**	.195	.120	.485	.385	.314	.248	**.455**	.360	**.359**	.255
		Qwen3-4B base model	.000	.148	.122	.020	.297	.334	.284	.181	**.455**	.320	.282	.128
		Qwen3-8B	**.417**	.259	**.244**	**.140**	**.508**	**.393**	**.386**	**.268**	**.455**	.320	**.359**	.277
		Qwen3-8B base model	.000	.185	.098	.020	.231	.353	.231	.158	.273	**.400**	.256	.191
		Llama-3.2-3B	.083	.148	**.244**	.100	.233	.308	.355	.231	.091	.280	**.359**	**.383**
		Llama-3.2-3B base model	.000	.074	.098	.060	.155	.228	.218	.212	.091	.160	.205	.170
discovered graph	vehicle CAN	Qwen3-4B	**.800**	.367	**.244**	**.218**	**.859**	.570	**.412**	**.397**	**.800**	**.367**	**.500**	**.367**
		Qwen3-4B base model	.100	.050	.022	.073	.343	.279	.236	.265	.000	.100	**.500**	.000
		Qwen3-8B	.550	**.400**	.200	.182	.679	**.582**	.375	.353	.200	**.367**	**.500**	.333
		Qwen3-8B base model	.150	.033	.044	.009	.320	.287	.239	.233	.000	.000	.150	.000
		Llama-3.2-3B	.700	.333	.089	.182	.800	.534	.309	.376	.400	.300	.000	.333
		Llama-3.2-3B base model	.050	.150	.022	.045	.245	.329	.195	.204	.000	.100	.000	.033

Table 30: Continued with the same rows, for identification, type accuracy and coordinate accuracy.

