Formalizing Consistency and Coherence of Representation Learning

Harald Strömfelt
Dept of Computing
Imperial College London
London, SW7 2AZ
h.stromfelt17@imperial.ac.uk

Luke Dickens
Dept of Info Studies
University College London
London, WC1E 6BT
l.dickens@ucl.ac.uk

Artur d'Avila Garcez
Department of Computer Science
City, University of London
London, EC1V 0HB
a.garcez@city.ac.uk

Alessandra Russo
Department of Computing
Imperial College London
London, SW7 2AZ
a.russo@imperial.ac.uk

June 7, 2024

Abstract

In the study of reasoning in neural networks, recent efforts have sought to improve consistency and coherence of sequence models, leading to important developments in the area of neuro-symbolic AI. In symbolic AI, the concepts of consistency and coherence can be defined and verified formally, but for neural networks these definitions are lacking. The provision of such formal definitions is crucial to offer a common basis for the quantitative evaluation and systematic comparison of connectionist, neuro-symbolic and transfer learning approaches. In this paper, we introduce formal definitions of consistency and coherence for neural systems. To illustrate the usefulness of our definitions, we propose a new dynamic relation-decoder model built around the principles of consistency and coherence. We compare our results with several existing relation-decoders using a partial transfer learning task based on a novel data set introduced in this paper. Our experiments show that relation-decoders that maintain consistency over unobserved regions of representation space retain coherence across domains, whilst achieving better transfer learning performance.

*Funding in direct support of this work: EPSRC Training Grant, project reference EP/L504786/1.

1 Introduction

Humans are capable of learning concepts that can be applied to many different scenarios Inhelder1964-TEG, Piaget2005-TPO, Lake2017-BMT. An important principle is that human-like concepts remain *coherent* across contexts Nye2021-ICA. As an example, consider the concept of *ordinality*, *e.g.* "A is larger than B", which allows comparisons to be made between ordered sets. Ordinality should apply equally whether A and B are digits or a tower of blocks. It is said that a concept may pertain to a multitude of properties: position, volume, reach, *etc.* As long as one of these properties can be attributed to an object, a set of objects can be compared on that basis. All in all, if the concept of ordinality was to be learned in its most general form, its use should be consistent across objects and coherent across object properties.

In Nye2021-ICA, empirical results on story generation and instruction-following have shown that an intuitive use of consistency and coherence can increase the accuracy of neural networks. Following a neuro-symbolic perspective Garcez2020-NAT, it is argued in Nye2021-ICA that *System 1* approaches, fast and capable of learning patterns efficiently from data, "are often inconsistent and incoherent", and that "adding *System 2*-inspired logical reasoning" as a logical consistency, training-free module allows for an improved selection of candidate stories generated by *System 1*. While Nye2021-ICA makes an important contribution by exploring several variations on the theme, in this paper we offer a formal definition for consistency and coherence in the context of neural networks, in particular neuro-symbolic autoencoders. We also apply and evaluate consistency and coherence in transfer learning tasks, where we believe that the theme will have its most practical impact.

We argue that for a concept to be useful during transfer learning, the system of relations that define the concept in the source domain must be coherent with the target domain, whereby logical consistency achieved in the source is retained in the target domain. This is to say that the concept-specific relations learned in the source ought to be consistent with a logical theory that defines their semantics, and that such consistency must extend beyond the representations learned in the source domain and, in particular, hold for the embeddings learned in the target domain.

In this paper, we offer a formal definition for consistency and coherence of sub-symbolic representation learners, inspired by analogous definitions from symbolic AI. This is expected to define the conditions that make a learned concept transfer well across properties and objects. To evaluate the practical value of these definitions in a real setting, we derive a simple neuro-symbolic autoencoder architecture consisting of a neural encoder for objects coupled with consistent modular object relation-decoders. Relations such as isGreater, isEqual,... are evaluated on a proposed Partial Relation Transfer (PRT) learning task, between a new CLEVR-style BlockStacks data set and the MNIST handwritten digits data set, such that the learning of ordinality among the MNIST digits is evaluated against the learning of the relative position of a red block in a stack of multi-colored blocks. Our evaluation includes a comparison with several

existing relation-decoder models and results show that relation-decoders which maintain consistency over unobserved regions of representation space retain coherence across domains whilst achieving better transfer learning performance.[1] In summary, the contributions of this paper are:

- A formal definition of consistency and coherence for sub-symbolic learners offering a practical evaluation score for concept coherence;
- A derived model implementation and PRT data set and experimental setup used to evaluate the interplay between concept coherence and concept transfer;
- A comprehensive critical evaluation of results and comparison of multiple relation-decoder models, showing that improvements in concept coherence, as defined in this paper, correspond with improved concept transfer.

In Section 2 we provide the notation and required logic background. Section 3 formally defines coherence and consistency. Section 4 defines a practical consistency loss and Section 5 outlines our neuro-symbolic autoencoder. After detailing the PRT task and introducing the data set in Section 6, comparative experimental results are discussed in Section 7. We provide an overview of the related work in Section 8 and Section 9 concludes the paper with a discussion, including limitations and future work. We expand on the experimental results and setup, together with data set characteristics, model details and parameterization in the Appendices.[2]

2 Preliminaries

Notation: We reserve uppercase calligraphic letters to denote sets, and lowercase versions of the same letter to denote their elements, e.g. $\mathcal{S} = \{s_1, \ldots, s_n\}$ is a set $\mathcal{S}$ of n elements s_i. We indicate with $|\mathcal{S}| = n$ the cardinality of $\mathcal{S}$. We use uppercase roman letters to denote a random variable (e.g. S), and use the uppercase calligraphic version of the same letter ($\mathcal{S}$) to denote the set from which the random variable takes values according to some corresponding probability distribution p_{S}, over the elements of the set, such that $\sum_{i=1}^{|\mathcal{S}|} p_{\mathrm{S}}(s_i) = 1$ for a discrete $\mathcal{S}$. For brevity, we may write $p_{\mathrm{S}}(s_i)$ as $p(s_i)$, where the random variable is implied by the argument. We use bold font lowercase letters to denote vector elements, e.g. $\boldsymbol{s}_i \in \mathbb{R}^d$ is an d-dimensional vector element from the set $\mathcal{S} = \mathbb{R}^d$.

Logic and model-theoretic background: our proposed theory is based upon logic and model theoretic primitives. To avoid making this paper overly dense, we defer the details of the logic background to Appendix E and include here only the most important definitions supported by an illustrative example.

[1] This paper formalizes the theory and extends the empirical results first reported in Harald2021-CAC.

[2] The codebase for this paper can be found at `https://github.com/HStromfelt/neurips22-FCA`.

Definition 2.1 (Signature, Arity, Domain, Interpretation, Structure)**.** The *signature* of a language $\mathcal{L}$ is a set of relations $\sigma = \{r \in \mathcal{L}\}$ whose elements have *arity* given by $\mathsf{ar} : \sigma \to \mathcal{N}$, where $\mathcal{N}$ is the set of natural numbers. Given a signature σ and a non-empty *domain* $\mathcal{S} = \{s_1, s_2, \ldots\}$, an *interpretation* $I_{\mathcal{S}_\sigma}$ of σ over elements of $\mathcal{S}$ assigns to each relation $r \in \sigma$ a set $I_{\mathcal{S}_\sigma}(r) \subseteq \mathcal{S}^{\mathsf{ar}(r)}$. A *structure* is a tuple $\mathcal{S}_\sigma = (\mathcal{S}, I_{\mathcal{S}_\sigma})$.

We construct universally quantified first-order logic formulae (called sentences) using the signature of $\mathcal{L}$. A set of sentences form a theory $\mathcal{T}$ and when a sentence τ is true in a structure $\mathcal{S}_\sigma$, we say that the structure satisfies τ, denoted as $\mathcal{S}_\sigma \models \tau$. This allows us to define a *model* of a theory:

Definition 2.2 (Model of a theory)**.** Let $\mathcal{T}$ be a theory written in a language $\mathcal{L}$ and let $\mathcal{S}_\sigma = (\mathcal{S}, I_{\mathcal{S}_\sigma})$ be a structure, where σ is the signature of $\mathcal{L}$. $\mathcal{S}_\sigma$ is a *model of* $\mathcal{T}$ if and only if $\mathcal{S}_\sigma \models \tau$ for every sentence $\tau \in \mathcal{T}$.

Example 1. Let $\mathcal{S}$ is a domain of images of handwritten digits and σ the signature of binary relations $\sigma = \{\mathsf{isGreater}, \mathsf{isEqual}, \mathsf{isLess}, \mathsf{isSuccessor}, \mathsf{isPredecessor}\}$, or for short $\sigma = \{\mathsf{G}, \mathsf{E}, \mathsf{L}, \mathsf{S}, \mathsf{P}\}$. Let $\mathcal{T}$ be the theory that defines ordinality including, for instance, the sentence $\forall i, j.\, \mathsf{G}(i, j) \to \neg\mathsf{E}(i, j)$ (if a digit is greater than another then they are not equal). Any structure $\mathcal{S}_\sigma = (\mathcal{S}, I_{\mathcal{S}_\sigma})$ with interpretations $I_{\mathcal{S}_\sigma}$ of σ that captures a total order over the elements of $\mathcal{S}$ is a model of $\mathcal{T}$.

3 A Formalization of Consistency and Coherence

In this section, we turn our attention to the challenge of learning a model of a theory (Def. 2.2) over a real-world domain $\mathcal{S}$ given a signature σ. Here, a learner must determine an appropriate interpretation over real-world data, such as images or other perceptions. This can be challenging because, firstly, we may only have a partial description of the interpretation, and secondly data may be noisy and contain information that is not relevant to the theory. For example, the handwritten digits in the MNIST data set contain stylistic details such as line thickness and digit skew that are irrelevant to the notion of ordinality, which makes learning the structure from Example 1 non-trivial.

Following the convention from the autoencoder disentanglement literature Bengio2013-RLR, Kingma2014-AEV, Higgins2017-BVL, Higgins2018-TDD, we make the assumption that real-world observations S are drawn from some conditional distribution $p_{\mathrm{S}|\mathrm{Z}}$, where Z is a latent random variable, itself drawn from prior p_{Z}. It is therefore useful to define a domain *encoding* of the form:

$$\psi_{\mathcal{S}} : \mathcal{S} \to \mathcal{Z}, \tag{1}$$

tasked with approximating the conditional expectation of the posterior, *i.e.* $\psi_{\mathcal{S}}(s) = \mathbb{E}_{p_{\mathrm{Z}|S}}[\mathrm{Z}|s]$. Since obtaining an interpretation from domain encodings for a given signature may require dealing with noise, we express the interpretation of

relations over real-world data by belief functions Paris1994-TUR, Paris2015-PIL over the space $\mathcal{Z}$, and refer to these as *relation-decoders*:

$$\phi_r : \mathcal{Z}^{\mathsf{ar}(r)} \to (0,1) \tag{2}$$

with $\phi = \{\phi_r : r \in \sigma\}$. Concretely, for a binary relation r and ordered pair $(s_i, s_j) \in \mathcal{S}^2$, $\phi_r(\psi_{\mathcal{S}}(s_i), \psi_{\mathcal{S}}(s_j))$ describes the belief that $(s_i, s_j) \in I_{\mathcal{S}_\sigma}(r)$. A belief $\phi_r(\psi_{\mathcal{S}}(s_i), \psi_{\mathcal{S}}(s_j)) \approx 1$ signifies a strong belief that $(s_i, s_j) \in I_{\mathcal{S}_\sigma}(r)$ and $\phi_r(\psi_{\mathcal{S}}(s_i), \psi_{\mathcal{S}}(s_j)) \approx 0$ signifies a strong belief that $(s_i, s_j) \notin I_{\mathcal{S}_\sigma}(r)$. Together, $\psi_{\mathcal{S}}$ and ϕ allow us to define a belief-based analogue to a structure.

