CHAPTER SIX

Learning structured representations from experience

Leonidas A.A. Doumas[a,*], **Andrea E. Martin**[b]
[a]University of Edinburgh, Edinburgh, United Kingdom
[b]Max Planck Institute for Psycholinguistics, Nijmegen, The Netherlands
*Corresponding author: e-mail address: alex.doumas@ed.ac.uk

Contents

Abstract

How a system represents information tightly constrains the kinds of problems it can solve. Humans routinely solve problems that appear to require structured representations of stimulus properties and the relations between them. An account of how we might acquire such representations has central importance for theories of human cognition. We describe how a system can learn structured relational representations from initially unstructured inputs using comparison, sensitivity to time, and a modified Hebbian learning algorithm. We summarize how the model DORA (*Discovery of Relations by Analogy*) instantiates this approach, which we call *predicate learning*, as well as how the model captures several phenomena from cognitive development, relational

Psychology of Learning and Motivation, Volume 69
ISSN 0079-7421
https://doi.org/10.1016/bs.plm.2018.10.002

reasoning, and language processing in the human brain. Predicate learning offers a link between models based on formal languages and models which learn from experience and provides an existence proof for how structured representations might be learned in the first place.

Humans routinely draw inferences based on relations—from the mundane ("my kid won't eat a portion that big"), to the sublime ("the cardinal number of the reals between 0–1 is larger than the cardinal number of the positive integers")—and relational reasoning is an important contributor to abilities such as analogical reasoning (e.g., Holyoak & Thagard, 1995), categorization (e.g., Medin, Goldstone, & Gentner, 1993), concept learning (e.g., Doumas & Hummel, 2013), visual cognition (e.g., Biederman, 1987; Hummel, 2013), and language (e.g., Gentner, 2010; Martin, 2016; Martin & Doumas, 2017). In fact, the capacity to represent and reason about relations has been posited as the key difference between human and non-human animal cognition (Penn, Holyoak, & Povinelli, 2008).

Perhaps the most plausible explanation of how humans are able to reason relationally is that we can represent relations as abstract structures that take arguments—i.e., as predicates (see, e.g., Holyoak, 2012; Holyoak & Hummel, 2000). A predicate is a symbolic (or structured) representational element that can be dynamically bound to an argument, specifying some property about that argument (see, e.g., Doumas & Hummel, 2005). Systems based on predicates or formally identical representations (e.g., labeled graphs) are powerful and successfully account for many aspects of human cognition (see Bringsjord, 2008 for a review).

Structured representations are frequently associated with traditional symbolic architectures (e.g., production systems, or models based on labeled graphs). While such models have successfully accounted for a wide range of human cognitive phenomena (e.g., Anderson, 2009) they have also been widely criticized. First, providing an account of how these representations are learned from experience has proven difficult (e.g., Kriete, Noelle, Cohen, & O'Reilly, 2013; Leech, Mareschal, & Cooper, 2008; Rogers & McClelland, 2008). Models that rely on structured representations either make strong nativist claims, positing that a large set of representational elements and rules for building compositions of these elements are innate, or, at the very least, require that the powerful representations that they use are hand-coded by the modeler (e.g., Goodman, Ullman, & Tenenbaum, 2011; Kemp & Tenenbaum, 2009; Lake, Salakhutdinov, & Tenenbaum, 2015). The lack of an account of how structured representations might

be learned from experience has been levied as one of the fundamental limitations of the symbolic approach to understanding cognition (e.g., Leech et al., 2008; O'Reilly & Busby, 2002; O'Reilly, Busby, & Soto, 2003; Rogers & McClelland, 2008; Rumelhart, McClelland, & The PDP Research Group, 1986). Second, traditional symbolic models do not capture the flexibility or semantic richness of human cognition (e.g., Rogers & McClelland, 2008; Rumelhart et al., 1986).

We argue that if structured representations are carefully considered, it becomes clear that these shortcomings are properties of the implementation of traditional symbolic systems rather than of structured representations per se. Moreover, implementing structured representations in other types of architectures—such as connectionist systems that exploit distributed representations—allows these limitations to be overcome.