			identification				type acc				coord acc			
structure	variables	model	20	50	75	90	20	50	75	90	20	50	75	90
supplied graph	simulated arms	Qwen3-4B	.167	.252	**.242**	.106	.542	**.612**	**.703**	**.525**	.149	.233	.216	.132
		Qwen3-4B base model	.109	.190	.078	.084	.375	.281	.214	.288	.043	.172	.074	.061
		Qwen3-8B	.229	**.267**	.222	**.188**	**.604**	**.612**	.692	.516	.170	**.267**	**.222**	**.183**
		Qwen3-8B base model	.085	.137	.084	.083	.458	.281	.220	.251	.043	.103	.074	.070
		Llama-3.2-3B	**.312**	.264	.228	.165	.583	.488	.588	.498	**.298**	.241	.188	.174
		Llama-3.2-3B base model	.000	.033	.034	.028	.312	.215	.203	.192	.000	.000	.017	.009
supplied graph	real logs	Qwen3-4B	**.500**	**.407**	.268	.208	**.750**	.593	.610	**.500**	**1.000**	**.682**	.455	.553
		Qwen3-4B base model	.333	.333	.128	.120	.167	.222	.195	.260	.875	.591	.485	.500
		Qwen3-8B	.333	.333	**.308**	**.260**	.500	**.630**	**.707**	.440	.500	.500	.545	.474
		Qwen3-8B base model	.333	.296	.171	.080	.167	.259	.415	.220	.875	.636	**.576**	.500
		Llama-3.2-3B	.364	.259	.268	.220	.500	.444	.488	.440	.875	.591	.545	**.605**
		Llama-3.2-3B base model	.167	.111	.103	.080	.000	.185	.195	.120	.000	.136	.030	.053
supplied graph	vehicle CAN	Qwen3-4B	**.750**	.317	.216	.200	.750	.650	.289	**.382**	**1.000**	.300	.400	.133
		Qwen3-4B base model	.105	.133	.091	.074	.250	.483	.000	.264	.000	.000	.000	.000
		Qwen3-8B	.450	**.433**	**.333**	.218	**.900**	**.750**	**.367**	.345	.100	**.333**	**.500**	.300
		Qwen3-8B base model	.100	.000	.079	.065	.300	.500	.011	.273	.000	.000	.000	.000
		Llama-3.2-3B	.700	.333	.311	**.227**	**.900**	.633	.344	.364	.600	**.333**	**.500**	**.333**
		Llama-3.2-3B base model	.100	.200	.125	.182	.200	.233	.033	.109	.000	.000	.000	.000
discovered graph	simulated arms	Qwen3-4B	.106	**.203**	**.208**	.116	.396	**.537**	**.577**	**.420**	.106	.181	**.159**	.145
		Qwen3-4B base model	.021	.076	.096	.089	.417	.364	.225	.265	.021	.035	.040	.052
		Qwen3-8B	**.174**	.202	.200	**.124**	.479	**.537**	.555	.406	**.170**	.190	**.159**	.145
		Qwen3-8B base model	.128	.094	.112	.057	.479	.380	.253	.210	.021	.052	.034	.042
		Llama-3.2-3B	.149	.186	.157	.120	**.500**	.471	.483	.361	.149	**.215**	.148	**.150**
		Llama-3.2-3B base model	.062	.041	.028	.009	.375	.248	.236	.201	.043	.009	.011	.005
discovered graph	real logs	Qwen3-4B	**.417**	.296	.244	.180	**.667**	.518	.512	**.460**	**.750**	.545	.606	.579
		Qwen3-4B base model	.083	**.370**	.171	.020	.083	.259	.293	.280	.625	.500	.485	.474
		Qwen3-8B	**.417**	.259	**.341**	**.245**	.583	**.556**	**.537**	.380	**.750**	.500	.545	.474
		Qwen3-8B base model	.167	**.370**	.225	.146	.250	.333	.390	.220	.500	**.591**	**.667**	.447
		Llama-3.2-3B	.250	.192	.293	.120	.250	.481	.512	.340	.375	.500	.606	**.658**
		Llama-3.2-3B base model	.000	.148	.073	.040	.000	.222	.073	.120	.000	.091	.091	.026
discovered graph	vehicle CAN	Qwen3-4B	**.800**	.400	**.300**	**.282**	.900	.683	**.356**	**.391**	**.800**	**.367**	**.500**	**.367**
		Qwen3-4B base model	.050	.133	.023	.135	.050	.367	.000	.009	.000	.000	.000	.000
		Qwen3-8B	.550	**.417**	.222	.191	.950	**.717**	.300	.364	.200	**.367**	**.500**	.333
		Qwen3-8B base model	.105	.117	.057	.028	.450	.533	.211	.273	.000	.000	.000	.000
		Llama-3.2-3B	.700	.383	.144	.236	**1.000**	.683	.222	.318	.400	.300	.000	.300
		Llama-3.2-3B base model	.050	.167	.056	.045	.000	.083	.022	.036	.000	.000	.000	.000

Table 31: Questionnaire items on three language models, each with CAUSALBRIDGE and as the base model, under both structure sources.

		NRR@1				MRR				SDA				identification			
structure	model	20	50	75	90	20	50	75	90	20	50	75	90	20	50	75	90
supplied graph	Qwen3-4B	**.126**	**.097**	**.064**	**.038**	**.254**	**.220**	**.179**	**.121**	**.200**	**.176**	**.147**	.129	**.351**	**.314**	**.304**	.227
	Qwen3-4B base model	.074	.051	.051	.017	.219	.182	.166	.100	.126	.102	.119	.109	.338	.309	.280	**.228**
	Qwen3-8B	.105	.079	.056	.032	.222	.199	.158	.113	.182	.151	.137	**.132**	.346	.294	.260	.214
	Qwen3-8B base model	.069	.053	.036	.012	.206	.180	.146	.095	.136	.107	.119	.115	.321	.296	.264	.223
	Llama-3.2-3B	.067	.071	.062	.033	.188	.181	.162	.107	.113	.116	.119	.120	.282	.254	.258	.180
	Llama-3.2-3B base model	.077	.068	.057	.032	.192	.176	.157	.106	.128	.123	.145	.129	.277	.270	.256	.209
discovered graph	Qwen3-4B	**.121**	**.098**	**.059**	.028	**.253**	**.216**	**.164**	**.105**	**.226**	**.189**	.164	**.147**	.356	.364	.283	.217
	Qwen3-4B base model	.085	.054	.046	.017	.227	.188	.162	.094	.174	.163	.154	.125	**.364**	**.384**	**.309**	**.229**
	Qwen3-8B	.082	.069	.051	**.031**	.199	.181	.140	.099	.162	.152	.147	.138	.285	.323	.241	.191
	Qwen3-8B base model	.074	.051	.042	.014	.221	.187	.156	.089	.167	.167	.161	.128	.338	.355	.285	.210
	Llama-3.2-3B	.097	.072	.046	.022	.226	.181	.139	.092	.208	.173	.141	.129	.331	.320	.245	.158
	Llama-3.2-3B base model	.085	.064	.057	**.031**	.208	.183	.155	.100	.190	.150	**.165**	.135	.356	.351	.284	.200