Definition 3.1 (Soft-Structure/Soft-Substructure)**.** Given a signature σ, a (possibly infinite) set $\mathcal{Z}$ and relation-decoders ϕ, a *soft-structure* is a tuple $\tilde{\mathcal{Z}}_\sigma = (\mathcal{Z}, \phi)$. For a finite domain $\mathcal{S}$ and encoding $\psi_{\mathcal{S}} : \mathcal{S} \to \mathcal{Z}$, $\tilde{\mathcal{S}}_\sigma = (\psi_{\mathcal{S}}(\mathcal{S}), \phi)$ is called a finite *soft-substructure* of $\tilde{\mathcal{Z}}_\sigma$, with sub-domain $\psi_{\mathcal{S}}(\mathcal{S}) = \{\psi_{\mathcal{S}}(s) | s \in \mathcal{S}\} \subseteq \mathcal{Z}$.

A soft-structure can be used to learn a logic structure over a real-world domain through learning $\psi_{\mathcal{S}}$ and ϕ. Clearly, a finite soft-substructure is a soft-structure. In a real-world domain, there may be only partial information about the values of an interpretation, and there may be errors in that partial interpretation. To determine the degree to which a soft-structure *supports* any given structure, we introduce the following measure:

$$p(\mathcal{S}_\sigma | \tilde{\mathcal{S}}_\sigma) = \prod_{r \in \sigma} \prod_{O \in \mathcal{S}^{\mathsf{ar}(r)}} f(\phi_r, \psi_{\mathcal{S}}, O, \gamma^r_{O,\mathcal{S}_\sigma}) \tag{3}$$

with

$$f(\phi_r, \psi_{\mathcal{S}}, O, \gamma^r_{O,\mathcal{S}_\sigma}) = (\phi_r(\psi_{\mathcal{S}}(O)))^{\gamma^r_{O,\mathcal{S}_\sigma}} \cdot (1 - \phi_r(\psi_{\mathcal{S}}(O)))^{1-\gamma^r_{O,\mathcal{S}_\sigma}}, \tag{4}$$

where $\gamma^r_{O,\mathcal{S}_\sigma} = 1$ if $O \in I_{\mathcal{S}_\sigma}(r)$, and 0 otherwise; we use $\phi_r(\psi_{\mathcal{S}}(O))$ as shorthand for $\phi_r(\psi_{\mathcal{S}}(s_1), \ldots, \psi_{\mathcal{S}}(s_n))$ for $n = \mathsf{ar}(r)$. Eqn. 3 expresses the assumption that, given a finite soft-structure, the beliefs in what constitutes the interpretations of different relations are independent of one another. It is straightforward to show that $\sum_{\mathcal{S}_\sigma} p(\mathcal{S}_\sigma | \tilde{\mathcal{S}}_\sigma) = 1$ (summed over all possible structures with domain $\mathcal{S}$ and signature σ) and so Eqn. 3 can be treated as a probability measure, where $p(\mathcal{S}_\sigma | \tilde{\mathcal{S}}_\sigma) \approx 1$ means that there is a high probability that the interpretation sampled from $\tilde{\mathcal{S}}_\sigma$ will be $I_{\mathcal{S}_\sigma}$. If we have a theory $\mathcal{T}$ over σ then it is natural to ask with what weight $\tilde{\mathcal{S}}_\sigma$ supports any given structure that is a model of $\mathcal{T}$. In the following, we use *model weight*, $\Gamma^{\tilde{\mathcal{S}}_\sigma}_{\mathcal{T}}$, to describe the support given by $\tilde{\mathcal{S}}_\sigma$ to models of $\mathcal{T}$:

$$\Gamma^{\tilde{\mathcal{S}}_\sigma}_{\mathcal{T}} = \sum_{\mathcal{S}_\sigma \in \mathcal{M}^{\mathcal{T}}_{\mathcal{S}}} p(\mathcal{S}_\sigma | \tilde{\mathcal{S}}_\sigma) \tag{5}$$

where $\mathcal{M}^{\mathcal{T}}_{\mathcal{S}}$ is the set of all structures with domain $\mathcal{S}$ that are models of $\mathcal{T}$. This lets us compare soft-structures, wherein a good soft-structure will be one that has a high model weight.

Definition 3.2 (ϵ-Consistency of soft-structures)**.** Given a finite soft-structure $\tilde{\mathcal{S}}_\sigma$ and an arbitrarily small number ϵ, if $1 - \Gamma_{\mathcal{T}}^{\tilde{\mathcal{S}}_\sigma} \leq \epsilon$ then we say that $\tilde{\mathcal{S}}_\sigma$ is ϵ-*consistent* with theory $\mathcal{T}$.

We propose ϵ-consistency as an appropriate measure of the notion of consistency presented in Nye2021-ICA. A consistent soft-structure $\tilde{\mathcal{S}}_\sigma$ ensures that ϕ gives high belief only to interpretations that satisfy, and therefore are logically consistent with, theory $\mathcal{T}$. As expected, consistency pertains to the domain encodings of $\tilde{\mathcal{S}}_\sigma$, *i.e.* $\psi_{\mathcal{S}}(\mathcal{S})$. For a concept to be learned in a manner comparable to how a human might learn, we would expect consistency to carry over to new domains with their corresponding soft-structures, which motivates our definition of coherence between soft-structures, as follows. Consider a situation where a deep network has already learned from the MNIST data set a soft-structure that has high model weight, given the relations $\{\mathsf{G}, \mathsf{E}, \mathsf{L}, \mathsf{S}, \mathsf{P}\}$ from Example 1. Now, consider a new domain of images, $\mathcal{Y}$, showing single block stacks of different heights, and we wish to re-use the signature of ordinal relations and $\mathcal{T}$ from Example 1. Let $I_{\mathcal{Y}_\sigma}$ be an interpretation in the new domain that orders images according to block stack height and that is a model of $\mathcal{T}$. We can summarise this with the following two structures:

$$\mathcal{X}_\sigma = (\mathcal{X}, I_{\mathcal{X}_\sigma}) \in \mathcal{M}_{\mathcal{X}}^{\mathcal{T}} \quad \text{and} \quad \mathcal{Y}_\sigma = (\mathcal{Y}, I_{\mathcal{Y}_\sigma}) \in \mathcal{M}_{\mathcal{Y}}^{\mathcal{T}}, \tag{6}$$

where $\mathcal{X}_\sigma$ is the structure from Example 1 with a domain of handwritten digits and $\mathcal{Y}_\sigma$ is our new structure, with a domain of block stack images. These can be learned by soft-structures:

$$\tilde{\mathcal{X}}_\sigma = (\psi_{\mathcal{X}}(\mathcal{X}), \phi) \qquad \text{and} \qquad \tilde{\mathcal{Y}}_\sigma = (\psi_{\mathcal{Y}}(\mathcal{Y}), \phi), \tag{7}$$

which use domain-specific encoders, $\psi_{\mathcal{X}}$ and $\psi_{\mathcal{Y}}$, but share the same relation-decoders. As we know that $\tilde{\mathcal{X}}_\sigma$ has a high model weight and since ϕ is shared with $\tilde{\mathcal{Y}}_\sigma$, a natural question to ask is: under what conditions will a ϕ that is consistent over domain-encodings $\psi_{\mathcal{X}}(\mathcal{X})$ also be consistent over $\psi_{\mathcal{Y}}(\mathcal{Y})$? Concretely, we are interested in specifying when the following *coherence* condition holds.

Definition 3.3 (ϵ-Coherence across soft-structures)**.** Two soft-structures, $\tilde{\mathcal{X}}_\sigma$ and $\tilde{\mathcal{Y}}_\sigma$ that share relation-decoders ϕ, are said to be ϵ-*coherent* with respect to a theory $\mathcal{T}$, if $\tilde{\mathcal{X}}_\sigma$ is ϵ_1-consistent with $\mathcal{T}$, $\tilde{\mathcal{Y}}_\sigma$ is ϵ_2-consistent with $\mathcal{T}$, $\epsilon_1 \leq \epsilon$, and $\epsilon_2 \leq \epsilon$.

Coherence between $\tilde{\mathcal{X}}_\sigma$ and $\tilde{\mathcal{Y}}_\sigma$ as defined above means that the concept of ordinality that applies to digit ordering can also be applied to block stack height ordering. It is desirable that learning ordinality on the domain of digits produces a coherent concept of ordinality with respect to other ordinal properties, such as height. Since it is possible that $\psi_{\mathcal{S}}(\mathcal{X})$ and $\psi_{\mathcal{S}}(\mathcal{Y})$ produce unique encodings, coherence relies on ϕ's ability to generalize over possibly disjoint subsets of $\mathcal{Z}$.[3]

[3]If soft-structure $\tilde{\mathcal{Z}}_\sigma$ defined over the full space $\mathcal{Z}$ is consistent then coherence is guaranteed between all possible soft-substructures.

4 Measuring Consistency and Coherence

Calculating Eqn. 5 can be computationally too expensive for larger domains. An efficient approach to measuring the consistency of soft-structures is therefore required. In this section, we introduce a proxy measure for a soft-structure's ϵ-consistency and ϵ-coherence with a given theory when access to every logical model is not available or is computationally intractable.

Suppose that there is a fixed domain $\mathcal{S}$ and theory $\mathcal{T}$ whose sentences use relations from a signature σ. Let $k \in \{1, ..., K_0\}$ denote the index associated with each unique ground instance of the sentences in $\mathcal{T}$. Take $B_{\mathcal{T}}$ to be a Boolean random variable. For the k-th grounding in $\mathcal{T}$, the probability of the theory being satisfied under soft-structure $\tilde{\mathcal{S}}_\sigma$ is expressed as $p(b_{\mathcal{T}} = 1|\tilde{\mathcal{S}}_\sigma, k)$. Conversely, the probability of non-satisfaction is given by $p(b_{\mathcal{T}} = 0|\tilde{\mathcal{S}}_\sigma, k)$. For a model of (universally-quantified) theory $\mathcal{T}$, $\mathcal{S}_\sigma \in \mathcal{M}_{\mathcal{S}}^{\mathcal{T}}$, any grounding k of the $\mathcal{T}$ is always satisfied (by definition), and thus $p(b_{\mathcal{T}} = 1|\mathcal{S}_\sigma, k) = 1$. When $\tilde{\mathcal{S}}_\sigma$ is consistent with $\mathcal{T}$ then we should also find that $p(b_{\mathcal{T}} = 1|\tilde{\mathcal{S}}_\sigma, k) \approx 1$ for all k. Hence, we define a consistency loss function as the expectation over a randomly-chosen grounding k of the binary cross-entropy between $p(B_{\mathcal{T}}|\mathcal{S}_\sigma, k)$ and $p(B_{\mathcal{T}}|\tilde{\mathcal{S}}_\sigma, k)$ for any $\mathcal{S}_\sigma \in \mathcal{M}_{\mathcal{S}}^{\mathcal{T}}$. This in turn simplifies to produce the expected negative log-likelihood of satisfying a random grounding of $\mathcal{T}$, as follows:

$$L(\mathcal{T}, \tilde{\mathcal{S}}_\sigma) = \mathbb{E}_{k\sim p(k)}[-\ln p(b_{\mathcal{T}} = 1|\tilde{\mathcal{S}}_\sigma, k)]. \tag{8}$$

where $p(k) = \frac{1}{K_0}$ is the uniform distribution over the set of unique groundings. A measure based on this loss is required to enable a practical evaluation of consistency, acting as an approximation for consistency. More precisely, we define $\bar{\Gamma}_{\mathcal{T}}^{\tilde{\mathcal{S}}_\sigma} = \exp(-L(\mathcal{T}, \tilde{\mathcal{S}}_\sigma))$ as a proxy measure of $\Gamma_{\mathcal{T}}^{\tilde{\mathcal{S}}_\sigma}$, and say that soft-structure $\tilde{\mathcal{S}}_\sigma$ is $\bar{\epsilon}$-proxy consistent with $\mathcal{T}$ if

$$\ln \frac{1}{1-\bar{\epsilon}} \geq L(\mathcal{T}, \tilde{\mathcal{S}}_\sigma) \tag{9}$$

where $\bar{\epsilon} \geq 1 - \bar{\Gamma}_{\mathcal{T}}^{\tilde{\mathcal{S}}_\sigma}$. Due to the relationship between $\bar{\epsilon}$ and $L(\mathcal{T}, \tilde{\mathcal{S}}_\sigma)$, we take the proxy measure of coherence to be the smallest satisfiable value of $L(\mathcal{T}, \tilde{\mathcal{S}}_\sigma)$ between domains.[4]

Although our treatment of consistency has thus far focused on a particular theory $\mathcal{T}$, notice that a subset of the sentences of $\mathcal{T}$ form a partial theory, which is itself a theory. This means that consistency can be evaluated given a partial (even single sentence) theory, allowing us to examine consistency losses for any partial specifications of a given domain of interest. In this paper, we evaluate the proposed consistency loss against two partial specifications within the theory of ordinality. These are named Consistency-Across (Con-A) and Consistency-Individual (Con-I) in the experiments that will follow.[5] Con-A includes the sentences that determine inter-relation behavior, for instance the

[4]The complete derivation of loss function and bounds is presented in Appendix G.

