Toward a minimal model for structure learning

Krista Bond[1,2,3*]

1 Department of Psychology, Carnegie Mellon University, United States
2 Center for the Neural Basis of Cognition, United States
3 Carnegie Mellon Neuroscience Institute, United States

***kbond@andrew.cmu.edu**

Abstract

Humans regularly make adaptive inferences that go far beyond the data they observe. Structure learning – the abstraction of relational invariance from data – is one process thought to underlie this capacity. Compositional inductive bias, the tendency to represent complex structure as a composite of simpler structures, is a common feature of the predominant models of structure learning. Further, computational models without this bias are sample inefficient and relatively inflexible, suggesting that compositionality may be a necessary computational requirement for structure learning. Graph networks can flexibly represent diverse structures in a compositional way. Using graph networks in the reinforcement learning context may offer a powerful and tractable framework for furthering our understanding of structure learning across domains.

> Our emotional, personal, and intellectual characteristics are not the mere algebraic summation of a near infinity of stimulus-response bonds. The learning of primary importance to the primates, at least, is the formation of learning sets; it is the learning how to learn efficiently in the situations the animal frequently encounters. This learning to learn transforms the organism from a creature that adapts to a changing environment by trial and error to one that adapts by seeming hypothesis and insight.
>
> *H.F. Harlow, 1949* [35]

Structure learning carves an adaptive hypothesis space

Akin to Plato's observers drawing inferences of form from the play of shadows cast by unobservable objects [68], humans regularly infer structure from variable input in domains as diverse as sensorimotor learning and concept learning [88]. This structural

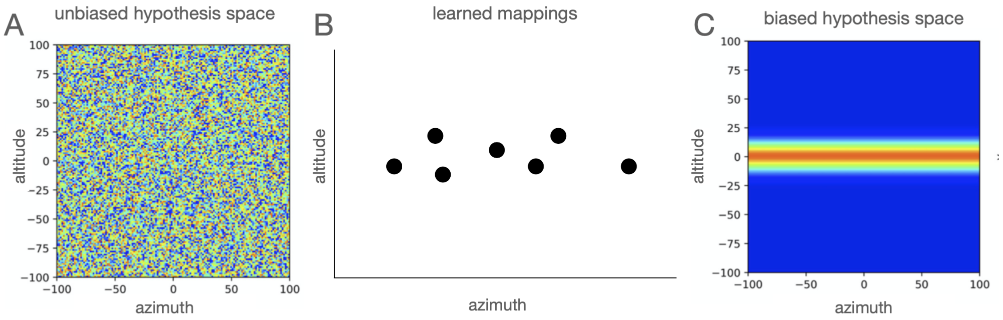


Figure 1. Economizing inquiry. A) An unbiased hypothesis space. Here, any combination of changes in azimuth or altitude are equally viable to explore. B) Adaptive mappings learned over time to oppose variable wind velocities. C) After learning variable mappings, the archer learns to explore a biased (azimuthal) hypothesis space. Limiting the hypothesis space to azimuthal changes in aim allows for faster adaptation to novel perturbations because the task is reduced to reparameterizing a learned function rather than learning a new function. Figure adapted with permission from [11].

inference is thought to rely on the isolation of features invariant to the task at hand [13, 40]. For example, we know that chairs have signature component features (e.g. armrests, a seatpan, a back rest, and four legs), independent of the particular parameters of those features. This isolation of invariance places constraints on the viable space of hypotheses for faster recognition or adaptation [3]. In other words, if you know that a chair has a set of invariant features, you should be able to more quickly recognize the novel object before you as a chair, even if you have never seen a chair with that seatpan depth or back rest height. Put differently, learning a structure is analogous to model discovery; for any given input, the output follows the structure defined by the functional form. Under this analogy, learning the form of the function thus places a constraint on the set of possible solutions given any input. Assuming that the learner approximates a useful function for a given context, the range of possible outputs is thus drastically reduced from a *tabula rasa* prediction in which the hypothesis space is unbiased.

Imagine a novice archer attempting to hit a target on a windy day. At first, she may not know how to counteract the wind – whether she should aim side-to-side or up-and-down. Her hypothesis space may be unbiased, with an equal probability of exploring a change in azimuth or a change in altitude (Fig. 1A). Now consider a single scenario in which the crosswind approaches from the left. She learns to adapt by aiming her arrow to the right with sufficient magnitude to counter the wind. Adaptation to this isolated learning situation represents an instance of strictly parametric, or associative learning. In this case, her previously unbiased hypothesis space shrinks to a single input-output mapping (Fig. 1B). Alternatively, if she is exposed to crosswinds with varied velocities, then she learns the general principle that she should aim her arrow opposite to the wind. After learning this principle, she is capable of quickly opposing novel wind velocities because her hypothesis space is now constrained to azimuthal changes in aim (Fig. 1C). In this way, structure learning economizes inquiry [64] by carving a hypothesis space consistent with the structure of invariant task features.

Structure learning has been broadly classified into learning three primitive structural forms: functions, clusters [48] and features [27, 40]. Learning a function is learning the relationship between two or more continuous variables. In the archery example this would correspond to learning the relationship between azimuth and altitude that opposes a wind perturbation. Here, the computational goal can be reduced to learning the function $y = f(x)$ using a noisy set of observations $X = (x_1...x_n) + \epsilon$ and their outcomes $Y = (y_1...y_n)$ [4]. In statistics, this is a regression problem [38]. Cluster learning is the problem of how to group observations into discrete units [38] and can be related to category learning [74], word segmentation [30], causal inference [26], and associative learning [28]. One example of cluster learning is deciding whether, an animal that has a tail and floppy ears is a dog. Cluster learning can be considered a density estimation problem [4]. In this case, what is the probability that this animal is a dog given these perceptual features? Given that we know the probability of an animal having a tail and floppy ears given that it is a dog; the probability that we encounter dogs; and the joint probability that we encounter those features, we can estimate the probability that this animal is a dog using Bayes' rule (see Nonparametric Hierarchical Bayesian Models (NPHBM) for more detail). Feature learning is the problem of how to define a perceptual unit or assign attributes to a stimulus [4]. Using the dog example, this is asking the question of how we represent a dog as a given set of perceptual features (a tail and floppy ears). Feature representation can be related to choice behavior and similarity judgements [69], and object recognition [67]. The computational problem of feature learning can be thought of as a factorization problem, with links to dimensionality reduction techniques [38]. Functional, cluster-based, and featural primitives can be recombined to capture a set of more complex structures or models [22, 80] (Fig. 2).

In an attempt to advance the understanding of structure learning as a domain-independent feature of human learning, I review the primary computational models developed to explain structure learning across learning domains. I identify one computational feature shared by these models and propose a powerful, tractable framework for further developing a model containing this shared computational feature.

To impose structure on my search for computational features, I borrow the minimal model approach to explanation from philosophy of science [8]. A minimal model isolates conceptual commonalities among extant models, often explaining diverse expressions of a given phenomenon. These commonalities guide the specification of one necessary – but not sufficient – property of the phenomenon in question. This property should be determinable, in that it is specific enough to guide experiment but abstract enough to unite different domains of learning. This minimal model approach has been used in cases where the phenomenon under question is underspecified or has competing explanations [5, 98, 102].

Integrating over theoretical orientations and computational architectures in this way can be thought of as balancing the assumptions of these models with their explanatory power.

I included theoretical perspectives on structure learning implemented as computational models, that successfully addressed structure learning in multiple learning domains, were described in detail sufficient to understand the algorithm, and were recent (updated within the past 20 years).

I end the review with a set of open questions and include a glossary of key terms.

Below, I summarize the mechanisms, strengths, and weaknesses of the primary cognitive computational models [41, 61] for structure learning that meet the above criteria.

Nonparametric Hierarchical Bayesian Models (NPHBM)

The predominant approach to modeling the computations underlying the acquisition of structured abstractions relies heavily on the probabilistic model of cognition framework [14, 33, 44]. This framework relies on a rational analysis of cognition, outlining the computations that should be used to solve a given statistical inference problem.

Bayes' rule can set a normative standard for how to update event probabilities given known observations [9]:

$$p(a|b) = \frac{p(b|a)p(a)}{p(b)} \tag{1}$$

Above, we can calculate the probability of an event a given that event b has occurred, provided that we know the independent probabilities of events a and b and the probability that event b is true if we observe event a.

Within this larger framework, Nonparametric Bayesian Hierarchical Models (NPHBM) of cognition assume that agents solve a similar statistical inference problem, but also account for uncertainty in the identification of structured abstractions by updating the pool of candidate structures (H) using new observations (d) [4]:

$$p(h|d) = \frac{p(d|h)p(h)}{\sum_{h' \in H} p(d|h')p(h')} \tag{2}$$

where H is the space of possible hypotheses or structures, h_i is a given hypothesis or structure, d is the observed data, $p(h)$ is the prior probability, or the belief in the data-generating structure prior to observing new data points, $p(h|d)$ is the posterior probability, or the updated belief after observations, and $p(d|h)$ is the likelihood the data are produced by a given structure.

Given a new observation d_{i+1} and previous observations d, the degree of belief in the data-generating structure can then be updated:

$$p(d_i|d) = \sum_h p(d_i|H)p(h|d) \tag{3}$$

where $p(h|d)$ corresponds to the posterior probability described in Equation 2.