Table 32: Latent variables on three language models, each with CAUSALBRIDGE and as the base model, under both structure sources.

		NRR@1				MRR				identification				naming quality			
structure	model	20	50	75	90	20	50	75	90	20	50	75	90	20	50	75	90
supplied graph	Qwen3-4B	.485	.415	**.415**	.238	.624	.560	**.551**	.400	**.627**	.537	**.535**	.351	**.596**	.518	.500	.332
	Qwen3-4B base model	.477	.419	.331	.212	.614	.560	.480	.376	.512	.492	.454	.285	.512	.492	.450	.277
	Qwen3-8B	.454	.404	.346	.235	.582	.552	.512	.405	.573	.537	.504	**.377**	.569	.537	.492	**.369**
	Qwen3-8B base model	.385	.323	.338	.219	.539	.490	.488	.381	.486	.469	.462	.279	.486	.454	.435	.244
	Llama-3.2-3B	**.535**	**.454**	.373	**.273**	**.646**	**.588**	.528	**.423**	.558	**.560**	.523	.350	.558	**.560**	**.515**	.350
	Llama-3.2-3B base model	.369	.350	.304	.185	.521	.506	.467	.363	.485	.473	.462	.292	.485	.473	.462	.292
discovered graph	Qwen3-4B	**.540**	.486	.386	**.240**	**.660**	.623	.560	**.424**	.531	**.511**	.441	.254	.446	.426	.390	.217
	Qwen3-4B base model	.394	.389	.403	.186	.576	.568	.579	.401	.531	.480	.446	.226	.520	**.480**	.437	.223
	Qwen3-8B	.529	.477	.394	.211	.646	.613	.569	.417	.524	.456	.378	.209	.513	.427	.361	.203
	Qwen3-8B base model	.411	.406	.411	.214	.583	.574	.580	.413	.511	.489	**.447**	.239	.494	.457	.390	.225
	Llama-3.2-3B	.514	**.517**	**.426**	.226	.650	**.651**	**.583**	**.424**	**.563**	.481	.444	.226	**.563**	.479	**.438**	.223
	Llama-3.2-3B base model	.454	.471	.391	.211	.609	.621	.573	.423	.543	.461	.397	**.256**	.543	.461	.391	**.256**

K SETTINGS, COMPUTE AND LICENSES

Settings. Table 33 lists the settings of every stage.

Table 33: Settings of every stage.

Causal structure learning, robot and vehicle channels	
first difference	every state channel of a simulated arm, and a channel of a real log or a drive when its lag-one autocorrelation exceeds .95
input edge added	partial correlation $\geq .03$, simulated arms only
edge dropped	partial correlation $< .05$ given the other parents and the inputs
Constrained embedding optimization	
term weights	$\lambda_{\text{ind}} = .3$, $\lambda_{\text{dep}} = .3$ with $\kappa = .5$ on the pairs above the .7 quantile of dependence, $\lambda_{\text{con}} = 1$
regularizer weights	$\lambda_{\text{bank}} = 1$, $\lambda_{\text{nbr}} = 1$, $\lambda_{\text{norm}} = .1$
Translation via prefix adaptation	
prefix	length $m = 10$, low-rank update of rank 8
training	AdamW (Loshchilov & Hutter, 2019) with early stopping on the validation split, 4 epochs for the questionnaires and 2 epochs for the robots
loss weights	$\lambda_{\text{sim}} = .5$, $\lambda_{\text{ctr}} = .15$, $\lambda_{\text{sib}} = .5$
backbones	Qwen3-4B-Instruct-2507 (main) and Qwen3-8B, Llama-3.2-3B-Instruct
Evaluation	
masking	20, 50, 75 and 90% of the names, with five folds per scale
scoring encoder	e5-large-v2, frozen
judge	Gemma 3 12B, instruction-tuned, temperature 0, both candidate orders