[5]Truth-tables for each consistency formula are given in Appendix F.

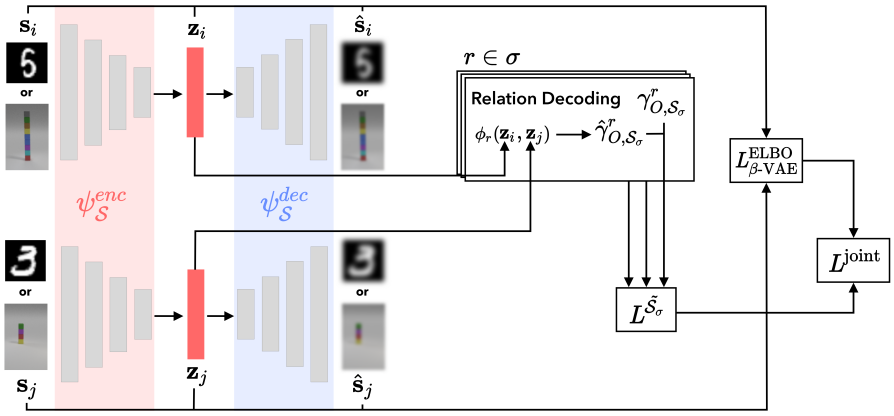

Figure 1: Network architecture used for PRT tasks. In our experiments $\boldsymbol{s}_{i/j}$ are either MNIST (domain $\mathcal{X}$) or BlockStacks (domain $\mathcal{Y}$) images. Relational learning is performed on the source ($\mathcal{S} = \mathcal{X}$) MNIST domain (to learn e.g. that digit 5 is greater than 3). Moving to the target ($\mathcal{S} = \mathcal{Y}$) domain (stacks of blocks) involves training a new image autoencoder together with a subset of the relation-decoders from MNIST with fixed parameters. The remaining relations are held-out to evaluate zero-shot transfer learning performance. $\gamma^r_{O,\mathcal{S}_\sigma}$ provides the ground truth for the given structure, which ϕ_r predicts as $\hat{\gamma}^r_{O,\mathcal{S}_\sigma}$. As in Section 3, O is used to abbreviate $(\psi^{enc}_{\mathcal{S}}(\boldsymbol{s}_i), \psi^{enc}_{\mathcal{S}}(\boldsymbol{s}_j))$.

sentence $\forall i, j.\ (\mathsf{isGreater}(i, j) \rightarrow \neg\mathsf{isEqual}(i, j) \wedge \neg\mathsf{isLess}(i, j))$, stating that if i is greater than j then i must not be equal to or less than j. Con-I includes the sentences that are about a single relation, describing any property of an individual relation over objects (or images in the case of our experiments). Each relation may satisfy a number of properties, for example $\forall i, j, k.\ (\mathsf{isGreater}(i, j) \wedge \mathsf{isGreater}(j, k) \rightarrow \mathsf{isGreater}(i, k))$ represents transitivity of the isGreater relation. Transitivity is true for isGreater, but is false for other relations investigated in this paper, e.g. isSuccessor. We will evaluate consistency loss of transitivity (Con-I-T), asymmetry (Con-I-A) and reflexivity (Con-I-R) for the relations in Example 1. The evaluation of consistency loss for any available partial theory will be shown to provide a more nuanced perspective on model performance than accuracy results and disentanglement pressure alone during transfer learning.

5 A Consistent and Coherent Neuro-symbolic Autoencoder

In order to ground our definitions of consistency (Def. 3.2) and coherence (Def. 3.3) into a real system and evaluate their practical value, in this section we derive a simple neuro-symbolic autoencoder architecture which offers one of many possible implementations of the theory defined in Section 3. Figure 1 outlines

the main components of our autoencoder: a domain-encoder $\psi_{\mathcal{S}}$ and modular relation-decoders ϕ form an autoencoding architecture that, given a domain of images $\mathcal{S} \subset \mathbb{R}^{C\times W\times H}$ (C color channels, width W and height H) and a d-dimensional latent space $\mathcal{Z} = \mathbb{R}^d$, converts sub-symbolic encodings from $\psi_{\mathcal{S}}$ into a modular relational representation via decoding for each $\phi_r, r \in \sigma$. Additionally, to retain information in $\mathcal{Z}$ pertaining to $\mathcal{S}$ which is beyond the requirements of ϕ, a domain-decoder produces domain reconstructions $\hat{\mathcal{S}}$. In Figure 1, we use ψ_S^{enc} to refer to the domain-encoder and ψ_S^{dec} to the domain-decoder. Although in this paper we opt for an autoencoding architecture, our definitions of consistency and coherence are applicable to a wider range of neural architectures. For instance, a multi-layer perception network can be viewed as a set of encoding and decoding layers Shwartz2017-OTB. As long as the architecture offers explicit soft relation decodings, provided we can define a partial theory over them, we can define a consistency loss over the outputs.

To train the model, ground-truth interpretations $I_{\mathcal{S}_\sigma}$ are provided, allowing us to maximize directly Eqn. 3 via the negative log-likelihood loss:

$$L^{\tilde{\mathcal{S}}_\sigma} = -\log p(\mathcal{S}_\sigma|\tilde{\mathcal{S}}_\sigma), \tag{10}$$

To obtain informative latent representations for $\mathcal{S}$, we use a Variational Autoencoder (VAE), specifically the β-VAE Burgess2017-UDI, Higgins2017-BVL, Kingma2014-AEV due to its simplicity and demonstrated ability to separate distinct factors in the latent representation (known as *disentanglement*, although disentanglement is not a requirement for consistency and coherence). We therefore take the Evidence Lower Bound (ELBO) objective with an additional β scalar hyperparameter from Higgins2017-BVL, that seeks to achieve disentanglement ($L^{\text{ELBO}}_{\beta\text{-VAE}}$), and combine it with $L^{\tilde{\mathcal{S}}_\sigma}$ to obtain the following aggregate objective:[6]

$$L^{\text{joint}} = L^{\text{ELBO}}_{\beta\text{-VAE}} - \lambda L^{\tilde{\mathcal{S}}_\sigma} \tag{11}$$

where λ is a scalar weighting parameter.

Together with the $L^{\text{ELBO}}_{\beta\text{-VAE}}$, the choice of relation-decoder can shape the domain encodings Gutierrez-Basulto2018-FKG. In our evaluation, the following choices are made. We propose a Dynamic Comparator (DC) composed of two modes, a distance-based measure, $\phi^\dagger_r$, to measure the distance between two inputs relative to a reference point, and a step-function, $\phi^\ddagger_r$, that determines the sign of the difference between two points, optionally with an offset. Although any function could be used that has the required characteristics for $\phi^\dagger$ and $\phi^\ddagger$, in this paper we use the following implementation:

$$\phi^{DC}_r(\boldsymbol{z}_i, \boldsymbol{z}_j) = a_{r,0} \cdot \phi^\dagger_r + a_{r,1} \cdot \phi^\ddagger_r \tag{12}$$

where,

$$\phi^\dagger_r = f_0\big(-\eta_{r,0} \cdot \|\boldsymbol{u}_r \odot (\boldsymbol{z}_i - \boldsymbol{z}_j + \boldsymbol{b}^\dagger_r)\|_2\big) \tag{13}$$

$$\phi^\ddagger_r = f_1\big(\eta_{r,1} \cdot \boldsymbol{u}_r^\top(\boldsymbol{z}_i - \boldsymbol{z}_j + \boldsymbol{b}^\ddagger_r)\big). \tag{14}$$

[6] a more detailed derivation of $L^{\text{ELBO}}_{\beta\text{-VAE}}$ is included in the Appendix C

Here, $\boldsymbol{a}_r = \texttt{Softmax}(\boldsymbol{A}_r) \in (0,1)^2$ is an attention weighting between the two modes, $\phi_r^\dagger$ and $\phi_r^\ddagger$; f_0 and f_1 are an exponential and sigmoid function, respectively; $\boldsymbol{u}_r = \texttt{Softmax}(\boldsymbol{U}_r) \in (0,1)^m$ is an attention mask which is applied to m-dimensional embeddings; $\boldsymbol{b}_r^\dagger, \boldsymbol{b}_r^\ddagger \in \mathbb{R}^m$ are learnable bias terms that enable an offset to each mode; $\eta_{r,0} \in \mathbb{R}^+$ are non-negative and $\eta_{r,1} \in \mathbb{R}$ are any-valued scalar terms, respectively. Lastly, $\odot$ denotes the Hadamard product and $\|\cdot\|_2$ is the Euclidean norm. The key innovation behind DC is its ability to model each of the ordinality relations whilst encouraging generalized consistency across the full latent subspace, as defined by each $\boldsymbol{u}_r$. This is achieved without explicit weight sharing, wherein relation-decoders discover parametric relationships from the data. Further details are provided in Appendix D.1.

6 Experiment Design: Partial Relation Transfer

We now describe an experimental design to compare coherence of different relation-decoders.

Partial Relation Transfer (PRT): We evaluate a novel PRT task across two soft-structures $\tilde{\mathcal{X}}_\sigma$ and $\tilde{\mathcal{Y}}_\sigma$. The soft-structures share a common signature σ and relation-decoders ϕ, but have disjoint domains $\mathcal{X}$ and $\mathcal{Y}$, respectively. The experimental design involves first learning ϕ on source domain $\mathcal{X}$, together with its domain-specific autoencoder. Then, a new domain-specific autoencoder is trained on the target domain $\mathcal{Y}$, alongside a selection of the now learned ϕ relation-decoders with fixed-parameters. The selection of relation-decoders is expected to help guide training of $\psi_{\mathcal{Y}}^{\text{enc}}$ (see Fig.1). Held-out relation-decoders are then evaluated in $\mathcal{Y}$, i.e. a zero-shot transfer learning task. For domain $\mathcal{X}$ we use the MNIST handwritten digits data set mnist, and for domain $\mathcal{Y}$ we use the proposed BlockStacks data set, consisting of a single stack of multi-colored cubes of differing heights, each containing one randomly-positioned red cube (see Appendix B for details and examples). The shared signature includes the ordinal relations $\sigma = \{\mathsf{G}, \mathsf{E}, \mathsf{L}, \mathsf{S}, \mathsf{P}\}$, and it is applied to digit ordering in MNIST and to red cube position ordering in BlockStacks. We provide results with respect to a theory of ordinality, as explored in Example 1. A formal specification of the theory is provided in Appendix F. When transferring relations from $\psi_{\mathcal{X}}^{\text{enc}}$ to $\psi_{\mathcal{Y}}^{\text{enc}}$, one could use the full set ϕ of relation-decoders. However, this is not necessary from a logical standpoint because the entire system of relations can be expressed in terms of the $\mathsf{isSuccessor}$ relation S (e.g. the successor of a number is larger than that number). We therefore only employ the $\mathsf{isSuccessor}$ relation-decoder as the fixed-parameter selection to guide the learning of $\psi_{\mathcal{Y}}^{\text{enc}}$. If coherence, as defined in this paper, is carried across domains, we would expect the transferring of $\mathsf{isSuccessor}$ to produce an improved performance on the remaining relations in the target domain.