To concretely relate this method of updating to the discovery of a structure, consider the problem of deciding which function among a set of functions best characterizes the relationship between two observed continuous variables. Recall from that our computational goal is to solve the regression problem of finding $y = f(x)$ using a noisy set of observations $X = (x_1 ... x_n) + \epsilon$ and their outcomes $Y = (y_1 ... y_n)$. We can combine our prior beliefs about the probability that we encountering different classes of functions in our lives with the data we observe, x and y:

$$p(f|X,Y) = \frac{p(Y|f,X)p(f)}{\int_F p(Y|f,X)p(f)df} \tag{4}$$

Here $p(f)$ is the prior distribution over classes of functions in the function hypothesis space F. $p(Y|f,X)$ is the likelihood of observing the Y if f were the true function, and $p(f|X,Y)$ is the posterior distribution over classes of functions given X and Y.

But how do we define the function hypothesis space, F? This hypothesis space of functions could be infinite. This is the primary flaw of Bayesian models of structure learning. Assumptions regarding the form of the function (e.g. $y = b_0 + xb_1$) over which we define the hypothesis space F makes our task tractable, but it also restricts our capacity for functional form discovery.

Moreover, what if we didn't even restrict our hypothesis space to a functional hypothesis space and instead were completely naive to the structural form we were encountering (Fig. 2)? Kemp and Tenenbaum [40] were able to discover not only the parameterization of structures, but also the form of the learning problem using a set of graph grammars as viable structural forms. Here, a Bayesian hierarchical model defined over a space of graph grammars can discover the form (F) and structure (S) that best characterize the data (D) by maximizing the posterior probability of a given structure

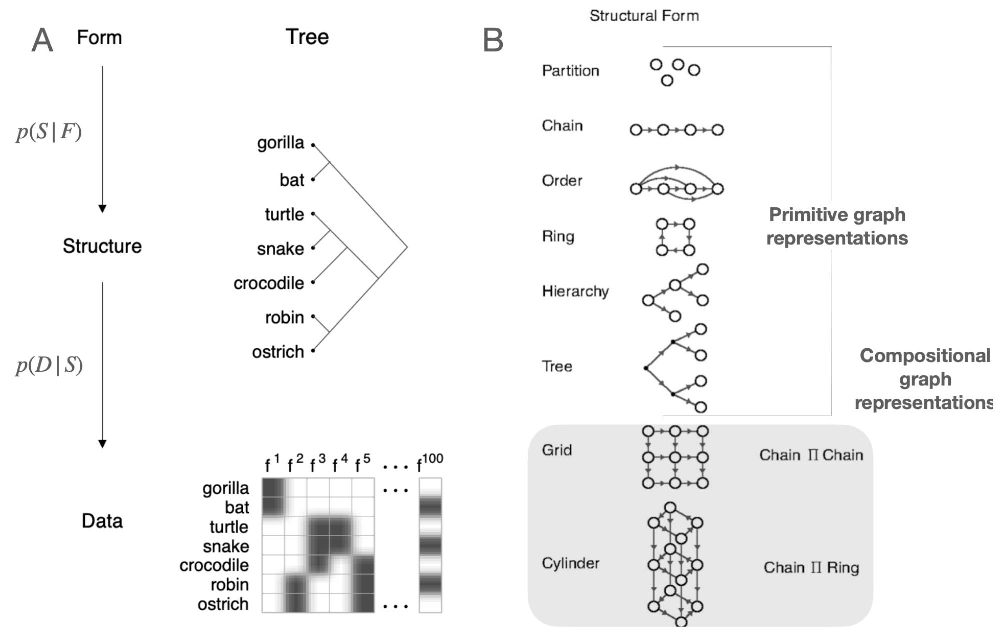


Figure 2. The relationship between form and structure. The versatile representational capacity of graph grammars. A) The relationship between form, structure, and raw data. The form defines the categorical type of model learned. Structure specifies the particular instantiation of a form. Structure is learned from raw data. Here, the raw data consist of the features $f_1 ... f_{100}$. Different animals have different feature compositions. The relationship between featural distributions defines the relationship between animals. The model, or representational form selected is that of a tree. A Bayesian hierarchical model defined over a space of graph grammars can discover the form (F) and structure (S) that best characterize the data (D) by maximizing the posterior probability of a given structure and form given the data: $P(S, F|D) \propto P(D|S)P(S|F)P(F)$ [40]. B) Many primitive forms can be represented using a graph grammar. Importantly, primitive graph grammars can be composed to represent more complex structures, as shown in the grid and cylinder graph representations. Figure adapted from [40]. Permission pending.

and form given the data: $P(S, F|D) \propto P(D|S)P(S|F)P(F)$ (Fig. 2A). Notably, graph grammars are well-suited to represent the three primitive forms of structure learned – clusters, features, and functions – (Fig. 2B). and using the principles of compositionality, they can be combined to create more complex structures 2B. While promising, this approach still relied on a predefined set of graph grammars, or a hypothesis space of structural forms. Altogether, the problem of selecting a prior on structural forms (also known as a hyperprior) remains for NPHBMs.

These models have successfully explained diverse phenomena [88] and, when the structure learning problem is framed as a graph grammar discovery problem [72], these models have been validated to detect true underlying structures [40]. For example, when provided with distances between cities, NPHBMs have rediscovered longitude and latitude; when given supreme court votes, NPHBMs have identified that the underlying voting pattern of US supreme court judges lies along a left-right spectrum; and when given sets of biological features of animals, they are clustered according to realistic taxa [40].

Impressively, when learning novel character concepts, NHBMs can not only discover statistical structure on par with humans, but they can also generate that structure in a manner often indistinguishable from human generated letters under data-limited conditions [46]. Notably, the success of this particular instantiation of NHBM relied on observing the composition of characters (the stroke-by-stroke character generation process) to detect primitives.

This compositional approach to structure learning popularized subgoal discovery, or how to solve components of learning problems, as a method for the efficient discovery of structure in both machine learning algorithms [47] and in updating models of human computation [22]. The specification of subgoals allows agents to navigate hierarchically composed reinforcement learning architectures [83] to benefit transfer to novel scenarios that share underlying structure [52, 53]. Humans have also been shown to identify subgoals during learning and reuse and recombine partial solutions for generalization [22, 79, 103].

These models are useful in that they can plausibly model the consideration of multiple hypotheses, detect true structures, use few samples in order to infer the proper structure, and generalize sparsely observed properties [25]. The issue with these nonparametric models of structure learning is that they are of limited use when the task becomes complex and as observations increase because the set of possible hypotheses that could account for the observations expands intractably. However, the specification of prior knowledge and the selection of efficient search algorithms can constrain the set of considered hypotheses [89].

Note that this approach hinges on the assumption that the language of probability represents subjective belief and captures the dynamics of belief updating. Specifically, it assumes that degrees of subjective belief can be described in terms of probability and that belief *updates* according to the laws of probability.

Limitations notwithstanding, ample evidence exists to suggest that these models are a useful stand-in for human structure discovery, with human-level performance achieved in learning concepts, causal inference, how to parse motion, and more [88].

In sum, NPHBMs can discover the primitive forms of structure, can exploit compositionality to discover complex structures composed of primitive forms and aid generalization, and they have been applied to an expansive set of learning domains.

The Tolman-Eichenbaum Machine (TEM)

When Tolman introduced the concept of cognitive maps [90], he introduced the concept of latent learning, particularly in terms of latent relational representations that go beyond stimulus-response association. The bulk of research citing Tolman's effects and his theoretical interpretation have been justifiably used to support the idea of spatial maps [62] and their corresponding neural representations of space [21]. Recent findings suggesting that the neural encoding of spatial maps also represent nonspatial features [2, 15] have prompted the re-examination of Tolman's cognitive maps to study relational structure at the level of knowledge organization for nonspatial inference in both humans and machines [10, 97], with empirical support for the idea that the reorganization of knowledge in terms of cognitive maps aids generalization to shared knowledge structures [39, 51, 55].

Recently, spatial and relational memory have been united in an artificial neural network model of the hippocampal-entorhinal system termed the Tolman-Eichenbaum machine (TEM) [97]. This model proposes that medial entorhinal cells encode structural knowledge and hippocampal cells link this encoding with sensory representations. Within this framework, both relational and spatial reasoning can be considered as forms of structural generalization. Structural features of nonspatial problems also facilitate adaptive inference. For example, when presented with a transitive inference problem [36] (if A > B and B > C, how does A compare to C?), the concept of a linear system of magnitude enables the needed inferential leap (A must be

> C). Likewise, if the learner's task is to find the shortest path back to a starting point using only movements in the cardinal directions (North, South, East, West), then the knowledge that we live in three-dimensional Euclidean space allows us to infer that moving South, East, North, and then West will bring us back to the starting point using the minimal number of cardinal movements [97].

As mentioned in Nonparametric Hierarchical Bayesian Models (NPHBM), representing knowledge in terms of components, or *factorizing* knowledge, and recombining those components enables combinatorial generalization. Factorized sensory representations are encoded in the lateral entorhinal cortex and factorized spatial representations are encoded in the medial entorhinal cortex [54]. TEM demonstrates that novel compositions of factorized sensory and spatial representations can build a relational memory system capable of generalizing over both space and relational structures [97].

Here an unsupervised learning agent is tasked with predicting the next image in a sequence with a particular transition probability (note that this learning agent is not tasked with action selection as the actions are provided; it is simply asked to predict the next image in the sequence). The transition probability is defined over a directed graph structure, where each node represents an image, edges represent transition probabilities between images, and the transition between images is unidirectional.