What each stage costs. Every run reported here used one A100 class GPU. Training the questionnaire adapter takes about an hour, and the robot adapter reaches its selected checkpoint after eight and a half hours. Both are trained a single time. At inference one name is one prompt and one decoding call.

Naming a new system. The deployment cost is the cost of naming a system nobody has documented, and the three steps together take minutes on one GPU. Ten vehicle drives go from raw traces to named channels in about seven minutes. Structure learning takes five seconds, the optimization 334 seconds and writing the names 67 seconds. The names cover the 280 unnamed channels of the four masking levels under each of the eight variants that the evaluation runs. These variants are the full method and its ablations, and they yield 2,240 names in all. Structure learning takes a few seconds on a real log and under half a minute on a simulated arm. The structure of a questionnaire scale is learned in seconds to a few minutes. The optimization at one masking level takes a few minutes on the smallest scale, and naming runs at 18.5 variables per second. In the timed runs, the three

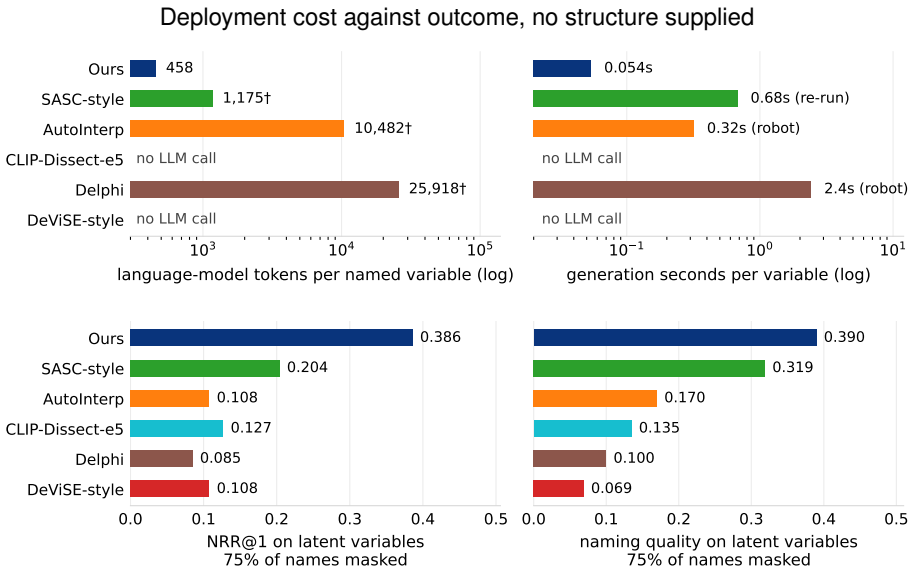

Figure 20: Cost and outcome per method with no structure supplied. Top, tokens and generation seconds per named variable on a log scale. Bottom, NRR@1 and naming quality at 75% masking. Tokens and naming metrics are for latent variables. Generation times are from questionnaire runs for CAUSALBRIDGE and SASC-style, and robot runs for Automated Interpretability and Delphi. A dagger marks a reconstructed count.

sampling methods take 0.32 to 2.4 seconds of generation per name against our 0.054 (Figure 20). They spend 2.6 to 56.6 times the tokens per latent variable (Table 34).

Cost per named variable. The cost of the sampling methods grows with the evidence a column offers, from the vehicle CAN to the questionnaires, since a channel description is shorter than an item text. Ours stays within a few hundred tokens on every column (Table 34). The lower row of Figure 20 reverses the order, and the cheapest method names the latent variables most accurately.