Neural model components and hyperparameters: Together with DC, existing relation-decoder models evaluated here are: TransR Lin2015-LEA, HolE Nickel2016-HEO, NTN Socher2013-RWN. We additionally include a basic feed-forward neural network (NN). To produce domain-encodings, all experiments use

a β-VAE Higgins2017-BVL. We provide further details for all models, including training regimen, parameterization and implementation in Appendix D. In the source domain, we explore β values in $\{1, 4, 8, 12\}$ and set $\lambda = 10^3$. In the target domain, we first normalise losses and set $\beta = 10^{-4}$ and $\lambda = 10^{-2}$, as these produced good image reconstructions while optimising $L^{\mathcal{Y}_\sigma}$. In all experiments we fix $\mathcal{Z} = \mathbb{R}^{10}$.

7 Experimental Results and Discussion

In this section, experimental results show that transfer learning performance is positively correlated with our measures for consistency and coherence. This holds particularly true for embeddings that are different but near in space to source domain embeddings. As we have argued, for a neural model to perform well on concept transfer, its representations must maintain high probability of consistency with a theory that provides a semantics for the concept. The most robust way of doing this is to maintain consistency across regions of embedding space, rather than relying exclusively on the specific data-points observed at training time in the source domain. In our analysis, consistency losses are evaluated when sampling from different regions of latent space $\mathcal{Z}$. We evaluate: *data-embeddings*, where all inputs are encodings of a domain's test data; *interpolation*, when we derive an empirical mean and variance for the domain's data-embeddings and sample from a corresponding Gaussian distribution; and *extrapolation*, when we sample from regions strictly outside the smallest, axis-aligned, hyper-rectangle that encloses all data-points.

Figure 2-top provides relation-decoder prediction accuracies in both the source (MNIST, left) and target (BlockStacks, right) domains.[7] The relations are S, P, E, G, L and relation S is transferred to the target domain. Key observations are that DC produces excellent PRT performance, whilst NN, NTN and HolE all see some degradation from their source-domain accuracies for relations other than isSuccessor (S). TransR maintains target-domain accuracies similar to its performance in the source domain, but this is significantly below the performance of other models in the source domain. We include the impact of adjusting β (disentanglement pressure) in Figure 2-bottom. Barring DC which has little discernible change in either source or target domains, PRT performance is significantly impacted by β for all models in the target domain, but has little effect in the source domain. TransR shows a strong positive correlation between target domain accuracy and β values, whereas the remaining models produce their best PRT performance with medium disentanglement pressure.

To investigate the broad trends that run across all β values and relation-decoder models, we ran a Spearman rank correlation analysis between consistency losses and PRT performance. Separate coefficients are produced for each combination of consistency loss: Con-A and Con-I further divided into Con-I-T (transitivity), Con-I-A (asymmetry) and Con-I-R (reflexivity), and regions of

[7] We take ϕ_r inferences of 0.5 or above to signify *true*, and otherwise *false*. An alternative, left as future work, would be to sample the space of ϕ values to produce a confidence measure.

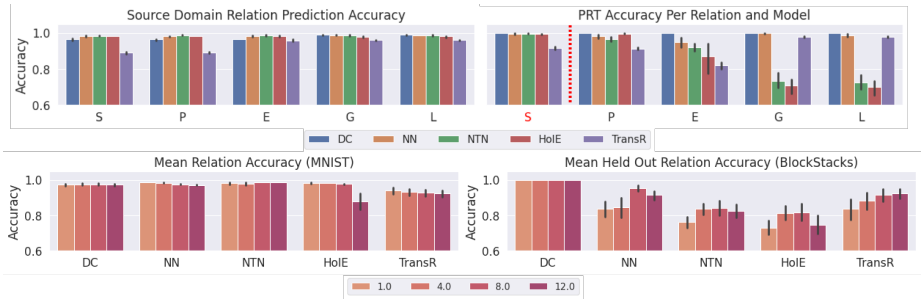

Figure 2: **[Top]** Relation-decoder prediction accuracy per model (DC, NN, NTN, HoIE, TransR) and relation (abbreviated on the x-axis as in Example 1), in the source domain (MNIST, left) and target domain (BlockStack, right). A red highlighted S and dotted line (top right) indicates that relation isSuccessor is included in training the target domain autoencoder, but none of the other relations are. Both DC and NN retain a good performance while all other models show a decrease of accuracy in the target domain for one or more of the relations not included in training. **[Bottom]** Impact of different values of $\beta \in \{1, 4, 8, 12\}$ for each relation-decoder averaged across all relations in the source domain (left) and held-out relations $\{\mathsf{P}, \mathsf{E}, \mathsf{G}, \mathsf{L}\}$ in the target domain (right). It can be seen that DC is not impacted by changes in β and it maintains performance in the target domain. All other models show a decrease of accuracy for the held-out relations in the target domain.

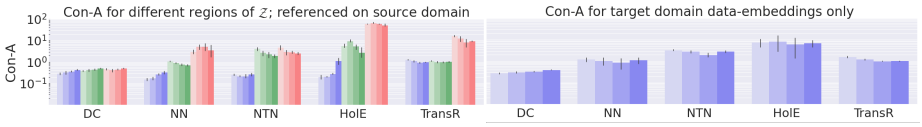

Figure 3: Consistency-Across (Con-A) losses (lower values are better) for the models (DC, NN, NTN, HolE, TransR) using the MNIST data set (source domain $\mathcal{X}$) **[left]** and BlockStacks (target domain $\mathcal{Y}$) **[right]**. The blue bars show the consistency loss of the data embeddings, with darker shades corresponding to models trained with higher β (disentanglement pressure). The green bars show the results for interpolation. The red bars show the results for extrapolation.

latent space: data-embedding, interpolation and extrapolation (see Appendix H for the full table). Lower consistency losses are expected to produce higher PRT performance as indicated by a negative Spearman rank coefficient. The coefficients show that the consistency losses of data-embeddings in the source domain are weakly rank correlated with PRT. The consistency losses in the case of interpolation are, in most cases, strongly rank correlated with PRT. The consistency losses in the case of extrapolation lie in between and are generally moderately rank correlated with PRT. This supports our thesis that consistency can facilitate reliable transfer. Furthermore, consistency of certain partial theories may matter more. Here, Con-A, Con-I-T and Con-I-A on interpolation are the most relevant partial theories for transfer learning performance. As we shall see, DC outperforms all other models on these losses and this result is mirrored by its PRT performance.

To gain a deeper insight as to which underlying characteristics can explain the observed PRT accuracy profiles, Figure 3 and Figure 4 present Con-A and Con-I loss profiles, respectively, for varied β and regions of latent space (for data-embeddings in blue, interpolation in green and extrapolation in red). Results refer to both source (left) and target domain embeddings (right). Firstly, we note that DC retains excellent Con-A for all regions of latent space. TransR retains consistency from data-embeddings to the interpolated regions, but not to the extrapolated regions. The remaining models show degradation of consistency between data-embeddings and interpolation and extrapolation regions, with extrapolation often being worse than interpolation. Looking at β trends, aside from DC, increasing β appears to have a positive but limited effect on interpolation and extrapolation performance. Considering the Con-A performance of data-embeddings in the target domain, DC shows the best performance. The Con-A performance in the target domain is in agreement with the PRT accuracies. For all the models, Con-A performance in the target domain appears to match the interpolation or extrapolation Con-A performance in the source domain. This points to the possibility of anticipating transfer learning performance by evaluating the consistency of partial theories.

Many of the same trends can be seen in the results for Con-I (Con-I-T, Con-I-A and Con-I-R) in Figure 4. Results are averaged over individual relations. As

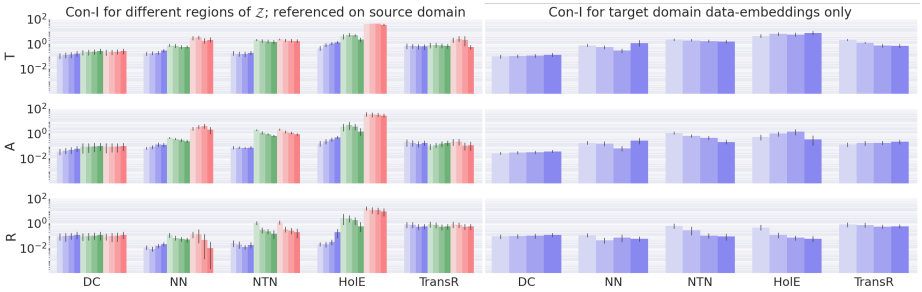

Figure 4: Consistency-Individual (Con-I) losses (lower values are better) for the models (DC, NN, NTN, HolE, TransR) using the MNIST data set (source domain $\mathcal{X}$) **[left]** and BlockStacks (target domain $\mathcal{Y}$) **[right]**. From top to bottom (following the same colour schematic as Figure 3): Con-I-T (**T**ransitivity), Con-I-A (**A**symmetry) and Con-I-R (**R**eflexivity).

Table 1: Coherence comparison with respect to source and target data-embeddings. Results are reported with the corresponding $\beta = \beta^*$ value (in parenthesis). The consistency loss abbreviations refer to: (A)cross, T(ransitivity), A(symmetry), R(eflexivity) and Aggr(egate), which gives the best obtained aggregate consistencies. DC outperforms all other approaches across most coherence scores.

ϕ	Aggr.	(β^*)	Con-A	(β^*)	Con-I-T	(β^*)	Con-I-A	(β^*)	Con-I-R	(β^*)
HolE	12.12	(1)	6.61	(8)	4.30	(1)	0.52	(12)	0.08	(8)
NTN	4.11	(8)	1.92	(8)	1.50	(12)	0.22	(12)	0.09	(12)
TransR	2.51	(8)	1.02	(12)	0.71	(12)	0.18	(4)	0.55	(8)
NN	1.71	(8)	0.82	(8)	0.44	(12)	0.18	(4)	**0.05**	(4)
DC	**0.53**	(1)	**0.29**	(1)	**0.11**	(1)	**0.04**	(1)	0.09	(1)

in Figure 3, results are presented with respect to source domain (left) and target domain (right).We firstly observe that DC and NN share the best overall Con-I performance profiles, with TransR following closely. DC and TransR again show comparable data-embedding versus interpolation/extrapolation performance, whereas NN, NTN and HolE suffer from degradation. With regards to β's impact, DC is not affected by β, while NN and NTN show a negative correlation between β and Con-I losses with comparable results for each underlying partial theory.

Finally, Table 1 provides a comparison between optimal coherences achieved for each relation-decoder model, as defined in Section 4. Results are partitioned according to each consistency type and aggregate value. DC clearly outperforms all other models on coherence. NN achieves strong aggregate coherence, followed by TransR and NTN, with HolE performing generally worse. Looking at β^*

profiles, we see that most models achieve optimum aggregate coherence at $\beta = 8$, apart from DC and HolE which perform better at $\beta = 1$. Overall, this is in broad agreement with the β profiles given by Figure 2-bottom (right). However, we can see that β^* profiles for Con-A coherence are in more direct agreement as TransR achieves its best at $\beta = 12$ and HolE at $\beta = 8$. This suggests that Con-A is more indicative of PRT performance, which is to be expected since PRT relies on inductive transfer across relations.