While the full TEM algorithm is out of the scope of this paper, the two key features are the graph-based representations of a path between images and relational memories that bind the abstraction of a location to particular sensory experiences (images). Relational memories combine factorized sensory representations (images) with factorized structural representations (location). Note that TEM is not a graph neural network [75] and has no representation of edges or nodes. Graph structures only represent the structure of the problem.

The full TEM model includes an inference model, an artificial neural network that learns network-level weights via backpropagation and synaptic weights via Hebbian learning, and a generative model. The inference model is fed sensory data at each time step. The inference model is so called because it infers the location (g_t) and the composition of the location g with sensory experience x as a "sensorially" grounded variable p. Thus, p is the composition of sensory experience with abstract relational knowledge. At each time step, the inference model infers g before inferring p and the generative model calculates a prediction error for the inferred variables. Specifically, the generative model compares g_t to its prediction of g_{t-1}, its prediction of p_t from g_t, and its prediction of x_t from p_t. This results in sensory prediction errors (x), abstraction errors (g), and compositional errors (p). After an image sequence ends, these prediction errors are accumulated and both the inference model and the generative model update their parameters and return to the cycle of computation with the next set of images.

While this model is quite complex relative to the others reviewed here, it provides another demonstration of the utility of compositional representations but for both spatial and relational generalization. Theoretically, TEM should be able to learn functions, clusters, and features, as these structures can be represented using graphs (see Fig. 2 and The Graph Neural Network (GNN) as a framework for developing a minimal model of structure learning), but this is a relatively new model and this capacity has yet to be tested. However, the current instantiation of TEM can learn and generalize not only spatial tasks, but tasks reliant on abstract relational structure. This includes transitive inference tasks in which TEM generalizes learned relationships independent of sensory features and learns social hierarchies to generalize to unseen relationships.

The Successor Representation (SR)

In the reinforcement learning context, the concept of cognitive maps encoding relational structure of the environment has recently re-emerged as an updated version [59] of the successor representation [17]. The successor representation (SR) is a reinforcement learning algorithm that builds a predictive map of the environment to summarize the relationship between states separated by multiple state transitions. To accomplish this long-range prediction of state, the SR occupies an intermediate position on the model-based to model-free continuum of reinforcement learning, balancing the tradeoff between biased and flexible decision-making [24]. Unlike standard temporal difference learning algorithms that operate over prediction errors in value (Eqn. 5), the SR can be learned via a form of temporal difference learning using the difference between observed and predicted state occupancy as the error signal [56] (Eqn. 6).

Standard model-free reinforcement learning updated using temporal difference learning:

$$V(S) = V(S) + \alpha(R_{observed} + \gamma V(s_{new} - V(S)) \tag{5}$$

The successor representation updated using temporal difference learning:

$$M(S) = M(S) + \alpha(onehot(s_{new}) + \gamma M(s_{new} - M(S)) \tag{6}$$

Here M represents the successor representation matrix. The $onehot(s_{new})$ keeps track of state visitation. When the agent visits a new state, one visit is added to the count of visits to that state in the row corresponding to that state in M. The sucessor prediction error is the difference between the expected successor of state s from predictive horizon discounted sucessors of the new state. A learning rate α applies to the prediction error.

Offline replay, a memory process in which the hippocampal network internally generates patterns of activation representing compressed versions of prior experience [82], has been suggested to combine current experience with previous memories [65] to guide future behavior [57, 58]. Offline replay is not solely a repetition of the past, but a dynamic process sensitive to goal-specification [66] that reverses in response to prediction error [1]. Specifically, human and animal studies have shown a role for offline replay in inferring latent environmental structure [58, 101]. Combining SR with a family of reinforcement learning algorithms called Dyna [84] (SR-Dyna) shows promise as a computational framework for learning relational structure [73]. Here, the predictive map learned by the SR is learned online and state transitions are replayed offline. Mounting evidence supports the plausibility of SR-Dyna in both humans and rats as a computational basis for reinforcement learning [18, 59].

Finally, the successor representation can be decomposed into successor features, which abstract successor representations from their context to define primitive components of state representation [56]. This decomposition allows the agent to generalize to tasks that require similar component features [53, 56]. As mentioned in the previous section, the complexity of the hierarchical reinforcement learning problem can be drastically reduced by defining subgoals, or “options” [83]. Similarly, decomposing successor features compactly represents abstracted subroutines to reduce the complexity of the problem space while maintaining a state-based representation, and, due to the recombinant nature of these features, this also increases the span of tasks to which the successor representation can generalize because multiple task solutions can be represented as the linear combination of features [52, 53].

While the SR shows great promise, the range of applications is limited in comparison to NPHBMs. While they can represent functions and features [56], it remains to be seen whether they can represent clusters. Despite this, they provide another computational data point for the importance of compositional inductive bias for structure learning.

Minimal representational recipe for an agent to flexibly learn structure

Bayesian approaches to cognition [88] (Nonparametric Hierarchical Bayesian Models (NPHBM)), the recently developed Tolman-Eichenbaum machine [97](The Tolman-Eichenbaum Machine (TEM)), and the successor representation [56]The Successor Representation (SR) have been successfully developed to describe structure learning in a range of psychological domains (Fig. 5). Among these models only Bayesian approaches are clearly capable of representing all three forms of primitive structure and composing those representations to create more complex forms of structure. However, compositional inductive bias is common to all of these successful models, suggesting that it may be important for structure learning.

Standard metalearning algorithms can form useful inductive biases, but they remain insensitive to compositional structure (even when trained directly on compositional task distributions), encoding statistical structure instead of compositional representations [32]. Additionally, it is easier for these standard metalearning models to learn noncompositional distributions. In contrast, as mentioned throughout this paper, humans form compositional representations that allow for flexible recombination for sample-efficient generalization to complex learning scenarios. When the utility of compositionality is viewed in light of the diminished capacity of standard metalearning models to learn efficiently when they do not have architectures compatible with compositional learning [31], an argument could be made for compositional inductive bias as necessary to flexibly learn diverse structured representations.

A minimal model isolates the common features of computational models that explain diverse expressions of a given phenomenon. These common features guide the specification of one necessary – but not necessarily sufficient – property of the phenomenon in question. Further this property should be determinable: sufficiently descriptive to form a concrete criterion but abstract enough that it can be further refined for the purpose of further theoretical development [8, 98].

A model specifying compositional inductive bias as a primary computational feature meets the criteria for a minimal model of explanation. First, compositional inductive bias is a primary feature of all dominant computational models of structure learning, which address the structure learning problem in diverse learning contexts (see Domains of Application in Fig. 5). Computational architectures without this feature are sample inefficient and less flexible in generalization, suggesting necessity. Finally, the concept of compositional inductive bias is general enough to apply to structure learning problems in diverse domains and specific enough to refine the development of experiments and computational models. For the remainder of the paper, I develop the case for graph neural networks [7] as a framework for developing a minimal model of structure learning featuring compositional inductive bias.

The Graph Neural Network (GNN) as a framework for developing a minimal model of structure learning

Battaglia and colleagues present a hybrid of graph-structured representations and deep learning in the form of a graph neural network [7]. While artificial intelligence attempts have traditionally viewed symbolic, or structured representations as in opposition to connectionist "end-to-end" approaches, graph networks leverage *a priori* structure and emergent representations as complements. Graph networks use strong relational inductive biases in the form of architectural assumptions within deep learning architectures to capture elemental entities, relations, and rules for compositional

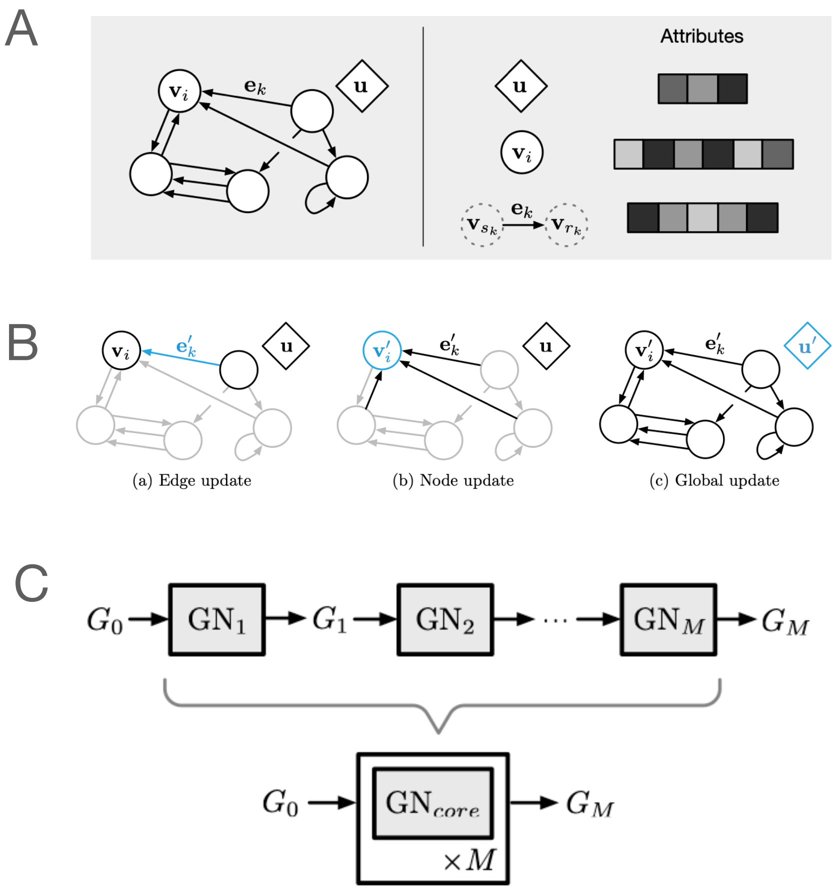


Figure 3. Graph Neural Networks. A) A graph-structured representation. A graph is composed of nodes (v_i) that represent entities, edges (e_k) that represent relationships between entities, and global attribute (u) that represent graph-level attributes [7]. Nodes and edges can also have attributes. Thus, any graph can be denoted as a 3-tuple ($G = u, V, E$). In this context, graphs are directed, meaning they propagate information in only one direction: from a sender node (v_{s_k}) to a receiver node (v_{r_k}). B) Different kinds of updates within a graph. Here, the element depicted in blue signifies the updated element, elements involved in the given update are black, and elements irrelevant to the update are gray. Edges (e_k), nodes (v_i), and global attributes (u) can be updated. C) Multiple graph network blocks (GN_i) can be composed to form a graph network core (GN_{core}) within which M internal processing substeps can be computed. The blocks that form a GN can share functions and parameters or they can be independent, with different functions and parameters. G_0 represents the initial graph passed to the GN_{core} and G_M represents the graph transformed by the GN_{core}. Figure adapted from [7]. Permission pending.