A cell of Table 34 is measured from stored request usage where the protocol was run against a served model. Otherwise it is reconstructed by replaying the published protocol through a token counter. Measured and reconstructed cells both count the protocol's nominal calls and exclude retry overhead. The generation seconds of Figure 20 all come from the same model snapshot. For CAUSALBRIDGE they are one inference pass over the questionnaire targets, and for SASC-style a timed sequential re-run of its six-call protocol. For Automated Interpretability and Delphi they are the median interval between consecutive cases of the robot run.

Table 34: Tokens spent per named variable, with the number of model calls the protocol makes. A dagger marks a cell reconstructed by replaying the published protocol through the token counter. The other cells are measured usage.

	questionnaire latent variables	simulated arms	real logs	vehicle CAN
CAUSALBRIDGE (1 call)	458	276	236	210
SASC-style (6 calls)	1,175†	1,125†	1,000†	774†
Automated Interpretability (2 calls)	10,482†	8,903	6,409	4,117
Delphi (3 calls)	25,918†	21,878	15,665	9,960

Licenses. The questionnaire responses come from the publicly available anonymized datasets listed in Table 4. The comma2k19 dataset, `robosuite` and two of the real logs are released under the MIT license, and the third log under Apache 2.0. ConceptNet Numberbatch is released under CC BY-SA 4.0, the Cognitive Atlas under CC BY-SA 3.0 and WordNet under its own permissive license. The scales themselves are the published instruments listed in Table 4.

L RECOVERED STRUCTURES

Questionnaires. Each figure draws the published scoring key of a scale above and the structure the discovery step recovers below. Each latent variable has a color, and the items it governs share that color. An item is drawn at its position along the row and marked with its identifier in the source data. A long identifier is truncated to fit. Table 6 in Appendix E gives the counts behind these figures and covers the three scales not drawn.

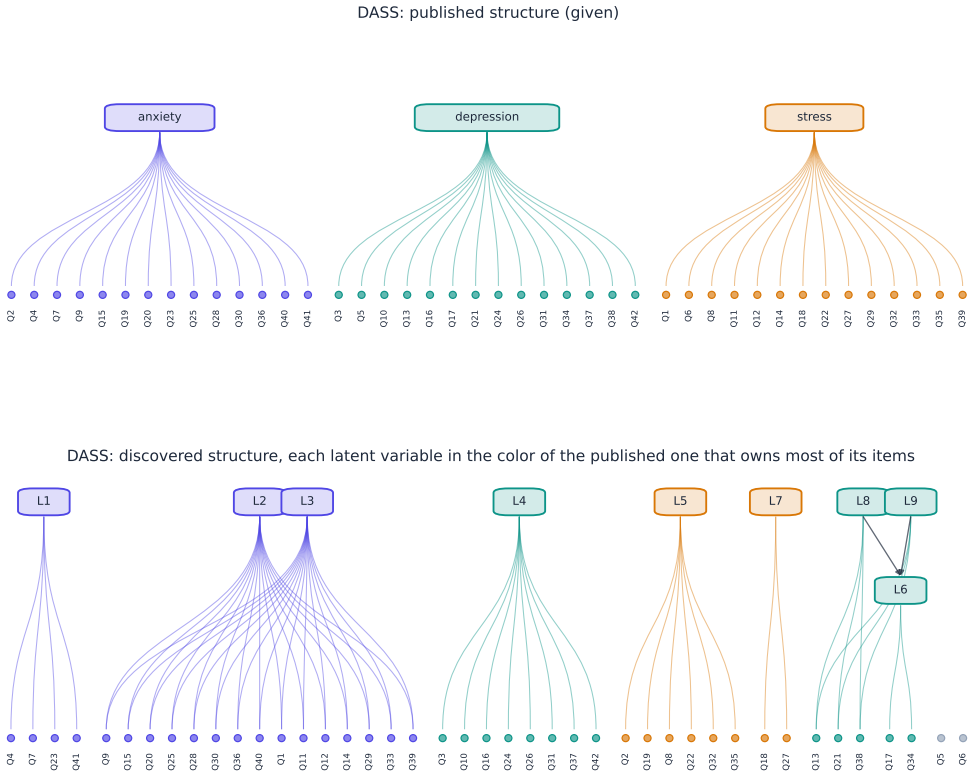

Figure 21: DASS, the published structure above and the discovered structure below. Six of the nine discovered latent variables lie inside one published latent variable.