All in all, these results paint a picture where source domain accuracy alone is not a strong enough indicator of concept transfer. Instead, it may be possible to anticipate transfer performance by evaluating consistency in regions beyond the source domain's data-embeddings. Depending on the task at hand, certain partial theories may be more relevant than others in this analysis.

8 Related Work

Relational representations play a prominent role in Knowledge Graph Embedding (KGE), wherein sets of relation-decoders are jointly learned to obtain a semantic latent representation from data Socher2013-RWN, Trouillon2016-CEF, Trouillon2019-OIA, Bordes2013-TEM, Nickel2016-ARO, Wang2017-KGE, Dai2020-ASO, Kazemi2018-SEL, Abboud2020-BAE, Serafini2016-LTN, Donadello2017-LTN, Badreddine2022-LTN. Although KGE approaches typically do not use a shared autoencoder as done in this paper, in Schlichtkrull2018-MRD an autoencoding framework is adopted, where a graph neural network is used as the encoder. However, Schlichtkrull2018-MRD did not work with visual data and the model was only applied to single data sets rather than transfer learning. Similarly, disentanglement is concerned with semantic representation learning Bengio2013-RLR, and it has been explored using a variety of methods including both Generative Adversarial Networks Chen2016-IIR and VAEs Burgess2017-UDI, Higgins2017-BVL, Chen2018-ISD, Ridgeway2018-LDD, Eastwood2018-FQE, Kumar2018-VID, Locatello2019-CCA. Disentangled representations have been evaluated on their transferability Steenkiste2019-ADR, Steenbrugge2018-IGA, Locatello2020-WSD. A bridge between these two fields, with relation-decoders employed in the semi-supervision of VAEs, can be found in Karaletsos2016-WCH, Chen2019-WSD, Chen2019-ROV. In Karaletsos2016-WCH, multiple relation-decoders are used, but to compute a triplet comparison-based query. In Chen2019-WSD, Chen2019-ROV, only a single binary relation is studied using functional forms that are not sufficient to model the full set of relations considered in this paper. Lastly, we note that our experimental setup is most remnant of domain adaptation, e.g. Redko2019-AID. To the best of our knowledge, this paper is the first to present a comprehensive analysis of the resulting concept coherence. No previous work has compared relation-decoders on their ability to learn consistently and coherently, as measured in this paper.

9 Conclusion and Future Work

This paper introduced formal definitions of consistency and coherence for representation learning. As a result, a sub-symbolic model can have its consistency and coherence measured with respect to a logical theory. The paper specified a neuro-symbolic model based on domain-encoders coupled with modular relation-decoders, and an experimental procedure that, together, allowed for the investigation of how concept coherence differs for various implementations of relation-decoders applied to transfer learning. Finally, consistency and coherence results showed that the models that can retain consistency (*i.e.* be coherent) across regions of latent space beyond the source data-embeddings are more likely to perform better at PRT learning tasks. The empirical evaluations in this paper only considered binary relations and a fixed signature which is learned "all at once" in a source domain. In practical applications, however, it should be possible to discover concepts gradually, e.g. as part of a curriculum and through gradual refinement of pre-learned relations after exposure to different contexts. This necessitates an adaptation of the approach presented here and further evaluations, as part of future work. Further evaluations of the formalization introduced here should consider the use of different models, theories (such as specifying periodic, *e.g.* rotation, and unordered categorical, *e.g.* shape, properties) and scenarios/data sets in the evaluation of consistency and coherence of neural models.

References

[1] Stewart Shapiro and Teresa Kouri Kissel. Classical Logic. In *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, Spring 2021 edition, 2021.

A Societal Impact Statement

This work does not have a negative societal impact, specifically it does not include any of the following: involvement of human subjects, sensitive data, harmful insights, methodologies and applications. The results, data sets and methodologies are objectively non-discriminatory, unbiased and fair. This work does not breach any privacy or security guidelines or laws, nor any other legal restrictions.

The proposed definition of coherent concepts and corresponding analysis provides more depth in the assessment of deep learning methods, which are typically otherwise opaque, and this can have a positive societal impact. Currently, we cannot provide interpretable descriptions regarding *how* a standard deep learning method produces its inferences, making it difficult to fully trust a model in critical applications. An important failure case is that biases are not easy to uncover from a trained deep learning model. The benefit of learning a coherent concept is that inferences uphold logical consistency, which can be formally expressed and tested. This can provide more trust in the model as practitioners can have confidence that the model should not obtain inputs that lead to incoherent inferences, wherein errors are certain. Further, if the logic does not include biases, the inferences of a coherent set of relation-decoders should not be biased. A caveat to these points is that unless the relation-decoder functional form allows us to analytically make comments/assertions about the model's performances for arbitrary regions of latent space, as with DC (see D.1), it is intractable to fully examine model coherence, as it requires a full extrapolation/interpolation evaluation. Nonetheless, a practical evaluation of coherence is an important step forward.

B BlockStacks dataset description

The BlockStacks data set consists of 12,000 RGB images (3×200×200 pixels but resized in code to $3 \times 128 \times 128$) of individual block stacks, of varying height (between 1-10 blocks), block colors (uniformly sampled from options: {gray, blue, green, brown, purple, cyan, yellow}) and position (uniformly sampled from x, y range (-3,-3) to (3,3)), but with the requirement that each instance consists of a single red block at a random height (see Figure 5 for example images). These were rendered using the CLEVR rendering agent with the help of code from CLEVRRendererASAI. The dataset is divided into 9000:1500:1500 train, validation and test splits.

C Explanation of the β-VAE

The VAE is derived by introducing an approximate posterior $q_\alpha(\boldsymbol{Z}|\boldsymbol{X})$, from which a lower bound (commonly referred to as the Evidence LOwer Bound (ELBO)) on the true marginal $\log p_\theta(\boldsymbol{X})$ can be obtained by using Jensen's

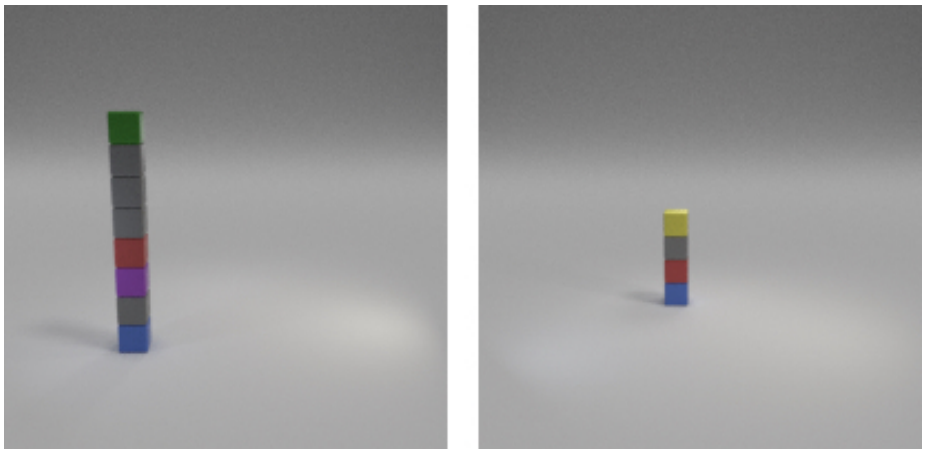

Figure 5: Example of two BlockStacks data set images. Each instance consists of a single red block varying in position within the block stack. On the left the red block is at height 3 (using a zero index) and on the right it is at height 1.

inequality Kingma2014-AEV. The VAE maximises the log-probability by maximising this lower bound, given by:

$$L^{\text{ELBO}}_{\beta\text{-VAE}} = \mathbb{E}_{q_\alpha(\boldsymbol{Z}|\boldsymbol{X})}[\log p_\theta(\boldsymbol{X}|\boldsymbol{Z})] - \beta D_{KL}(q_\alpha(\boldsymbol{Z}|\boldsymbol{X})\|p_\theta(\boldsymbol{Z})), \tag{15}$$

where $q_\alpha(\boldsymbol{Z}|\boldsymbol{X})$ is typically modelled as a neural-network encoder with parameters α. Similarly $p_\theta(\boldsymbol{X}|\boldsymbol{Z})$ is often modelled as a neural-network decoder with parameters θ and is calculated as a Monte Carlo estimation. A reparameterization trick is used to enable differentiation through an otherwise undifferentiable sampling from $q_\alpha(\boldsymbol{Z}|\boldsymbol{X})$ (see Kingma2014-AEV). In the β-VAE Higgins2017-BVL, Burgess2017-UDI, an additional β scalar hyperparameter was added as it was found to influence disentanglement through stronger distribution matching pressure with respect to the prior $p_\theta(\boldsymbol{Z})$, where this prior is typically set to an isotropic zero-mean Gaussian $\mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$). When $\beta = 1$ we obtain the standard VAE objective Kingma2014-AEV.

D Model Descriptions

In this section we firstly present an in-depth analysis of the key innovations presented by DC which provides insight into how it can learn a coherent notion of ordinality. We then provide model details for each of the compared relation-decoders in the main results and the backbone β-VAE architecture that we employ for each data set.

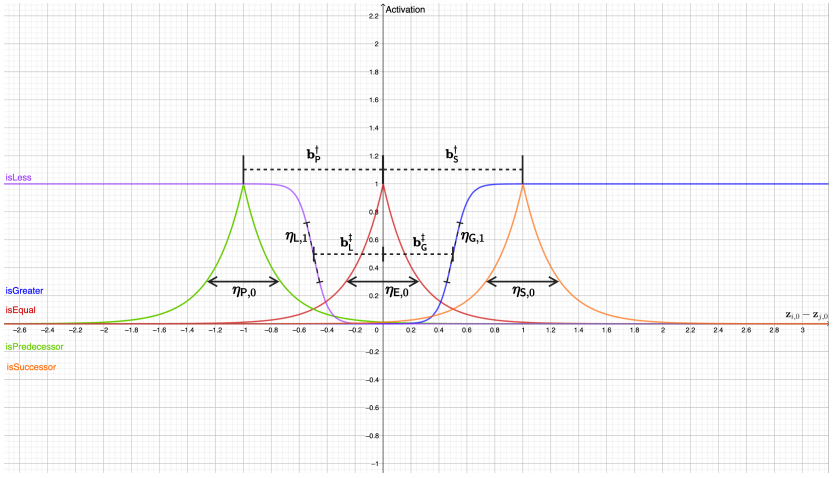

Figure 6: Depiction of a set of DC relation-decoders for binary relations isGreater, isLess, isEqual, isSuccessor and isPredecessor. Each DC relation-decoder (for each relation) shown here has a one-hot mask, $\boldsymbol{u}_r$, that is in this example the same across relations, which ensures only the zeroth dimensions of the embedding arguments are compared, giving $\boldsymbol{z}_{i,0}$ and $\boldsymbol{z}_{j,0}$.