representations [7]. A graph is described by a set of nodes (v_i), edges (e_k) that represent relationships between nodes, and global attribute (u) that represent graph-level attributes [7]. Graphs are denoted as a 3-tuple ($G = u, V, E$) (Fig. 3A).

Three core features of graph neural networks suggest they show promise as a framework for developing a minimal model of structure learning across domains.

GNNs are versatile. They have the capacity to represent diverse structures, including the three primitives of structure learning (functions, features, and clusters). This comes from the versatility of graph-structured representations (see Fig. 2B) and from the attributes assigned at the graph, node, and edge level. The output of a GN block can also be customized to prioritize learning over different aspects of the graph (Fig. 3B), as appropriate for the task domain. For example, if the task requires reasoning about interobject relations, learning over edges may be most appropriate. Hamrick and colleagues [34] investigated human physical reasoning using a task that involved gluing pairs of blocks together to stabilize a tower. Using a graph network with a recurrent message-passing neural network [7], they modeled the positions and orientations of the blocks as nodes and the presence of glue as edges, with the global attribute corresponding to global pieces of information such as the overall stability of the tower. They found that this GNN outperformed humans and nonrelational networks in physical reasoning [34].

Within-block update dynamics are configurable. To describe how the functions internal to the GN block allow for flexible dynamics, I step through how each element is updated and how those updates are aggregated below, then summarize the algorithm underlying one cycle of computation within a block.

A GN block has two classes of functions: update functions and aggregation functions. Each feature of the graph has separate update functions, denoted ϕ:

$$e'_k = \phi^e(e_k, v_{r_k}, v_{s_k}, u) \tag{7}$$

$$v'_i = \phi^v(\bar{e}'_i, v_i, u) \tag{8}$$

$$u' = \phi^u(\bar{e}', \bar{v}', u) \tag{9}$$

Here, each of k edges (e) is updated (e') using the current value of the edge e_k, the receiving node v_{r_k}, the sending node v_{s_k}, and the global graph attribute u (Eqn. 7). Each of i nodes (v) is updated (v') using the aggregation of edges that project to that node ($\bar{e}'_i$, the current value of the node v_i, and the global attribute u (Eqn. 8). The global attribute (u) is updated (u') using the aggregate of all edge updates ($\bar{e}'$), the aggregate of all node updates ($\bar{v}'$), and the current value of the global attribute u (Eqn. 9).

The edges projecting to each node, all edges in the graph, and all nodes of the graph have separate aggregation functions, denoted ρ:

$$\bar{e}'_i = \rho^{e \to v}(E'_i) \tag{10}$$

$$\bar{e}' = \rho^{e \to u}(E') \tag{11}$$

$$\bar{v}' = \rho^{v \to u}(V') \tag{12}$$

Here, the set of updates for the edges between each receiving node (r_k) and sending node (s_k) is denoted E'_i, where $E'_i = \{(e'_k, r_k, s_k)\}_{r_k = i, k = 1:N^e}$ and the aggregation of these node-specific edge updates E'_i is denoted $\bar{e}'_i$(Eqn. 10). The set of updates over all edges within the GN block is denoted E', where $E' = \cup_i E'_i = \{(e'_k, r_k, s_k)\}_{k_1:N^e}$ and the aggregation of these edge updates over the graph is denoted $\bar{e}'$ (Eqn. 11). The set of updates over all nodes is denoted V' where $V' = \{v_i\}_i = 1 : N^v$ and the aggregation of these node updates over the graph is denoted $\bar{v'}$ (Eqn. 12). This aggregation of node updates $\bar{v'}$ and the aggregation of edge updates over the graph $\bar{e'}$ will then be used in the next global update u.

In total, the full algorithm for computation within a GN block (Fig. 4) goes as follows.

Given a set of edges E, a set of nodes V, and a global attribute u as inputs:

1. A graph defined by a set of edges E, a set of nodes V, and a global attribute u (G_0) is fed to the GN block.
2. For all edges, attributes are updated for all nodes according to (Eqn. 7).
3. Edge attributes are aggregated for each node (Eqn. 10).
4. Individual node attributes are updated given the edge updates aggregated for each node (Eqn. 8).
5. Edge attributes are aggregated over nodes (Eqn. 11).
6. Node attributes are aggregated over the graph (Eqn 12).
7. The global attribute is updated using the aggregation of edge attributes over nodes, the aggregation of node attributes over the graph, and the current global attribute (Eqn. 9).
8. A graph with the updated set of edge attributes, the updated set of nodes, and an updated global attribute is generated (G_M).

The flexibility lies in the attribute update functions for edges, nodes, and global attributes, and in the method of aggregation over edges per node, the aggregation over all edges, and in node aggregation. This within-block flexibility is important because it allows a primitive representation of structure (Fig. 2B) to change dynamically over time. You might imagine that, given new observations, a learner decides that the data are more consistent with a categorically different structural form (e.g. a featural primitive rather than a clustering primitive); within-block dynamics enable such a switch between primitive forms.

GNNs are compositional. GNN blocks can be combined to produce more complex architectures. The most basic form of a multi-block GNN uses two blocks, GN_1 and GN_2, with the output of the first acting as input to the secon to generate a final graph G': $G' = GN2(GN1(G))$. Arbitrary numbers of GN blocks can compose a multi-block GNN [7] (Fig. 3C). Blocks can share functions and parameters, akin to a recurrent neural network, or have independent functions and parameters.

To summarize, I make the following claims.

- Given that graph structures can represent all three forms of established primitive structures in human structure learning [4] (Fig. 2) and that GN blocks represent the atomic unit of computation for a GNN, the GN block may effectively represent primitive structural forms.
- Representing these primitive structural forms as GN blocks with adaptable within-block configuration enables a representation of structural form to change over time, possibly capturing dynamic shifts in structural form in response to experience.

- Graph neural networks allow for the combination of GN blocks with shared or unshared update functions. This means that graph neural networks can represent compositional structure [80], or more complex structures formed from simpler building blocks. This is critical for generalization because compositional learning allows agents to reason about entirely new learning problems formed from a composite of previously learned structures [22]. Further, it seems reasonable to expect that, even if a new learning problem only partly overlaps with a previously learned set of simpler structures, the hypothesis generation process might be economized (at the time of writing, I know of no explicit studies of this nature).

Critically, there is emerging empirical support for multi-block graph network architectures leveraging combinatorial representation to improve sample efficiency and increase the flexibility of generalization, as has been amply demonstrated in humans [22, 43, 45, 47, 71, 78–80, 91]. This compositional capacity of GNNs has been shown to result in sample-efficient combinatorial generalization in diverse domains [7]. In the realm of physical reasoning [42], GNNs can make predictions about n-body problems, rigid-body collision, and non-rigid dynamics and then generalize to systems with different numbers of objects and different configurations of elements [6]. Most impressively, GNNs have been shown to recover symbolic representations of known physical laws and have been used to discover novel physical principles; specifically, a new analytical formula predicting the concentration of dark matter from the mass distribution of surrounding cosmic structures was discovered, and predictions from this formula generalized to out-of-distribution data [16]. GN-based decision policies have also shown notable transfer [34, 50, 94], with a recent study demonstrating the development of a single reinforcement learning policy to successfully control the movement of agents of different morphologies if the agents were represented as graph-structures with shared properties, or modules [37]. Finally, message-passing GNNs outperform Bayesian belief propagation models when making inferences in the context of graph-structured tasks and generalize out-of-set to larger graph structures and graphs with different structures [105].

Despite their success, the representational capacity of GNNs can be limited. Without additional assumptions, representations that require recursion and conditional logic are nontrivial to represent as a graph [7]. The capacity to discriminate between simple graph structures can be compromised in GNNs based on convolutional networks but simpler GNNs exist that do fare well on graph classification benchmarks [104]. Care must be taken to ensure that the GNN developed can arbitrate between graph structures before drawing conclusions. The basis for design decisions regarding the initial structure of the input graph is also underspecified given a particular learning problem [7], though either 1) iterative experiments for which a prior can be plausibly induced on the graph structure or 2) a theoretical basis for default graph structure representations may be useful in addressing this problem. Relatedly, it's unclear whether graph operations are sensitive to initial conditions, or ”permutation invariant” (e.g. would two graphs with different numbers of nodes or edges subject to the same computational architecture result in different output graphs?) and generalization can be limited in message-passing GNNs [23].