Robot and vehicle channels. Each figure draws the supplied graph of one arm, log or drive on the left and the discovered graph on the right, with identical node positions. The graphs are drawn at the channel level. The lagged graph the search returns is folded so that an edge between two time copies of a pair of channels becomes one edge, and self edges are dropped. Table 7 in Appendix E counts the edges of the unfolded graphs of every arm.

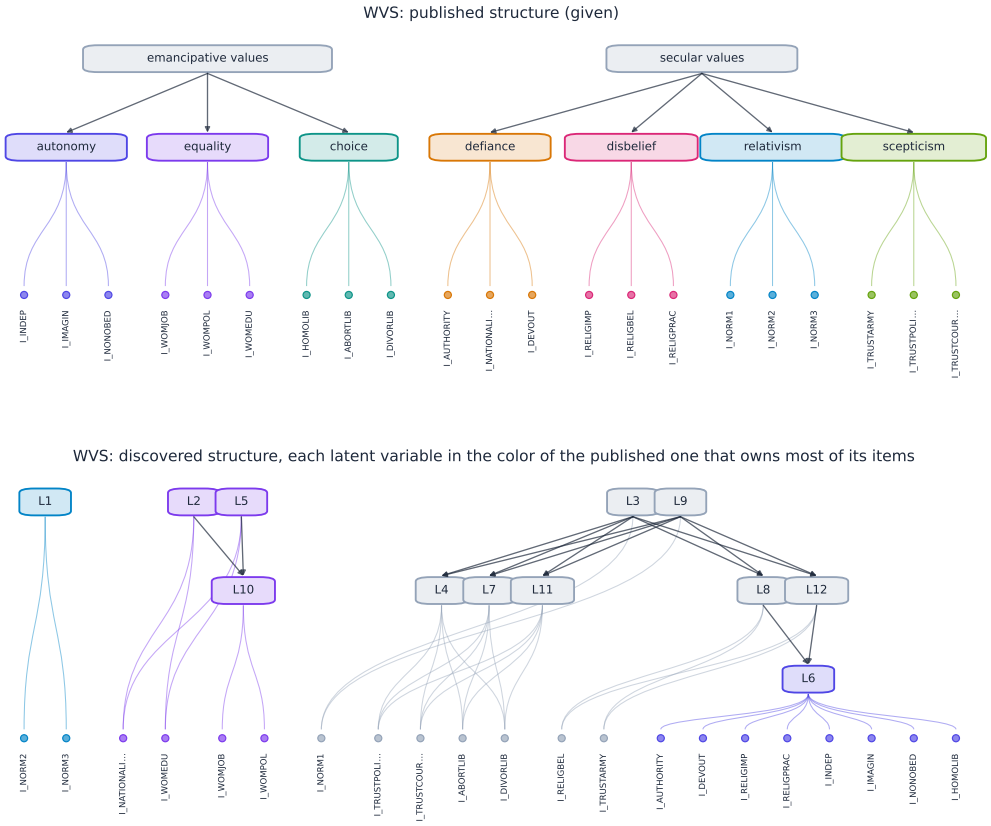

Figure 22: World Values Survey, the published structure above and the discovered structure below. The discovered groups cut across the Welzel sub-indices more than on any other scale.