D.1 Dynamic Comparator Analysis

Figure 6 depicts how DC is able to learn the isGreater, isLess, isEqual, isSuccessor and isPredecessor family of binary ordinal relations, assuming each corresponding relation-decoder has learned a common one-hot mask on the zeroth dimension *i.e.* $\boldsymbol{u}_{\mathsf{G}} = \boldsymbol{u}_{\mathsf{E}} = \ldots = \boldsymbol{u}_{\mathsf{P}} = [1, 0, \ldots, 0]$, such that activations only depend on the $\boldsymbol{z}_{i,0} - \boldsymbol{z}_{i,1}$ difference. An important capability of DC is its ability to dynamically *select* via $\boldsymbol{a}_r$ an appropriate functional mode, either $\phi_r^\dagger$ or $\phi_r^\ddagger$, depending on the type of relation it needs to model. As shown by Figure 6, this allows isEqual to exhibit its reflexive, symmetric and transitive characteristics, whilst isGreater and isLess both carry transitivity but are asymmetric and irreflexive. Furthermore, the use of a subtraction between $\boldsymbol{z}_i$ and $\boldsymbol{z}_j$ (which, via mask $\boldsymbol{u}$, ends up only being a subtraction between their zeroth dimensions) leads to a relative comparison, not an absolute comparison, which generalises to arbitrary $\boldsymbol{z}_i$ and $\boldsymbol{z}_j$ sampled from anywhere in $\mathcal{Z}$.

Note that there is no built in parameter sharing, meaning each relation-decoder (for each individual relation r) is trained independently and has its own set of $\boldsymbol{a}_r, \boldsymbol{u}_r, \eta_{r,0}, \eta_{r,1}, \boldsymbol{b}_r^\dagger$ and $\boldsymbol{b}_r^\ddagger$ parameters. However, our experiments show that DC reliably obtains settings such that *e.g.* $\boldsymbol{u}_{\mathsf{G}} = \boldsymbol{u}_{\mathsf{E}}$, or $\boldsymbol{a}_{\mathsf{G}} = \boldsymbol{a}_{\mathsf{L}} = [0, 1]$, or $\boldsymbol{b}_{\mathsf{G}}^\ddagger = -\boldsymbol{b}_{\mathsf{L}}^\ddagger$ and so on. DC is thus able to discover the interdependencies between families of relations. By learning to loosely 'tie' together parameters in this way, whilst still being expressive enough to model each type of relation, DC can facilitate a data-driven binding between relation-decoder outputs. This

helps ensure consistent generalisation across a latent subspace, as defined by the common/overlapped $\boldsymbol{u}_r$ masks.

D.2 Relation-Decoder implementations

TransR Lin2015-LEA:

$$\phi_r^{\text{TransR}}(\boldsymbol{z}_i, \boldsymbol{z}_j) = \|\boldsymbol{h}_r + \boldsymbol{r} - \boldsymbol{t}_r\|_2^2$$

with,

$$\boldsymbol{h}_r = \boldsymbol{M}_r \boldsymbol{z}_i \quad \text{and} \quad \boldsymbol{t}_r = \boldsymbol{M}_r \boldsymbol{z}_j.$$

where for $\boldsymbol{z}_i, \boldsymbol{z}_j \in \mathbb{R}^{d_z}$ vectors, $\boldsymbol{M}_r \in \mathbb{R}^{d_z \times d_z}$ and $\boldsymbol{r} \in \mathbb{R}^{d_z}$. As we want to obtain a $(0, 1)$ output, we modify TransR through $\phi_r^{\text{TransR}^+} = \sigma(c - \phi_r^{\text{TransR}})$, where σ is a sigmoid function and c is a scalar that ensures that at $\phi_r^{\text{TransR}}(\boldsymbol{z}_i, \boldsymbol{z}_j) = 0$, then $\phi_r^{\text{TransR}^+}(\boldsymbol{z}_i, \boldsymbol{z}_j) \approx 1$. In all experiments we set $c = 10$.

NTN (modified version of Socher2013-RWN from Donadello2017-LTN, Serafini2016-LTN):

$$\phi_r(\boldsymbol{z}_1, \ldots, \boldsymbol{z}_n) = \sigma\big(\boldsymbol{u}_r^\top [\tanh(\boldsymbol{z}^{c\top} \boldsymbol{M}_r \boldsymbol{z}^c + \boldsymbol{V}_r \boldsymbol{z}^c + \boldsymbol{b}_r)]\big) \tag{16}$$

where $\boldsymbol{u}_r \in \mathbb{R}^k, \boldsymbol{M}_r \in \mathbb{R}^{n \cdot d_z \times n \cdot d_z \times k}, \boldsymbol{V}_r \in \mathbb{R}^{k \times n \cdot d_z)}$ and $\boldsymbol{b}_r \in \mathbb{R}^k$. The only hyperparameter to consider is k, which controls the NTN's capacity - in all experiments, we set this to 1. If $k > 1$, $\boldsymbol{z}^{c\top} \boldsymbol{M}_r \boldsymbol{z}^c$ produces a k-dimension vector by applying the bilinear operation to each of the k $\boldsymbol{M}_r$ slices. Here $\boldsymbol{z}^c \in \mathbb{R}^{n \cdot d_z}$ is a concatenation of the inputs $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_n$, which was introduced in Donadello2017-LTN, Serafini2016-LTN. In contrast, the original NTN (see Socher2013-RWN) is only applicable to binary relations and does not include the outer sigmoid.

HolE Nickel2016-HEO:

$$\phi_r^{\text{HolE}}(\boldsymbol{z}_i, \boldsymbol{z}_j) = \sigma(\boldsymbol{r}^\top (\boldsymbol{z}_i \star \boldsymbol{z}_j))$$

where $\boldsymbol{r} \in \mathbb{R}^{d_z}$ and $\star : \mathbb{R}^{d_z} \times \mathbb{R}^{d_z} \to \mathbb{R}^{d_z}$ denotes the circular correlation operator and is given by,

$$[\boldsymbol{z}_i \star \boldsymbol{z}_j]_k = \sum_{m=0}^{d-1} z_{i,m} z_{j,(k+m) \mod d}$$

NN: a simple four-layer neural-network with layer sizes $l_{\text{in}} = 2d_z, l_1 = 2d_z$ and $l_2 = d_z$, with ReLU activations Nair2010-RLU. The final output layer, l_{out}, is a single value passed through a sigmoid function, to bound the output within $(0, 1)$.

D.3 β-VAE configuration

The model configurations used for both MNIST and BlockStacks data sets are given in Table 2.

D.4 L^{joint} configuration

In the source domain, we vary β values between $\{1, 4, 8, 12\}$ and fix $\lambda = 10^3$. In the target domain, we fix β to 10^{-4} and $\lambda = 10^{-2}$ and normalise the $\mathcal{L}^{ELBO}_{\beta\text{-VAE}}$ reconstruction term by dividing by a factor $\frac{1}{\sqrt{H \cdot W \cdot C}}$, for height H, width W and color channels C, and normalize the distribution matching term by a factor $\frac{1}{d_z}$, for latent representation size d_z (set to 10 across all experiments).

To train relation-decoders over a given domain $\mathcal{S}$, it is necessary to supervise estimates of $\phi_r(\psi^{enc}_{\mathcal{S}}(O)), O \in \mathcal{S}^2$, against corresponding ground-truth labels, $\gamma^r_{O,\mathcal{S}_\sigma}$. However, doing so for every $O \in \mathcal{S}^2$ can easily become intractable and we instead only sample a subset of possible $\mathcal{S}^2$ tuples. Our sampling strategy involves first selecting a ratio $R = \frac{|\mathcal{B}|}{|\mathcal{S}|}$ where $\mathcal{B} \subset \mathcal{S}^2$ is a set of O tuples. We then sample relation-decoder specific subsets $\mathcal{B}_r$ where $|\mathcal{B}_r| = \frac{|\mathcal{B}|}{|\sigma|}$, to ensure a balanced distribution of tuples between relation-decoders. Furthermore, we ensure that each $\mathcal{B}_r$ contains a balanced ratio of $\gamma^r_{O,\mathcal{S}_\sigma} = 1$ versus $\gamma^r_{O,\mathcal{S}_\sigma} = 0$ instances. We found that each $|\mathcal{B}_r|$ set can be small without jeopardising the final relation-decoder performance level, allowing us to use $R = 1$ for MNIST experiments and $R = 3$ for BlockStacks experiments.

Finally, in all experiments we use a β-VAE trained for up to 300,000 steps, following accepted practice from Locatello2019-CCA, Steenbrugge2018-IGA, together with any included relation-decoders. However, to ensure computation efficiency across experiments, we employ an early stopping procedure, where if the validation score does not increase over 30 and 120 training epochs for MNIST and Blockstacks experiments, respectively, we end the training early.

E Preliminaries in further detail

Logic and model-theoretic background: to support Section 2 we provide additional logic and model theoretic background. In this paper, we assume a formal language $\mathcal{L}$ composed of variables, predicates (i.e. relations), logical connectives $\neg$ (negation), $\vee$ (disjunction), $\wedge$ (conjunction), $\rightarrow$ (implication), and universal quantification $\forall$ (for all) with their conventional meaning (see [1]). The set of relations in $\mathcal{L}$ form the *signature*, σ, of the language. Relations have an associated arity, denoted as $\mathsf{ar}(\cdot)$, that defines the number of arguments they take. For example, a binary relation r has arity $\mathsf{ar}(r) = 2$. Relations are used to express knowledge over the elements of a *domain* $\mathcal{S}$, where $\mathcal{S}$ is a non-empty set. For instance, $r(s_1, s_2)$ states that elements s_1 and s_2 are related through the binary relation r. The meaning of a relation is defined by an *interpretation* $I_{\mathcal{S}_\sigma}$ which captures the $\{T, F\}$ (true or false) values of the relation over elements of $\mathcal{S}$. Together, a domain $\mathcal{S}$ and an interpretation $I_{\mathcal{S}_\sigma}$ of a given signature σ form a *structure* $\mathcal{S}_\sigma = (\mathcal{S}, I_{\mathcal{S}_\sigma})$.

Note that for a fixed domain $\mathcal{S}$ and signature σ, different interpretations yield different structures. As stated in the main text, we construct universally quantified first-order formulae (called sentences) using the signature σ of $\mathcal{L}$,

whose truth-value is defined with respect to a given structure $\mathcal{S}_\sigma$. To do so, we first consider *ground* instances of a formula. These are given by replacing all the variables in the formula with elements from the domain $\mathcal{S}$. For example, $r(s_1, s_2)$, where s_1 and s_2 are elements of $\mathcal{S}$, is a *ground* instance of an atomic formula $r(i, j)$ where i and j are variables in $\mathcal{L}$. Given a structure $\mathcal{S}_\sigma = (\mathcal{S}, I_{\mathcal{S}_\sigma})$, a relation r, and a tuple $(s_1, \ldots, s_{\mathsf{ar}(r)}) \in \mathcal{S}^{\mathsf{ar}(r)}$, a ground instance $r(s_1, \ldots, s_{\mathsf{ar}(r)})$ is true in the structure $\mathcal{S}_\sigma$ if and only if $(s_1, \ldots, s_{\mathsf{ar}(r)}) \in I_{\mathcal{S}_\sigma}(r)$. The truth value of a sentence in a given structure $\mathcal{S}_\sigma$ depends on the truth value of its respective ground instances. Specifically, a sentence is true in a structure $\mathcal{S}_\sigma$ if and only if all of its ground instances are true in $\mathcal{S}_\sigma$. For example, $\forall i.\, r(i, i)$ is true in $\mathcal{S}_\sigma$ if and only if all of its ground instances $r(s_h, s_h)$ are true in $\mathcal{S}_\sigma$, for every $s_h \in \mathcal{S}$. When a sentence, τ, is true in a structure, $\mathcal{S}_\sigma$, we say that the structure *satisfies* τ, denoted as $\mathcal{S}_\sigma \models \tau$. A set of sentences form a *theory*, $\mathcal{T}$ and any subset of the sentences in $\mathcal{T}$ form a partial theory with respect to $\mathcal{T}$. A theory can be seen as a way of constraining the type of interpretations that we want to "accept" for our signature. Finally, a *model* of $\mathcal{T}$ is a structure that satisfies every sentence in $\mathcal{T}$.