Using GNNs to study structure learning in the reinforcement learning context is both a tractable and powerful framework with which to develop a minimal model of structure learning featuring compositional inductive bias. Multi-block GNNs imbued with compositional inductive bias have shown generalization across diverse domains of learning. Further, they are well-suited to represent structural primitives and their dynamics while harnessing the power of both symbolic and representational architectures. At a pragmatic level, remaining within the sphere of reinforcement learning is a tractable way to manipulate the structures to which humans (or agents

Algorithm 1 Steps of computation in a full GN block.

```
function GraphNetwork($E$, $V$, $\mathbf{u}$)
    for $k \in \{1 \ldots N^e\}$ do
        $\mathbf{e}'_k \leftarrow \phi^e\left(\mathbf{e}_k, \mathbf{v}_{r_k}, \mathbf{v}_{s_k}, \mathbf{u}\right)$        ▷ 1. Compute updated edge attributes
    end for
    for $i \in \{1 \ldots N^n\}$ do
        let $E'_i = \left\{\left(\mathbf{e}'_k, r_k, s_k\right)\right\}_{r_k=i,\; k=1:N^e}$
        $\bar{\mathbf{e}}'_i \leftarrow \rho^{e \rightarrow v}\left(E'_i\right)$        ▷ 2. Aggregate edge attributes per node
        $\mathbf{v}'_i \leftarrow \phi^v\left(\bar{\mathbf{e}}'_i, \mathbf{v}_i, \mathbf{u}\right)$        ▷ 3. Compute updated node attributes
    end for
    let $V' = \{\mathbf{v}'\}_{i=1:N^v}$
    let $E' = \left\{\left(\mathbf{e}'_k, r_k, s_k\right)\right\}_{k=1:N^e}$
    $\bar{\mathbf{e}}' \leftarrow \rho^{e \rightarrow u}\left(E'\right)$        ▷ 4. Aggregate edge attributes globally
    $\bar{\mathbf{v}}' \leftarrow \rho^{v \rightarrow u}\left(V'\right)$        ▷ 5. Aggregate node attributes globally
    $\mathbf{u}' \leftarrow \phi^u\left(\bar{\mathbf{e}}', \bar{\mathbf{v}}', \mathbf{u}\right)$        ▷ 6. Compute updated global attribute
    return $(E', V', \mathbf{u}')$
end function
```

Figure 4. Pseudocode for the atomic unit of computation for a graph network, a *block*. A block takes a graph as input, performs these computations, then outputs an updated graph. A *block* can represent features, clusters, and functions. Blocks can be composed to form multi-block architectures. Figure from [7]. Permission pending.

more generally) are sensitive using external influences. At a theoretical level, reinforcement learning has a substantial literature on its biological realization [49], allowing for testable hypotheses about the neural implementation of structure learning in a reinforcement learning scenario. Finally, the relationship between reinforcement-learning-driven structure learning and behavior can be clearly observable [11].

Limitations and future directions

Structure learning is currently teleologically defined

Currently, structure learning is defined in terms of its functional consequences (i.e. generalization). The problem with a functional definition of a learning phenomenon is that it does not distinguish between causal mechanisms with the same outcome. That said, the reason that we have this teleological definition of structure learning is because the mechanism is not established. This leaves us in a paralyzing (circular) quandary.

For the purpose of a first pass in identifying candidate structure learning computations, we have to deal with false positives (i.e. generalization generated by learning processes other than structure learning). To the extent that false positives contribute to this effort, the case for compositional inductive bias as a necessary computational feature of structure learning is compromised. Integrating over these theoretical perspectives may minimize the influence of net false positives, but, to avoid this issue, it is important to begin to whittle the space of viable hypotheses.

Under the assumption that studying learning dynamics improves the identifiability of candidate computational mechanisms, we need to look at how structure learning evolves to better arbitrate between alternative computational explanations. Moreover, studying both the neural and behavioral dynamics of structure learning may even further reduce the degeneracy problem to better triangulate the viable computational hypotheses.

Structure learning without reinforcement learning

Structure learning in humans is often studied in reinforcement learning contexts [12, 29, 56, 92, 93, 96] for pragmatic reasons, such as the tractability of experimental manipulation and the extent of the current literature on its neuroscientific basis [49]. Further, the rich historical links between reinforcement learning, machine learning, and computer science [85, 86] promote a reinforcement learning-based approach to theoretical development. As a consequence, the algorithms designed to learn structure have often exploited reinforcement learning computational architectures or derivatives thereof [56, 59].

However, structures are not restricted to learning via reinforcement. There is evidence for structure learning in the formation of implicit forward models in motor learning [96] which are thought to be less sensitive to reinforcement learning signals [87]. Further, the formation of learning sets in nonhuman animals spans a set of domains without an established consensus as to their reception to reinforcement learning [35, 76, 95]. To the extent that structure learning exists in learning systems that do not interact with reinforcement learning, this review and its conclusions are limited.

Exploration algorithms guide structured representations and vice versa

To the extent that exploration is a source of data acquisition, it is essential to detecting statistical regularities in our environment. From a computational perspective, exploration balances a bias-variance tradeoff in action selection space, functioning to avoid local minima in the acquisition of information [99] or reward [60]. Forming a global estimate of the environment should aid in adapting to unexpected states with similar properties. Indeed, imbuing an artificial agent with a curiosity signal that rewards reinforcement prediction errors assists structure learning and aids generalization to sparse reward contexts [63]. Similarly, automated structure discovery using intrinsically motivated exploration algorithms results in diverse pattern discovery in self-organizing systems, and adapts using minimal user feedback [70]. The existence of random exploration [99], separable from the directed pursuit of information, suggests a biological imperative to maintain behavioral variability. Broadly, the variability inherent to biological learning systems [20, 81] provides a persistent level of behavioral flexibility; a faint echo of alternatives.

Importantly, emerging evidence provides empirical support for the notion that generalization guides future exploration [77, 100]. Altogether, this suggests a bidirectional relationship between exploration as a source of data for detecting environmental structure and generalization as a sculpting influence on exploration (a direct influence on the data one has to operate over). The computational models considered in this paper do not explicitly address the role of exploration in the discovery of structural form. To the extent that exploration and generalization influence one another, the computational models described here do not capture the dynamics of structure learning.

Conclusions

Predominant computational models of structure learning describing diverse forms of structure discovery prominently feature compositional inductive bias. Computational architectures without this bias are sample inefficient and relatively inflexible in generalization, suggesting the possibility that it is a necessary computational feature. Lastly, compositional inductive bias is determinable: it can be refined for the purpose of experiment and further theoretical development. Thus, a model featuring compositional inductive bias meets the criteria for a minimal model approach to explanation. *Graph networks*, which balance representational structure with representational emergence, enable compositional generalization and present an opportunity to study the hierarchical dynamics of structure learning. Assuming that compositional inductive bias is a necessary feature of structure learning across learning domains, this paper posits that using graph networks in the reinforcement learning context offers a powerful and tractable framework for developing a minimal model of structure learning and furthering our understanding of structure learning as a whole.