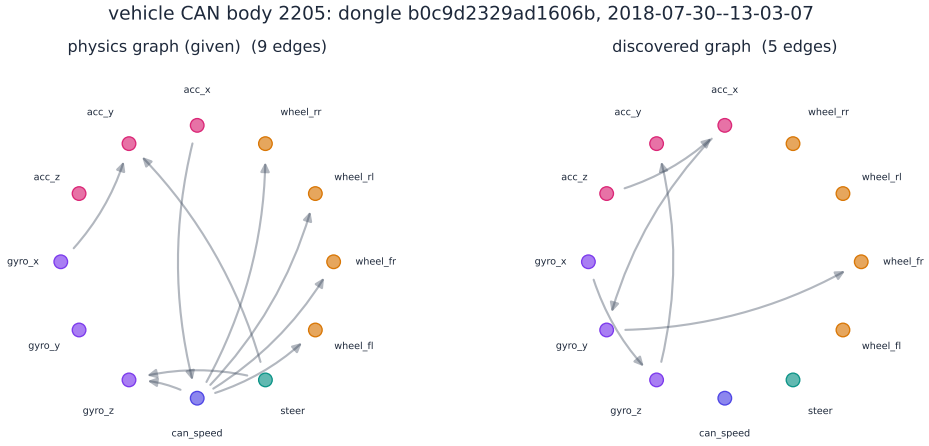

Figure 23: Vehicle CAN drive 2205, which is the drive of the introduction. In the supplied graph, the vehicle speed has edges to the four wheel speeds. The discovered graph has a path from the vertical acceleration through the longitudinal acceleration to the pitch rate. It also has an edge from the pitch rate to the front right wheel speed and two more edges among the rates and the lateral acceleration.

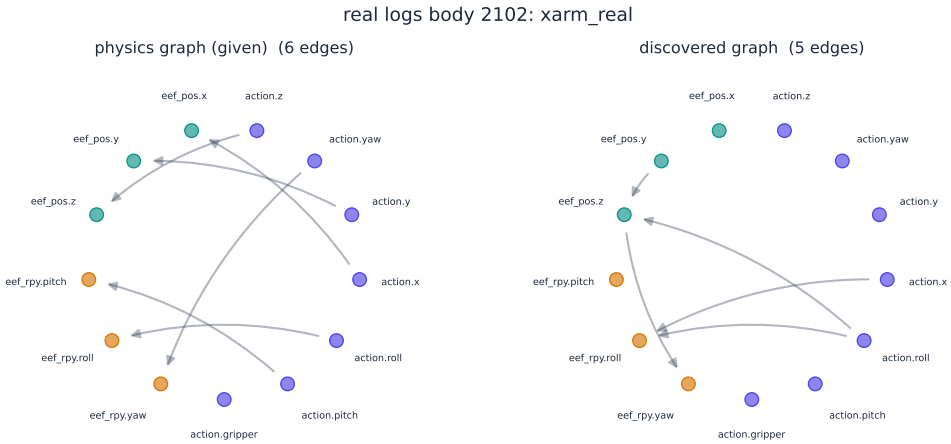

Figure 24: Real robot log 2102, which is the UTokyo xArm pick-and-place log.

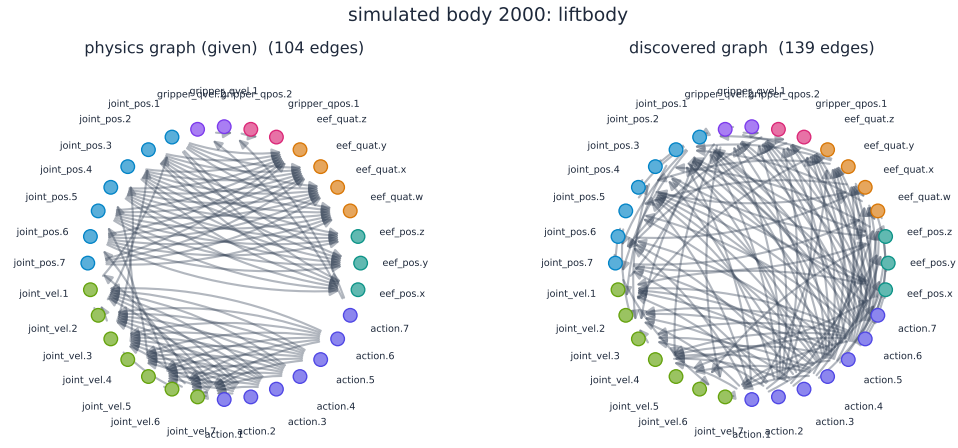

Figure 25: Simulated arm 2000, which is the Panda of the lift task. Its discovered graph is drawn against its true graph.