F Specification for theory of ordinality

To support our claim that we can use only the $\mathsf{isSuccessor}$ relation as the target encoder guide due to its logical relationship with the remaining relations, we include here the logical clauses:

$$
\begin{aligned}
\forall i, j, k.\ \ (\mathsf{isSuccessor}(i, j) \wedge \mathsf{isSuccessor}(k, j) &\rightarrow \mathsf{isEqual}(i, k)) \\
\forall i, j.\ \ (\mathsf{isSuccessor}(i, j) &\rightarrow \mathsf{isGreater}(i, j)) \\
\forall i, j, k.\ \ (\mathsf{isSuccessor}(i, j) \wedge \mathsf{isGreater}(j, k) &\rightarrow \mathsf{isGreater}(i, k)) \\
\forall i, j.\ \ (\mathsf{isSuccessor}(i, j) &\leftrightarrow \mathsf{isPredecessor}(j, i)) \\
\forall i, j.\ \ (\mathsf{isPredecessor}(i, j) &\rightarrow \mathsf{isLess}(i, j)) \\
\forall i, j, k.\ \ (\mathsf{isPredecessor}(i, j) \wedge \mathsf{isLess}(j, k) &\rightarrow \mathsf{isLess}(i, k)).
\end{aligned}
$$

Therefore, by knowing all of the successor relations between data instances, it should be possible to infer the remaining relationships that they share.

For completeness, we provide the truth tables for each of the sub-theories that our consistency losses evaluate against. We only include configurations that are valid under the constraints, indicated by $\subset \mathcal{T} = T$, where this notation highlights the fact each incomplete set of constraints form a subset of the overall theory $\mathcal{T}$.

Firstly, the truth-table that describes constraints shared between relation

truth-values is given by the following, $\forall i, j$:

$\mathsf{G}(i,j)$	$\mathsf{E}(i,j)$	$\mathsf{L}(i,j)$	$\mathsf{S}(i,j)$	$\mathsf{P}(i,j)$	$\subset \mathcal{T}$
T	F	F	F	F	T
T	F	F	T	F	T
F	T	F	F	F	T
F	F	T	F	F	T
F	F	T	F	T	T

where we use the same relation abbreviations as in the main text results.

Next, we provide each of the three consistency individual (Con-I) truth-tables. These are referred to as being "individual" due to the fact that they describe constraints applied to the truth-state of a single relation. For transitivity, given by the rule *e.g.* $\mathsf{G}(i,j) \wedge \mathsf{G}(j,k) \rightarrow \mathsf{G}(i,k)$, we have that $\forall i, j$:

$\mathsf{G}(i,j)$	$\mathsf{G}(j,k)$	$\mathsf{G}(i,k)$	$\subset \mathcal{T}$
F	F	F	T
F	F	T	T
T	F	F	T
T	F	T	T
F	T	F	T
F	T	T	T
T	T	T	T

(17)

For asymmetry, where $\mathsf{S}(i,j) \rightarrow \neg\mathsf{S}(j,i)$, we have $\forall i, j$:

$\mathsf{S}(i,j)$	$\mathsf{S}(j,i)$	$\subset \mathcal{T}$
F	F	T
T	F	T
F	T	T

(18)

.

Finally, for reflexivity, given by $\mathsf{E}(i,i) \rightarrow \top$ (in this case describing that an object is always equal to itself) we have $\forall i$:

$\mathsf{E}(i,i)$	$\subset \mathcal{T}$
T	T

(19)

Truth-table matrices for each of the above truth-tables can be obtained by replacing T with 1 and F with 0. The full set of individual constraints that are applicable to each relation covered in this paper are given by Table 3.

G Expanded consistency loss derivation

In this section, we present the expanded justification for reporting $-\ln 1 - \bar{\epsilon}$ consistency and coherence as a proxy for ϵ-consistency/coherence as defined in Section 3. For notational clarity, in the following we omit $\psi_{\mathcal{S}}$, such that $\phi_r(\psi_{\mathcal{S}}(O))$ is abbreviated to $\phi_r(O)$.

In the following, we make no assumptions about the sizes of domain $\mathcal{S}$, signature σ and arities of each $r \in \sigma$. Further, we take $\mathcal{T}$ to be an arbitrary theory over σ consisting of universally quantified formula, and the validity of each ground instances of atomic formula with respect to $\mathcal{T}$, can be expressed by a single ground truth-table matrix, $\mathbf{T} \in \{0,1\}^{K_0 \times K_1 \times K_2}$, wherein each slice, $\mathbf{T}_{k,:,:}$ gives a unique grounding of domain objects to the variables, v, required by $\mathcal{T}$. For each grounding of the $K_0 = |\mathcal{S}|^{|v|}$ possible groundings, there are $K_1 = 2^l$ unique truth-assignments to the l atomic formulae that constitute $\mathcal{T}$, giving $K_2 = l + 1$ assignments per $\mathbf{T}_{k,t,:}$ row - one per atomic formulae and an additional value that denote whether the particular row satisfies $\mathcal{T}$. $\boldsymbol{T}$ can be obtained by taking any truth-table from the previous section and switching true (T) for 1 and false (F) for 0, and producing K_0 copies for each assignment of domain elements to the variables. Given this truth-table matrix, notice that a structure $\mathcal{S}_\sigma$ can be composed by selecting a single row of $\boldsymbol{T}$ for each grounding (kth slice), giving a vector $\boldsymbol{c}_{kt} = \boldsymbol{T}_{k,t,1:l}$. If the structure is a model of $\mathcal{T}$, *i.e.* $\mathcal{S}_\sigma \in \mathcal{M}_\mathcal{S}^\mathcal{T}$, then only rows with $\boldsymbol{T}_{k,t,K_2} = 1$ are allowed. Taking t^+ to be the set of rows such that $\boldsymbol{T}_{k,t,K_2} = 1$ (which is identical for each k) *i.e.* $t^+ = \{\ t\ |\boldsymbol{T}_{k,t,K_2} = 1 \wedge t \in \{1,\dots,K_1\}\}$, we can then rewrite $\Gamma_\mathcal{T}^{\tilde{\mathcal{S}}_\sigma}$ in terms of samples from $\boldsymbol{T}$:

$$\Gamma_\mathcal{T}^{\tilde{\mathcal{S}}_\sigma} = \sum_{\mathcal{S}_\sigma \in \mathcal{M}_\mathcal{S}^\mathcal{T}} \prod_{r \in \sigma} \prod_{O \in \mathcal{S}^{\mathsf{ar}(r)}} \phi_r(O)^{\gamma^r_{O,\mathcal{S}_\sigma}} (1 - \phi_r(O))^{1-\gamma^r_{O,\mathcal{S}_\sigma}} \qquad \text{(Eqn. 3)}$$

$$= \sum_{\mathcal{S}_\sigma \in \mathcal{M}_\mathcal{S}^\mathcal{T}} \prod_{k=1}^{K_0} \sum_{t \in t^+} \mathbf{1}_{t_k^{\mathcal{S}_\sigma}}(t) \prod_{m=1}^{l} f(\phi_{r^m}, O_{km}, c_{ktm})^{N(\phi_{r^m}, O_{km}, c_{ktm}, \mathcal{S}_\sigma)^{-1}} \quad (20)$$

with

$$f(\phi_{r^m}, O_{km}, c_{ktm}) = \phi_{r^m}(O_{km})^{c_{ktm}} \left(1 - \phi_{r^m}(O_{km})\right)^{1-c_{ktm}}. \quad (21)$$

In the above, $\mathbf{1}_{t_k^{\mathcal{S}_\sigma}}(t)$ is an indicator function which equals 1 if $t = t_k^{\mathcal{S}_\sigma}$ and 0 otherwise, for active row $t_k^{\mathcal{S}_\sigma}$ under structure $\mathcal{S}_\sigma$ and grounding k. $\mathbf{1}_{t_k^{\mathcal{S}_\sigma}}(t)$ has the role of only including the *single* summand where t corresponds with $t_k^{\mathcal{S}_\sigma}$. $N(\phi_{r^m}, O_{km}, c_{ktm}, \mathcal{S}_\sigma)$ is a function that counts the number of repeat products of term $f(\phi_{r^m}, O_{km}, c_{ktm})$, such that the appropriate root can be applied. We use r^m to denote the relation for atomic formula at column m and O_{km} its corresponding arguments under grounding k; and we use c_{ktm} to denote the truth-assignment of the atomic formula for column m, as designated by row t.

At this point, we are left with an expression for $\Gamma_\mathcal{T}^{\tilde{\mathcal{S}}_\sigma}$ in terms of truth-table matrix $\boldsymbol{T}$ entries, which is more reminiscent of $L(\mathcal{T}, \tilde{\mathcal{S}}_\sigma)$ as defined in Section 4. However, we must go further to expose the relationship between $\Gamma_\mathcal{T}^{\tilde{\mathcal{S}}_\sigma}$ and $L(\mathcal{T}, \tilde{\mathcal{S}}_\sigma)$ for arbitrary $\mathcal{T}$ expressed by $\boldsymbol{T}$. We will now show that the consistency loss $L(\mathcal{T}, \tilde{\mathcal{S}}_\sigma)$ gives the negative log-likelihood of satisfying $\mathcal{T}$ given a grounding $k \in \{1,\dots,K_0\}$, which can be further seen as a relaxation of $\Gamma_\mathcal{T}^{\tilde{\mathcal{S}}_\sigma}$ to sum over all rows $t \in t^+$ and without normalising via the $N(\phi_{r^m}, O_{km}, c_{ktm}, \mathcal{S}_\sigma)^{-1}$ exponent.

With Boolean random variable $B_{\mathcal{T}}$ denoting whether $\mathcal{T}$ is ($b_{\mathcal{T}} = 1$) or is not ($b_{\mathcal{T}} = 0$) satisfied, the consistency loss for a soft-structure $\tilde{\mathcal{S}}_\sigma$ against theory $\mathcal{T}$ is given by,

$$L(\mathcal{T}, \tilde{\mathcal{S}}_\sigma) = \mathbb{E}_{k\sim U[\{1,\ldots,K_0\}]}[H(p(B_{\mathcal{T}}|\mathcal{S}_\sigma, k), p(B_{\mathcal{T}}|\tilde{\mathcal{S}}_\sigma, k))] \qquad \text{Eqn. 8 base}$$

which can be expanded to,

$$\begin{aligned} L(\mathcal{T}, \tilde{\mathcal{S}}_\sigma) = -\sum_{k=1}^{K_0} \frac{1}{K_0} & p(b_{\mathcal{T}} = 1|\mathcal{S}_\sigma, k) \ln p(b_{\mathcal{T}} = 1|\tilde{\mathcal{S}}_\sigma, k) \\ & + (1 - p(b_{\mathcal{T}} = 1|\mathcal{S}_\sigma, k)) \ln(1 - p(b_{\mathcal{T}} = 1|\tilde{\mathcal{S}}_\sigma, k)). \end{aligned} \tag{22}$$

where $\mathcal{S}_\sigma \in \mathcal{M}_{\mathcal{S}}^{\mathcal{T}}$. Given $\mathcal{S}_\sigma \in \mathcal{M}_{\mathcal{S}}^{\mathcal{T}}$, then $p(b_{\mathcal{T}} = 1|\mathcal{S}_\sigma, k) = 1$ always holds. This means the negative case in Eqn. 22 can be ignored, yielding the following simplified form:

$$\begin{aligned} L(\mathcal{T}, \tilde{\mathcal{S}}_\sigma) &= -\sum_{k=1}^{K_0} \frac{1}{K_0} \ln p(b_{\mathcal{T}} = 1|\tilde{\mathcal{S}}_\sigma, k) \\ &= -\mathbb{E}_{k\sim U[1,\ldots,K_0]}[\ln p(b_{\mathcal{T}} = 1|\tilde{\mathcal{S}}_\sigma, k)]. \qquad \text{Eqn. 8} \end{aligned}$$

and so $L(\mathcal{T}, \tilde{\mathcal{S}}_\sigma)$ is simply the negative log-likelihood of sampling a satisfied theory ($b_{\mathcal{T}} = 1$) from soft-structure $\tilde{\mathcal{S}}_\sigma$, for randomly sampled grounding k. Next, we show the similarities between $L(\mathcal{T}, \tilde{\mathcal{S}}_\sigma)$ and $\Gamma_{\mathcal{T}}^{\tilde{\mathcal{S}}_\sigma}$ by looking at the likelihood $p(b_{\mathcal{T}} = 1|\tilde{\mathcal{S}}_\sigma, k)$. First, we define $\bar{\Gamma}_{\mathcal{T}}^{\tilde{\mathcal{S}}_\sigma}$ by isolating the likelihood:

$$\begin{aligned} \exp(-L(\mathcal{T}, \tilde{\mathcal{S}}_\sigma)) &= \prod_{k=1}^{K_0} p(b_{\mathcal{T}} = 1|\tilde{\mathcal{S}}_\sigma, k)^{\frac{1}{K_0}} \\ &\doteq \bar{\Gamma}_{\mathcal{T}}^{\tilde{\mathcal{S}}_\sigma} \end{aligned} \tag{23}$$