Call for a taxonomy of structure learning

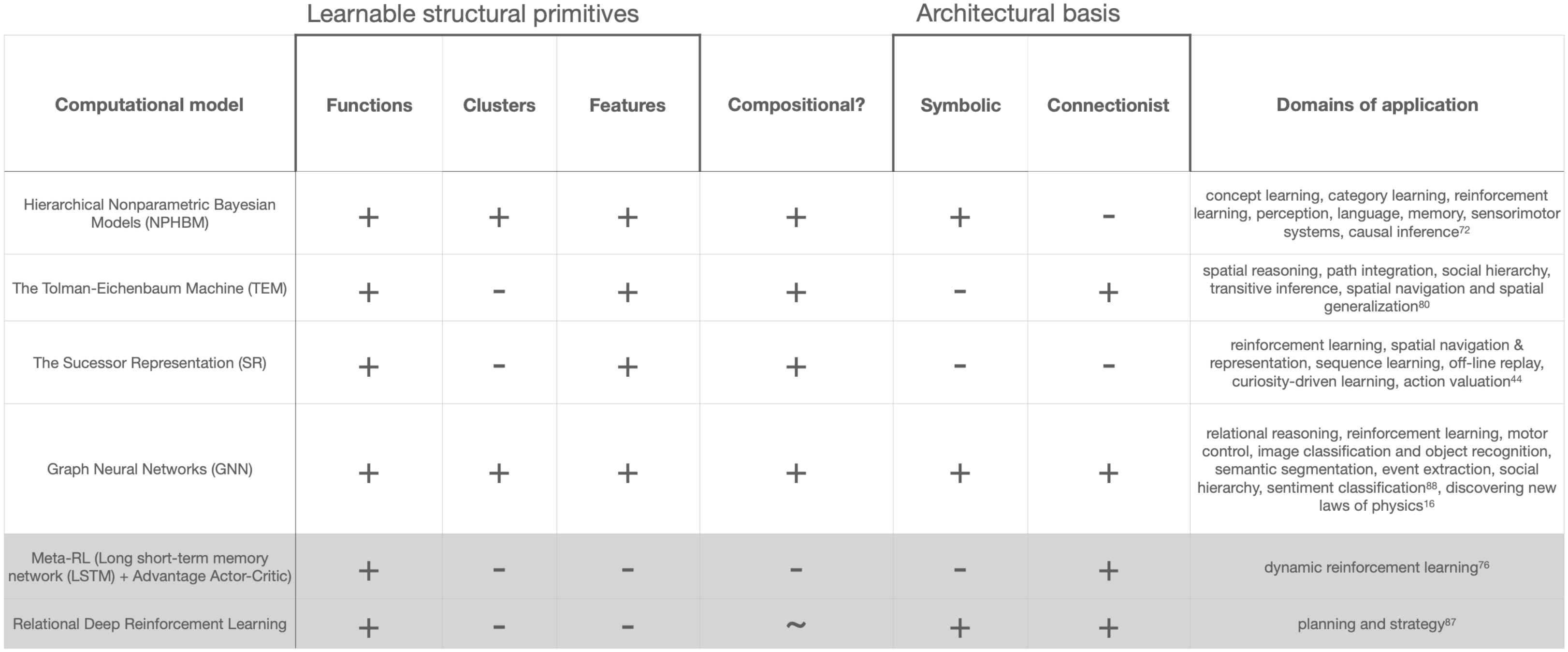
*

Open questions

- How can we interrogate the dynamics of structure learning to improve identifiability?
- What is the relationship between curiosity-driven exploration and structure learning?
- What are the factors that influence the mutual relationship between exploration and structure learning?
- How do structured representations emerge without reinforcement learning?
- What are the limitations of drawing an isomorphism between current analytical approaches and the primitives of structural form?
- Where does the pool of hypotheses regarding the set of structural forms relevant to a given learning problem come from?
- What is the taxonomy of structure learning?

Acknowledgments

I thank Reginald Raye and Roderick Seow for many generative discussions.



	Learnable structural primitives				Architectural basis		
Computational model	Functions	Clusters	Features	Compositional?	Symbolic	Connectionist	Domains of application
Hierarchical Nonparametric Bayesian Models (NPHBM)	+	+	+	+	+	-	concept learning, category learning, reinforcement learning, perception, language, memory, sensorimotor systems, causal inference[72]
The Tolman-Eichenbaum Machine (TEM)	+	-	+	+	-	+	spatial reasoning, path integration, social hierarchy, transitive inference, spatial navigation and spatial generalization[80]
The Sucessor Representation (SR)	+	-	+	+	-	-	reinforcement learning, spatial navigation & representation, sequence learning, off-line replay, curiosity-driven learning, action valuation[44]
Graph Neural Networks (GNN)	+	+	+	+	+	+	relational reasoning, reinforcement learning, motor control, image classification and object recognition, semantic segmentation, event extraction, social hierarchy, sentiment classification[88], discovering new laws of physics[16]
Meta-RL (Long short-term memory network (LSTM) + Advantage Actor-Critic)	+	-	-	-	-	+	dynamic reinforcement learning[76]
Relational Deep Reinforcement Learning	+	-	-	~	+	+	planning and strategy[87]

Figure 5. Comparison of primary computational models. For each of the primary computational models considered, the structural primitives that can be learned, whether the model can be compositional, whether the model has a symbolic and/or connectionist architectural basis, and the set of domains over which the model has been successfully applied is indicated. If the model has a given feature, then the cell is marked with a (+); otherwise the cell is marked with a (-). The grayed models shown in the last two rows merit mentioning, but have relatively limited scope at the time of writing. Superscripts refer to reviews, except in the case of the grayed models shown in the last two rows and in the instance of physical law discovery for GNNs.

Glossary

Structure learning The abstraction of relational invariance from data. Depending on the context, structure learning may be conflated with the idea of learning structural *form*. Functions, features, and clusters are hypothesized to be the three primitive forms of structure. Primitive structures can be combined to compose more complex structures.

form The categorical type of model learned (e.g. function, feature, cluster).

structure A specific instantiation of the form learned. This can also be thought of as parameterizing a model.

data The set of observations from which inferences of structure and form are drawn..

inductive bias The set of assumptions with which a learner approaches a learning problem. In the context of structure learning, inductive bias refers to the set of models considered by the learner that allows them to make adaptive inferences beyond the data they observe. In the context of Bayesian models of cognition, inductive bias may also be called a prior. *Relational inductive bias* refers to assumptions that impose constraints on learned relationships.

compositionality The idea that a complex model or structure can be decomposed into simpler constituent expressions. Similarly, constituent expressions can compose a complex model or structure. Compositionality enables combinatorial generalization, or the capacity to make inferences under novel scenarios by recombining simpler learned relationships.

hypothesis space The set of hypotheses most probably considered as a function of the structural form learned. Note that this concept is linked to C.S. Peirce's notion of *abductive inference*, in which the set of hypotheses considered is constrained by previous belief [64]. Viewed in light of Bayesian theory, abduction can also be conceptualized as a probabilistic form of inductive inference.

sample efficiency The amount of experience (or number of relational observations) needed to learn a given structural form or model.

bias-variance tradeoff Bias is the degree to which a model makes assumptions regarding the structure of the data. The stronger the set of assumptions, the greater the likelihood of underfitting the data. Variance is the degree to which the data informs the model. The greater the variance, the greater the likelihood of overfitting the data. Bias and variance are inversely proportional to one another [19], and a balance between bias and variance is necessary to minimize generalization error.

generalization The use of prior learning to guide learning in situations judged to be similar to previous learning contexts.

Nonparametric Bayesian Hierarchical Models A model that leverages Bayes' rule to learn multiple levels of abstraction with distributions of arbitrary structure. These models infer the complexity of the model from the given data.

nonparametric model A model defined by arbitrary distributions (not confined to a normal distribution)..

parametric model A model defined by distributions with a fixed number of parameters (e.g. mean and variance in the case of a normal distribution).

hierarchical model A model describing multiple levels of organization or abstraction).

prior probability The degree of belief in a hypothesis prior to collecting new observations.

likelihood The probability of observing data if a given hypothesis is correct.

posterior probability The degree of belief in a hypothesis after collecting new observations. This is a function of the *likelihood* and the *prior* as described in Nonparametric Hierarchical Bayesian Models (NPHBM).

Bayes' rule At its heart, simply a formula for calculating the probability of an event given prior knowledge of related conditional probabilities. In cognitive science, this rule has been extended to provide a normative standard for updating beliefs in the face of evidence. Operates over the *likelihood* and the *prior probability* to give a *posterior probability*.

subgoals A learning problem can be decomposed into simpler learning problems, or subgoals. Subgoals can have their own reward functions and policies. Solving subgoals can be an economical way to solve the full learning problem and learning subgoals instead of how to solve the composite learning problem is conducive to compositional generalization. The policies developed to solve subgoals are called options in the reinforcement learning context.

cognitive maps The systematic organization of spatial memory and, more abstractly, the relationships between objects or events.

latent learning A form of learning not immediately expressed but later invoked upon recognition of a similar context with sufficient incentive.

stimulus-response association An automatic, learned behavior tied to a particular physical stimulus. Stands in contrast to latent learning.

successor representation A form of reinforcement learning that predicts transitions between states to estimate the optimal trajectory to reward. To do this, it combines an estimate of how often future states are expected to be visited in the future with an estimate of their reward. Occupies an intermediate space between model-based and model-free reinforcement learning to balance flexibility and computational efficiency.

reinforcement learning The process by which an agent learns to predict the actions that maximize long-run reward given the state of the environment.

model-based A relatively flexible, slow form of reinforcement learning in which action values are recomputed according to an internal model of state transitions. Computationally expensive relative to model-free reinforcement learning.

model-free A relatively inflexible, fast form of reinforcement learning in which the long-run value of actions is precomputed and cached. At the time of a decision, the cached value of an action is retrieved.

temporal difference learning A model-free approach to gradually learning and updating state-action values.

state A complete description of the environment.

Offline replay A memory process in which the hippocampal network generates patterns of activation that represent compressed versions of previous experience while off-task.

artificial neural networks A computational model loosely based on the concept of a network of neurons. Organized in layers of interconnected nodes, with an input layer, one or more hidden layers which operate over their successive inputs, and an output layer. A learning rule modifies the weights of the connections between elements as a function of the inputs to each layer and nodes contain an activation function to determine their output behavior.

recurrent neural network An artificial neural network with nodes that send feedback to one another. This allows previous outputs to be used as inputs, allowing for recent historical information to influence computation.

Long Short-Term Memory A type of recurrent neural network. An LSTM cell can process sequential data. Often a LSTM unit is composed of a cell, an input gate, an output gate, and a forget gate. These gates regulate the passage of information into and out of the cell.

graph neural network Artificial neural networks that operate on graph-structured representations.

graph-structured representations Mathematical structures used to model relationships between nodes. Denoted as a 3-tuple $G = (u, V, E)$, where u is a global attribute, V is the set of nodes, and E is the set of edges connecting the nodes.
nodes The entities considered..
edges The connections between nodes. An edge can represent the relationship between entities, or nodes.
global attribute Graph-level properties. Nodes and edges may also have attributes.
graph network block The elemental unit of computation in a graph network. Takes a graph as input, performs computations on the graph, and then outputs a transformed graph. Graph network blocks can be composed to create multi-block architectures.
multi-block graph network architectures Complex architectures can be *composed* from graph network blocks. In multi-block architectures, the output of one GN block can be passed as the input graph to another GN block, even with different within-block structures. Blocks can have independent functions and/or parameters or reused functions and/or parameters.