It is important to be clear about what we mean by structured, and unstructured (or holistic), representations. To start, representations are constructions that explicitly carry information (that is to say, a representation is any structure that contains information about something else). For example, the word "cat" is a representation of a cat, a painting of an orange is a representation of that orange, and a mathematical equation is a representation of a particular system. The human mind, like the mediums of language and canvas, regularly produces states that hold information (e.g., I can have a thought about my mother that carries some information about her). A representational *system* minimally consists of two things: (i) representational elements (e.g., symbols in a symbolic model or nodes in a neural network) and (ii) a set of rules or processes for producing new states from existing states (see Markman, 1999).

A *structured* (or symbolic) representational system must both include a set of representational elements and must support the composition—or binding—of basic representational elements into more complex structures. That is, it must make it possible to form an open-ended set of relational statements with a finite vocabulary of representational elements (Peirce, 1879/1903).

Two common instantiations of symbolic systems—propositional notation and labeled graphs—illustrate these two representational requirements. Propositional notation generally takes a form, *predicate* (argument$_1$, argument$_2$,… argument$_n$), where *predicate* specifies some property or relation, arguments 1…*n* specify the arguments of that predicate, and the arrangement of the arguments within the parentheses specifies the binding information. For example, *told* (Mary, Sally, secret) specifies that Mary told

Sally a secret (i.e., that Mary is bound to the *teller* role, Sally to the *receiver* role, and secret to the *told-item* role). Likewise, labeled graphs specify the same information as propositional notation (i.e., the two systems are isomorphic), only they do so in graphical form. In this case nodes represent predicates and their arguments, and arcs represent the bindings of arguments to roles of the predicate.

Note, that in structured systems the representational elements and the binding mechanism are independent (see Doumas, Hummel, & Sandhofer, 2008; Hummel, 2010; von der Malsburg, 1986, 1999). That is, the mechanism that carries the binding information (e.g., order in the case of propositional notation, and arcs in labeled graphs) is independent of the representational elements that specify the identity of the specified objects and predicates. This independence is an important formal property of that must be generalized to any implementation in order to achieve symbolic competence. For example, the representational elements *fur-covered* and cat, and *bald* and dog might be bound to form the propositions *fur-covered* (cat) and *bald* (dog). While the statement *fur-covered* (cat) has meaning (a cat that has the property of having fur) the elements *fur-covered* and cat retain their meaning when bound.[a] That is, the predicate *fur-covered* means the same thing whether it is bound to "cat," "dog," or "automobile."[b] As described below, that representational elements maintain their meaning across bindings is an important property of symbol systems that underlies fundamental cognitive operations such as relational reasoning and language processing.

In addition to the representational requirements, structured representational systems must also include processes that generate new representations from existing representations. These processes are usually instantiated as a set of inference rules (for propositional systems) or graph matching operations (for labeled graphs). These operations allow a given system the power to

[a] Importantly, the binding tag (the signal carrying the binding information) must also be dynamic. That is, it must allow bindings to be created and destroyed on the fly. For instance, if the cat in the above example gets a an extreme hair cut the binding of *fur-covered* and cat must be broken, and "cat" must be bound instead to the *bald* predicate to form *bald* (cat) where the exact same representational element coding for *bald* in *bald* (dog) is bound to the exact same representational element coding for the cat in *fur-covered* (cat).

[b] We are not arguing that there are no "shades-of-meaning" in predicate representations. A predicate like *furry* might mean different things in the context of a dog or a toy ball. We are only arguing that there is some core aspect of the concept that remains true across all instances of *furry* (e.g., having some hair-covering; see also Doumas & Hummel, 2005, 2012; Holyoak & Hummel, 2000; Hummel, 2017).

infer new informational states from existing informational states, and to perform operations in the service of solving problems (e.g., Anderson, 2009).

To summarize, for a representational system to be considered structured, it must include: (i) representational elements that carry identity information (specifying what elements are present in a given situation), (ii) a mechanism that carries binding information (specifying how those elements are arranged), and (iii) processes by which new representational structures are produced from existing structures. In addition, these three sources of information must be independent (e.g., the binding mechanism must be independent of the elements so bound; Doumas & Hummel, 2012; Hummel, 2010). It follows, then, that to support structured representations, a system must include, at minimum, three independent sources of information, or *informational degrees of freedom* (see also Doumas & Hummel, 2012). Moreover, any system that can represent three independent sources of information can, at least in principle, support structured representations.