We then expand $p(b_{\mathcal{T}} = 1|\tilde{\mathcal{S}}_\sigma, k)$ to:

$$\begin{aligned} p(b_{\mathcal{T}} = 1|\tilde{\mathcal{S}}_\sigma, k) &= \sum_{t=1}^{K_1} p(b_{\mathcal{T}} = 1|\boldsymbol{c}_{kt}) p(\boldsymbol{c}_{kt}|\tilde{\mathcal{S}}_\sigma, k) \\ &= \sum_{t\in t^+} p(\boldsymbol{c}_{kt}|\tilde{\mathcal{S}}_\sigma, k) \end{aligned} \tag{24}$$

where t^+ is defined as before. For all other $t \neq t^+$, $p(b_{\mathcal{T}} = 1|\boldsymbol{c}_{kt}) = 0$ and so this acts as a filter, yielding:

$$\bar{\Gamma}_{\mathcal{T}}^{\tilde{\mathcal{S}}_\sigma} = \prod_{k=1}^{K_0} \sum_{t\in t^+} p(\boldsymbol{c}_{kt}|\tilde{\mathcal{S}}_\sigma, k)^{\frac{1}{K_0}}. \tag{25}$$

$p(\boldsymbol{c}_{kt}|\tilde{\mathcal{S}}_\sigma, k)$ is calculated by evaluating the belief of each relation-decoder against the expected truth-assignment as defined by truth-table row $\boldsymbol{c}_{kt}$:

$$\begin{aligned} p(\boldsymbol{c}_{kt}|\tilde{\mathcal{S}}_\sigma, k) &= \prod_{m=1}^{l} \phi_{r^m}(O_{km})^{c_{ktm}} (1 - \phi_{r^m}(O_{km}))^{1-c_{ktm}} \\ &= f(\phi_{r^m}, O_{km}, c_{ktm}) \end{aligned}$$

where r^m is the relation for atomic formula associated with column m (which is the same for each k slice and t row) and O_{km} is the grounding of this entry for slice k (which is the same across rows). Putting it all back together, we finally have that:

$$\bar{\Gamma}_{\mathcal{T}}^{\tilde{\mathcal{S}}_\sigma} = \prod_{k=1}^{K_0} \sum_{t \in t^+} \prod_{m=1}^{l} f(\phi_{r^m}, O_{km}, c_{ktm})^{\frac{1}{K_0}}, \tag{26}$$

which makes the similarities between $\Gamma_{\mathcal{T}}^{\tilde{\mathcal{S}}_\sigma}$ and $\bar{\Gamma}_{\mathcal{T}}^{\tilde{\mathcal{S}}_\sigma}$ clear and exposes their relationship. In particular, for the special case where $|\mathcal{M}_{\mathcal{S}}^{\mathcal{T}}| = 1$, the outer sum for $\Gamma_{\mathcal{T}}^{\tilde{\mathcal{S}}_\sigma}$ can be removed, and the remaining differences between $\Gamma_{\mathcal{T}}^{\tilde{\mathcal{S}}_\sigma}$ and $\bar{\Gamma}_{\mathcal{T}}^{\tilde{\mathcal{S}}_\sigma}$ are the sum over t^+ rows and difference in exponent over $f(\phi_{r^m}, O_{km}, c_{ktm})$. For $\Gamma_{\mathcal{T}}^{\tilde{\mathcal{S}}_\sigma}$ to be maximised, through $p(\mathcal{S}_\sigma|\tilde{\mathcal{S}}_\sigma) \approx 1$, we would find that $\tilde{\mathcal{S}}_\sigma$ maximally supports only the rows associated with $\mathcal{S}_\sigma$ for each k grounding. Notice that $\bar{\Gamma}_{\mathcal{T}}^{\tilde{\mathcal{S}}_\sigma}$ is again bound to $(0, 1)$ and achieves $\bar{\Gamma}_{\mathcal{T}}^{\tilde{\mathcal{S}}_\sigma} \approx 1$ when $\Gamma_{\mathcal{T}}^{\tilde{\mathcal{S}}_\sigma} \approx 1$. We use the correspondence between $\Gamma_{\mathcal{T}}^{\tilde{\mathcal{S}}_\sigma}$ and $\bar{\Gamma}_{\mathcal{T}}^{\tilde{\mathcal{S}}_\sigma}$ to define a practical ϵ-proxy consistency measure as follows. We firstly re-express ϵ-consistency/coherence but for $\bar{\Gamma}_{\mathcal{T}}^{\tilde{\mathcal{S}}_\sigma}$ and a different $\bar{\epsilon}$. We then trace this back to $L(\mathcal{T}, \tilde{\mathcal{S}}_\sigma)$ so a bound in terms of the consistency loss can be reported as the overall ϵ-proxy. Together this yields the following:

$$\begin{aligned} \bar{\epsilon} &\geq 1 - \bar{\Gamma}_{\mathcal{T}}^{\tilde{\mathcal{S}}_\sigma} \\ \ln \frac{1}{1 - \bar{\epsilon}} &\geq -\ln(\bar{\Gamma}_{\mathcal{T}}^{\tilde{\mathcal{S}}_\sigma}) \\ &\geq L(\mathcal{T}, \tilde{\mathcal{S}}_\sigma) \end{aligned} \tag{27}$$

and, via the relationship between $\bar{\epsilon}$ and $L(\mathcal{T}, \tilde{\mathcal{S}}_\sigma)$, we can use the consistency loss $L(\mathcal{T}, \tilde{\mathcal{S}}_\sigma)$ as a proxy measure for ϵ-consistency/coherence.

H Spearman's Rank Correlation Analysis

A Spearman rank correlation analysis was performed between each consistency loss (Con-A, Con-I-T, Con-I-A, Con-I-R) and PRT performance. Coefficients are reported for each combination of consistency loss and region of latent space (data-embedding, interpolation and extrapolation). Note that coefficients are not separated between β and relation-decoder choice. Overall, each coefficients

aims to characterize the PRT performance change when a relation-decoder is more or less consistent with a given partial theory, over a particular region of latent space. The key findings are reported in the main text and we tabulate the values in Table 4.

Unlike the popular Pearson correlation, the Spearman rank correlation can describe monotonic curvilinear relationships between variables. A Spearman rank coefficient varies between -1 and $+1$, where a coefficient ± 1 indicate a perfect rank correlation. If the coefficient is negative (positive) this means a reduction (increase) in one variable corresponds with an increase in the other.

Table 2: Specification of our β-VAE encoder and decoder model parameters, for both 28×28 (top) and 128×128 (bottom) size input data. I: Input channels, O: Output channels, K: Kernel size, S: Stride, P: Padding, A: Activation

Encoder
Input: $28 \times 28 \times N_C = 1$
Layer_ID ; I ; O ; K ; S ; P ; A
Conv2d_1 ; N_C ; 32 ; 4×4 ; 2 ; 1 ; ReLU
Conv2d_2 ; 32 ; 32 ; 4×4 ; 2 ; 1 ; ReLU
Conv2d_3 ; 32 ; 64 ; 3×3 ; 2 ; 1 ; ReLU
Conv2d_4 ; 64 ; 64 ; 2×2 ; 2 ; 1 ; ReLU
Layer_ID ; Num Nodes : In - Out ; A
FC_z ; 576 - 144 ; ReLU
FC_z_mu ; 144 - 10 ; None
FC_z_logvar ; 144 - 10 ; None

Decoder
Input: $\mathbb{R}^{10}$
Layer_ID ; Num Nodes : In - Out ; A
FC_z ; 10 - 144 ; ReLU
FC_z_mu ; 144 - 576 ; ReLU
Layer_ID ; I ; O ; K ; S ; P ; A
UpConv2d_1 ; 64 ; 64 ; 2×2 ; 2 ; 1 ; ReLU
UpConv2d_2 ; 64 ; 32 ; 3×3 ; 2 ; 1 ; ReLU
UpConv2d_3 ; 32 ; 32 ; 4×4 ; 2 ; 1 ; ReLU
UpConv2d_4 ; 32 ; N_C ; 4×4 ; 2 ; 1 ; Sigmoid

Encoder
Input: $128 \times 128 \times N_C = 3$
Layer_ID ; I ; O ; K ; S ; P ; A
Conv2d_1 ; N_C ; 32 ; 4×4 ; 2 ; 1 ; ReLU
Conv2d_2 ; 32 ; 32 ; 4×4 ; 2 ; 1 ; ReLU
Conv2d_3 ; 32 ; 64 ; 4×4 ; 2 ; 1 ; ReLU
Conv2d_4 ; 32 ; 64 ; 4×4 ; 2 ; 1 ; ReLU
Conv2d_5 ; 64 ; 64 ; 4×4 ; 2 ; 1 ; ReLU
Layer_ID ; Num Nodes : In - Out ; A
FC_z ; 1024 - 256 ; ReLU
FC_z_mu ; 256 - 10 ; None
FC_z_logvar ; 256 - 10 ; None

Decoder
Input: $\mathbb{R}^{10}$
Layer_ID ; Num Nodes : In - Out ; A
FC_z ; 10 - 256 ; ReLU
FC_z_mu ; 256 - 1024 ; ReLU
Layer_ID ; I ; O ; K ; S ; P ; A
UpConv2d_1 ; 64 ; 64 ; 4×4 ; 2 ; 1 ; ReLU
UpConv2d_2 ; 64 ; 32 ; 4×4 ; 2 ; 1 ; ReLU
UpConv2d_3 ; 32 ; 32 ; 4×4 ; 2 ; 1 ; ReLU
UpConv2d_4 ; 32 ; 32 ; 4×4 ; 2 ; 1 ; ReLU
UpConv2d_5 ; 32 ; N_C ; 4×4 ; 2 ; 1 ; Sigmoid

Table 3: Characteristic properties of ordinal relations.

Relation	asymmetric	transitive	reflexive
G	Y	Y	N
E	N	Y	Y
L	Y	Y	N
S	Y	N	N
P	Y	N	N

Table 4: Spearman rank coefficients between consistency loss and PRT accuracy. Coefficients are calculated for each consistency loss reported in the main text, across all models, β settings and regions of latent space. Results show a strong inverse rank correlation between interpolation Con-A/Con-I-T/Con-I-A and PRT performance.

	Spearman Rank Coefficient			
$\mathcal{Z}$ region	Con-A	Con-I-T	Con-I-A	Con-I-R
Data-Embeddings	0.2530	-0.4451	-0.4655	0.2307
Interpolation	-0.7655	-0.7479	-0.7120	-0.4233
Extrapolation	-0.6005	-0.6586	-0.6140	-0.4895