References

1. R. E. Ambrose, B. E. Pfeiffer, and D. J. Foster. Reverse replay of hippocampal place cells is uniquely modulated by changing reward. *Neuron*, 91(5):1124–1136, 2016.

2. D. Aronov, R. Nevers, and D. W. Tank. Mapping of a non-spatial dimension by the hippocampal–entorhinal circuit. *Nature*, 543(7647):719–722, 2017.

3. J. Austerweil and T. Griffiths. Learning hypothesis spaces and dimensions through concept learning. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, 2010.

4. J. L. Austerweil, S. J. Gershman, J. B. Tenenbaum, and T. L. Griffiths. Structure and flexibility in bayesian models of cognition. *Oxford handbook of computational and mathematical psychology*, pages 187–208, 2015.

5. A. Baddeley and L. Cermak. Amnesia: A minimal model and an interpretation. *Human Memory and Amnesia, LS*, pages 305–336, 1982.

6. P. Battaglia, R. Pascanu, M. Lai, D. Jimenez Rezende, et al. Interaction networks for learning about objects, relations and physics. *Advances in neural information processing systems*, 29:4502–4510, 2016.

7. P. W. Battaglia, J. B. Hamrick, V. Bapst, A. Sanchez-Gonzalez, V. Zambaldi, M. Malinowski, A. Tacchetti, D. Raposo, A. Santoro, R. Faulkner, et al. Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*, 2018.

8. R. W. Batterman and C. C. Rice. Minimal model explanations. *Philosophy of Science*, 81(3):349–376, 2014.

9. T. Bayes. Lii. an essay towards solving a problem in the doctrine of chances. by the late rev. mr. bayes, frs communicated by mr. price, in a letter to john canton, amfr s. *Philosophical transactions of the Royal Society of London*, (53):370–418, 1763.

10. T. E. Behrens, T. H. Muller, J. C. Whittington, S. Mark, A. B. Baram, K. L. Stachenfeld, and Z. Kurth-Nelson. What is a cognitive map? organizing knowledge for flexible behavior. *Neuron*, 100(2):490–509, 2018.

11. K. M. Bond and J. A. Taylor. Structural learning in a visuomotor adaptation task is explicitly accessible. *Eneuro*, 4(4), 2017.

12. M. Botvinick, S. Ritter, J. X. Wang, Z. Kurth-Nelson, C. Blundell, and D. Hassabis. Reinforcement learning, fast and slow. *Trends in cognitive sciences*, 23(5):408–422, 2019.

13. D. A. Braun, C. Mehring, and D. M. Wolpert. Structure learning in action. *Behavioural brain research*, 206(2):157–165, 2010.

14. N. Chater, J. B. Tenenbaum, and A. Yuille. Probabilistic models of cognition: Conceptual foundations, 2006.

15. A. O. Constantinescu, J. X. O'Reilly, and T. E. Behrens. Organizing conceptual knowledge in humans with a gridlike code. *Science*, 352(6292):1464–1468, 2016.

16. M. Cranmer, A. Sanchez Gonzalez, P. Battaglia, R. Xu, K. Cranmer, D. Spergel, and S. Ho. Discovering symbolic models from deep learning with inductive biases. *Advances in Neural Information Processing Systems*, 33, 2020.

17. P. Dayan. Improving generalization for temporal difference learning: The successor representation. *Neural Computation*, 5(4):613–624, 1993.

18. W. de Cothi, N. Nyberg, E.-M. Griesbauer, C. Ghanamé, F. Zisch, J. Lefort, L. Fletcher, C. Newton, S. Renaudineau, D. Bendor, et al. Predictive maps in rats and humans for spatial navigation. *bioRxiv*, 2020.

19. A. Derumigny and J. Schmidt-Hieber. On lower bounds for the bias-variance trade-off. *arXiv preprint arXiv:2006.00278*, 2020.

20. A. K. Dhawale, M. A. Smith, and B. P. Ölveczky. The role of variability in motor learning. *Annual review of neuroscience*, 40:479–498, 2017.

21. A. D. Ekstrom and C. Ranganath. Space, time, and episodic memory: The hippocampus is all over the cognitive map. *Hippocampus*, 28(9):680–687, 2018.

22. N. T. Franklin and M. J. Frank. Compositional clustering in task structure learning. *PLoS computational biology*, 14(4):e1006116, 2018.

23. V. K. Garg, S. Jegelka, and T. Jaakkola. Generalization and representational limits of graph neural networks. *arXiv preprint arXiv:2002.06157*, 2020.

24. S. J. Gershman. The successor representation: its computational logic and neural substrates. *Journal of Neuroscience*, 38(33):7193–7200, 2018.

25. S. J. Gershman and J. M. Beck. Complex probabilistic inference. *Computational models of brain and behavior*, 453, 2017.

26. S. J. Gershman, D. M. Blei, and Y. Niv. Context, learning, and extinction. *Psychological review*, 117(1):197, 2010.

27. S. J. Gershman, P. I. Frazier, and D. M. Blei. Distance dependent infinite latent feature models. *IEEE transactions on pattern analysis and machine intelligence*, 37(2):334–345, 2014.

28. S. J. Gershman and Y. Niv. Exploring a latent cause theory of classical conditioning. *Learning & behavior*, 40(3):255–268, 2012.

29. S. J. Gershman, K. A. Norman, and Y. Niv. Discovering latent causes in reinforcement learning. *Current Opinion in Behavioral Sciences*, 5:43–50, 2015.

30. S. Goldwater, T. L. Griffiths, and M. Johnson. A bayesian framework for word segmentation: Exploring the effects of context. *Cognition*, 112(1):21–54, 2009.

31. T. L. Griffiths. Understanding human intelligence through human limitations. *arXiv preprint arXiv:2009.14050*, 2020.

32. T. L. Griffiths, F. Callaway, M. B. Chang, E. Grant, P. M. Krueger, and F. Lieder. Doing more with less: meta-reasoning and meta-learning in humans and machines. *Current Opinion in Behavioral Sciences*, 29:24–30, 2019.

33. T. L. Griffiths, N. Chater, C. Kemp, A. Perfors, and J. B. Tenenbaum. Probabilistic models of cognition: Exploring representations and inductive biases. *Trends in cognitive sciences*, 14(8):357–364, 2010.

34. J. B. Hamrick, K. R. Allen, V. Bapst, T. Zhu, K. R. McKee, J. B. Tenenbaum, and P. W. Battaglia. Relational inductive bias for physical construction in humans and machines. *arXiv preprint arXiv:1806.01203*, 2018.

35. H. F. Harlow. The formation of learning sets. *Psychological review*, 56(1):51, 1949.

36. S. Heckers, M. Zalesak, A. P. Weiss, T. Ditman, and D. Titone. Hippocampal activation during transitive inference in humans. *Hippocampus*, 14(2):153–162, 2004.

37. W. Huang, I. Mordatch, and D. Pathak. One policy to control them all: Shared modular policies for agent-agnostic control. *arXiv preprint arXiv:2007.04976*, 2020.

38. G. James, D. Witten, T. Hastie, and R. Tibshirani. *An introduction to statistical learning*, volume 112. Springer, 2013.

39. R. Kaplan, N. W. Schuck, and C. F. Doeller. The role of mental maps in decision-making. *Trends in Neurosciences*, 40(5):256–259, 2017.

40. C. Kemp and J. B. Tenenbaum. The discovery of structural form. *Proceedings of the National Academy of Sciences*, 105(31):10687–10692, 2008.

41. N. Kriegeskorte and P. K. Douglas. Cognitive computational neuroscience. *Nature neuroscience*, 21(9):1148–1160, 2018.

42. J. R. Kubricht, K. J. Holyoak, and H. Lu. Intuitive physics: Current research and controversies. *Trends in cognitive sciences*, 21(10):749–759, 2017.

43. S. Kumar, I. Dasgupta, J. D. Cohen, N. D. Daw, and T. L. Griffiths. Meta-learning of compositional task distributions in humans and machines. *arXiv preprint arXiv:2010.02317*, 2020.

44. T. L Griffiths, C. Kemp, and J. B Tenenbaum. Bayesian models of cognition. 2008.

45. B. M. Lake. *Towards more human-like concept learning in machines: Compositionality, causality, and learning-to-learn.* PhD thesis, Massachusetts Institute of Technology, 2014.

46. B. M. Lake, R. Salakhutdinov, and J. B. Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015.

47. B. M. Lake, T. D. Ullman, J. B. Tenenbaum, and S. J. Gershman. Building machines that learn and think like people. *Behavioral and brain sciences*, 40, 2017.

48. T. Lau, H. T. Pouncy, S. J. Gershman, and M. Cikara. Discovering social groups via latent structure learning. *Journal of Experimental Psychology: General*, 147(12):1881, 2018.

49. D. Lee, H. Seo, and M. W. Jung. Neural basis of reinforcement learning and decision making. *Annual review of neuroscience*, 35:287–308, 2012.

50. Q. Li, F. Gama, A. Ribeiro, and A. Prorok. Graph neural networks for decentralized multi-robot path planning. *arXiv preprint arXiv:1912.06095*, 2019.

51. Y. Liu, R. J. Dolan, Z. Kurth-Nelson, and T. E. Behrens. Human replay spontaneously reorganizes experience. *Cell*, 178(3):640–652, 2019.

52. M. C. Machado, M. G. Bellemare, and M. Bowling. A laplacian framework for option discovery in reinforcement learning. *arXiv preprint arXiv:1703.00956*, 2017.