By contrast, in systems that use holistic encoding of information, knowledge is represented not as discrete representational elements that enter into bound compositions, but as patterns of activation distributed over processing elements. Representations in a holistic system are formally equivalent to activation vectors, with each element in the vector corresponding to the activation of a specific processing element (see, e.g., Jordan, 1986). This property is well illustrated by connectionist networks (the most common instantiation of holistic systems). In connectionist models, representations are distributed in the sense that (a) any single concept is represented as a pattern (i.e., vector) of activation over many elements ("nodes" or "units" which are typically assumed to correspond roughly to neurons or small collections of neurons), and (b) any single element will likely participate in the representation of many different concepts. As a result, two patterns of activation will tend to be similar to the extent that they represent similar concepts. For example, consider a very simple network. One pattern of activation (e.g., [0,0,1,1,0]) might represent the concept "rose," while another pattern (e.g., [0,0,1,1,1]) represents the concept "daisy." Other concepts like "animal," "large," or "food" would be represented as different patterns again. Holistic representations are often distributed, but they need not be (e.g., Bowers, 2017).

Holistic representations may also be localist (Hummel, 2017). Holistic representations are limited because, by design, they do not capture symbolic structure (see Doumas & Hummel, 2005; Marcus, 2003). For example, in a structured representation like *brown* (cow) the predicate *brown* qualifies the

cow, but the representation is easily decomposable to its individual components, brown-ness and cow. The mechanism for linking *brown* and cow keeps the components independent. By contrast, to represent brown cow in a holistic system, the representation for brown, say [0 0 0 1 0 1 0], is superimposed with the representation for cow, say [1 1 0 0 0 0 0], to form a new pattern, here [1 1 0 1 0 1 0]. It is impossible from the resulting pattern to know what patterns were combined to create it (e.g., it could just as easily have been the vectors [1 0 0 1 0 0 0] and [0 1 0 0 0 1 0] that were combined).

However, holistic representations are quite powerful. First, holistic representations do capture many aspects of human cognition, particularly those that occur automatically or during reflexive reasoning. Second, learning new representations in such systems is reasonably straightforward: amounting to learning associations between elements (or patterns of activation). For example, to learn that strawberries are red, the system must simply associate the representations of strawberries with red.

By contrast, accounting for how structured representations themselves are acquired is a much thornier question. In the following we review the properties of structured—specifically predicate—representations that make them hard to learn and give an overview of the previous work in the domain of predicate learning. Next, we outline the set of problems that must be solved in the service of learning abstract predicate representations from examples. We then describe a solution to these problems instantiated in a computational model.

1. Why are predicates hard to learn?

As described above, predicates—and other structured representations—as they are instantiated in theories of human mental representations have two fundamental attributes (e.g., Doumas & Hummel, 2005, 2012). First, a predicate specifies a property (or set of properties) about its argument in a manner that is invariant with that argument. Second, bindings between predicates and their arguments are dynamic—that is, they can be created and destroyed on the fly.

As a result of these properties, structured representations are powerful (see above). However, these properties that make relations so powerful are also, in large part, what make accounting for how they are learned so difficult. While our representation of a relation like *taller* (x,y) is completely independent of any specific x or y, the instances from which we learn that relation in the world are exquisitely tied to very specific objects—that is, we

do not get to experience instances of disembodied *taller*-ness in our environment. Every instance of *taller* that we experience in the world involves some specific object of a greater height than some other specific object. Predicates in the form that we end up representing them (abstract and invariant structures) just are not "out there" in the world of concrete and context-laden instances.

2. Current approaches to the problem of learning predicates

While the symbolic approach to cognition dominated the field for decades, the difficulty in providing an account of how structured representations are learned led to a tension in the field (see, e.g., Franklin, 1999). Approaches to modeling human cognition now tend to fall into one of two camps. On the one hand, the connectionist (or eliminative) approach explicitly eschews predicates, solving the problem of where predicates come from by rendering it moot. On the other hand, current symbolist approaches make—either explicitly or implicitly—strong nativist claims about the origins of knowledge. We now outline these two approaches as well as a third more recent approach focused on attempting to address the problem of learning structured representations from experience.