53. M. C. Machado, C. Rosenbaum, X. Guo, M. Liu, G. Tesauro, and M. Campbell. Eigenoption discovery through the deep successor representation. *arXiv preprint arXiv:1710.11089*, 2017.

54. J. R. Manns and H. Eichenbaum. Evolution of declarative memory. *Hippocampus*, 16(9):795–808, 2006.

55. S. Mark, R. Moran, T. Parr, S. W. Kennerley, and T. E. Behrens. Transferring structural knowledge across cognitive maps in humans and models. *Nature Communications*, 11(1):1–12, 2020.

56. I. Momennejad. Learning structures: Predictive representations, replay, and generalization. *Current Opinion in Behavioral Sciences*, 32:155–166, 2020.

57. I. Momennejad, A. R. Otto, N. D. Daw, and K. A. Norman. Offline replay supports planning: fmri evidence from reward revaluation. *bioRxiv*, page 196758, 2017.

58. I. Momennejad, A. R. Otto, N. D. Daw, and K. A. Norman. Offline replay supports planning in human reinforcement learning. *Elife*, 7:e32548, 2018.

59. I. Momennejad, E. M. Russek, J. H. Cheong, M. M. Botvinick, N. D. Daw, and S. J. Gershman. The successor representation in human reinforcement learning. *Nature Human Behaviour*, 1(9):680–692, 2017.

60. D. J. Navarro, B. R. Newell, and C. Schulze. Learning and choosing in an uncertain world: An investigation of the explore–exploit dilemma in static and dynamic environments. *Cognitive psychology*, 85:43–77, 2016.

61. A. Newell. *Unified theories of cognition.* Harvard University Press, 1994.

62. J. O'keefe and L. Nadel. *The hippocampus as a cognitive map.* Oxford: Clarendon Press, 1978.

63. D. Pathak, P. Agrawal, A. A. Efros, and T. Darrell. Curiosity-driven exploration by self-supervised prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 16–17, 2017.

64. C. S. Peirce. *Collected papers of charles sanders peirce*, volume 2. Harvard University Press, 1960.

65. B. E. Pfeiffer. The content of hippocampal "replay". *Hippocampus*, 30(1):6–18, 2020.

66. B. E. Pfeiffer and D. J. Foster. Hippocampal place-cell sequences depict future paths to remembered goals. *Nature*, 497(7447):74–79, 2013.

67. J. H. Piater and R. A. Grupen. Feature learning for recognition with bayesian networks. In *Proceedings 15th International Conference on Pattern Recognition. ICPR-2000*, volume 1, pages 17–20. IEEE, 2000.

68. Plato, C. J. Emlyn-Jones, and W. Preddy. *Plato, Republic.* Harvard University Press, 2013.

69. J. Rahnama and E. Hüllermeier. Learning tversky similarity. In *International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*, pages 269–280. Springer, 2020.

70. C. Reinke, M. Etcheverry, and P.-Y. Oudeyer. Intrinsically motivated discovery of diverse patterns in self-organizing systems. *arXiv preprint arXiv:1908.06663*, 2019.

71. C. Reverberi, K. Görgen, and J.-D. Haynes. Compositionality of rule representations in human prefrontal cortex. *Cerebral cortex*, 22(6):1237–1246, 2012.

72. G. Rozenberg. *Handbook of graph grammars and computing by graph transformation*, volume 1. World scientific, 1997.

73. E. M. Russek, I. Momennejad, M. M. Botvinick, S. J. Gershman, and N. D. Daw. Predictive representations can link model-based reinforcement learning to model-free mechanisms. *PLoS computational biology*, 13(9):e1005768, 2017.

74. A. N. Sanborn, T. L. Griffiths, and D. J. Navarro. Rational approximations to rational models: alternative algorithms for category learning. *Psychological review*, 117(4):1144, 2010.

75. F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini. The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1):61–80, 2008.

76. A. M. Schrier. Learning how to learn: The significance and current status of learning set formation. *Primates*, 25(1):95–102, 1984.

77. E. Schulz, R. Bhui, B. C. Love, B. Brier, M. T. Todd, and S. J. Gershman. Structured, uncertainty-driven exploration in real-world consumer choice. *Proceedings of the National Academy of Sciences*, 116(28):13903–13908, 2019.

78. E. Schulz, F. Quiroga, and S. J. Gershman. Communicating compositional patterns. *Open Mind*, 4:25–39, 2020.

79. E. Schulz, J. Tenenbaum, D. K. Duvenaud, M. Speekenbrink, and S. J. Gershman. Probing the compositionality of intuitive functions. *Advances in neural information processing systems*, 29:3729–3737, 2016.

80. E. Schulz, J. B. Tenenbaum, D. Duvenaud, M. Speekenbrink, and S. J. Gershman. Compositional inductive biases in function learning. *Cognitive psychology*, 99:44–79, 2017.

81. R. B. Stein, E. R. Gossen, and K. E. Jones. Neuronal variability: noise or part of the signal? *Nature Reviews Neuroscience*, 6(5):389–397, 2005.

82. I. Stoianov, D. Maisto, and G. Pezzulo. The hippocampal formation as a hierarchical generative model supporting generative replay and continual learning. *bioRxiv*, 2020.

83. M. Stolle and D. Precup. Learning options in reinforcement learning. In *International Symposium on abstraction, reformulation, and approximation*, pages 212–223. Springer, 2002.

84. R. S. Sutton. Dyna, an integrated architecture for learning, planning, and reacting. *ACM Sigart Bulletin*, 2(4):160–163, 1991.

85. R. S. Sutton and A. G. Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

86. R. S. Sutton, A. G. Barto, et al. *Introduction to reinforcement learning*, volume 135. MIT press Cambridge, 1998.

87. J. A. Taylor, J. W. Krakauer, and R. B. Ivry. Explicit and implicit contributions to learning in a sensorimotor adaptation task. *Journal of Neuroscience*, 34(8):3023–3032, 2014.

88. J. B. Tenenbaum, C. Kemp, T. L. Griffiths, and N. D. Goodman. How to grow a mind: Statistics, structure, and abstraction. *science*, 331(6022):1279–1285, 2011.

89. D. G. R. Tervo, J. B. Tenenbaum, and S. J. Gershman. Toward the neural implementation of structure learning. *Current opinion in neurobiology*, 37:99–105, 2016.

90. E. C. Tolman. Cognitive maps in rats and men. *Psychological review*, 55(4):189, 1948.

91. S. W. Townsend, S. Engesser, S. Stoll, K. Zuberbühler, and B. Bickel. Compositionality in animals and humans. *PLoS Biology*, 16(8):e2006425, 2018.

92. J. X. Wang, Z. Kurth-Nelson, D. Kumaran, D. Tirumala, H. Soyer, J. Z. Leibo, D. Hassabis, and M. Botvinick. Prefrontal cortex as a meta-reinforcement learning system. *Nature neuroscience*, 21(6):860–868, 2018.

93. J. X. Wang, Z. Kurth-Nelson, D. Tirumala, H. Soyer, J. Z. Leibo, R. Munos, C. Blundell, D. Kumaran, and M. Botvinick. Learning to reinforcement learn. *arXiv preprint arXiv:1611.05763*, 2016.

94. T. Wang, R. Liao, J. Ba, and S. Fidler. Nervenet: Learning structured policy with graph neural networks. In *International Conference on Learning Representations*, 2018.

95. J. Warren and A. Baron. The formation of learning sets by cats. *Journal of Comparative and Physiological Psychology*, 49(3):227, 1956.

96. A. Weinstein and M. M. Botvinick. Structure learning in motor control: A deep reinforcement learning model. *arXiv preprint arXiv:1706.06827*, 2017.

97. J. C. Whittington, T. H. Muller, S. Mark, G. Chen, C. Barry, N. Burgess, and T. E. Behrens. The tolman-eichenbaum machine: Unifying space and relational memory through generalization in the hippocampal formation. *Cell*, 2020.

98. W. Wiese. The science of consciousness does not need another theory, it needs a minimal unifying model. *Neuroscience of Consciousness*, 2020(1):niaa013, 2020.

99. R. C. Wilson, A. Geana, J. M. White, E. A. Ludvig, and J. D. Cohen. Humans use directed and random exploration to solve the explore–exploit dilemma. *Journal of Experimental Psychology: General*, 143(6):2074, 2014.

100. C. M. Wu, E. Schulz, M. Speekenbrink, J. D. Nelson, and B. Meder. Generalization guides human exploration in vast decision spaces. *Nature human behaviour*, 2(12):915–924, 2018.

101. X. Wu and D. J. Foster. Hippocampal replay captures the unique topological structure of a novel environment. *Journal of Neuroscience*, 34(19):6459–6469, 2014.

102. C. D. Wynne. A minimal model of transitive inference. *Models of action*, pages 269–307, 1998.

103. L. Xia and A. G. E. Collins. Temporal and state abstractions for efficient learning, transfer and composition in humans. *bioRxiv*, 2020.

104. K. Xu, W. Hu, J. Leskovec, and S. Jegelka. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*, 2018.

105. K. Yoon, R. Liao, Y. Xiong, L. Zhang, E. Fetaya, R. Urtasun, R. Zemel, and X. Pitkow. Inference in probabilistic graphical models by graph neural networks. In *2019 53rd Asilomar Conference on Signals, Systems, and Computers*, pages 868–875. IEEE, 2019.