2.1 Eliminative connectionism: "We do not represent predicates"

The lack of an account of where symbols come from was one of the initial impetuses of the development of the connectionist approach, and it remains one of the primary motivations of more modern connectionist approaches (Leech et al., 2008; O'Reilly & Busby, 2002; O'Reilly et al., 2003; Rogers & McClelland, 2008). In response to the question of how symbolic representations are learned to begin with, the prevailing answer in the connectionist tradition has been: They aren't (e.g., McClelland & Cleeremans, 2009).

Traditional connectionist models operate at the so-called sub-symbolic level. Connectionist representations are holistic, realized in patterns of activation in the system rather than structured. Traditional connectionist models explicitly eschew structured representations, and the lack of structured representations in these models was one of the core principles of the connectionist approach from its inception (see Rumelhart et al., 1986).

The persistence of the traditional connectionist approach is due, at least in part, to its wild successes (including those of its more recent offshoots, e.g., deep learning; e.g., LeCun, Bengio, & Hinton, 2015). Connectionist models have been used to simulate a wide range of human cognitive phenomena from perceptual inference (Usher & McClelland, 2001) to language processing (Christiansen & Chater, 2001) to strategic video-game playing (Mnih et al., 2015).

However, connectionist models have had limited success in domains like analogy-making and relational reasoning (Gentner & Forbus, 2011). The lack of symbolic representations in traditional connectionist systems may impose important restrictions in their capacity to simulate human level cognition. Indeed, some aspects of human cognition seem to require symbolic representations (e.g., solving cross-mappings [analogies where the relations point to one mapping, and the literal features of objects point to an orthogonal mapping]; e.g., Holyoak & Thagard, 1995; or integrating multiple relations when making an analogy; e.g., Spellman & Holyoak, 1992, 1996). Models without symbolic representations have repeatedly and systematically failed on these kinds of tasks, leading to arguments that systems without symbolic capacities are patently insufficient to account for the entirety of human cognition (e.g., Fodor & Pylyshyn, 1988; Holyoak & Hummel, 2000; Lake, Ullman, Tenenbaum, & Gershman, 2017; Marcus, 1998; Pinker & Prince, 1988).

2.2 Classic and neo-classic symbolism: "Necessary predicates are innate"

Unsurprisingly, models that use structured (i.e., symbolic) representations have had success in accounting for aspects of human cognition that traditional connectionist models have struggled with. For example, models with production system architectures like ACT (e.g., Taatgen & Anderson, 2008) have successfully accounted for a range of phenomena including problem solving, memory retrieval, and parsing of long-distance dependencies in sentence processing, while models such as SME (Falkenhainer, Forbus, & Gentner, 1989), STAR (Halford, Wilson, & Phillips, 1998, 2010), and LISA (Hummel & Holyoak, 1997, 2003) successfully account for many phenomena from the literature on human relational reasoning.

More recently, Bayesian models have come to the fore as accounts of human cognition. Bayesian models of concept learning generally follow a learning-by-hypothesis-testing framework (e.g., Goodman et al., 2011; Kemp & Tenenbaum, 2009; Lake et al., 2015; for a notable counterexample,

however, see Lu, Chen, & Holyoak's, 2012, BART model). In these models, the system starts with a large set of representations and rules for combining them, and then learns combinations of these elements that best fit a given set of data. For example, in Lake et al.'s (2015) model of letter recognition, the system starts with representations of all possible line segments that could be used to construct a letter in any possible alphabet set, along with predicate representations for *connected-at-top* (*x*, *y*), *connected-at-bottom* (*x*, *y*), and *connected-at-middle* (*x*, *y*). The model might learn that the representation *connected-at-top* (squiggle1, squiggle4) is a legal letter in an alphabet, but all of the elements of the representation are present in the model before any learning occurs. Similarly, in Kemp and Tenenbaum's (2009) account of relational concept learning, the model begins with all possible graph primitives and rules for combining them. The model might learn that a line of nodes is the best way to represent the legal leanings of supreme court justices, or that a hierarchical tree best describes inheritance in object-oriented programming languages, but, again, all those structures are represented by the model before any actual learning takes place.

These models do not provide an account of how the representations that they use might be learned in the first place. They require that the modeler hand code or pre-specify the representational structures that the model exploits, and, as a consequence, make very strong, though often implicit, nativist assertions.

2.3 Learning structured representations from experience

A complete account of how we learn structured representations of abstract relations from experience entails solving three problems. First, the perceptual/cognitive system must learn to detect the basic featural *invariants* that remain constant across instances of the relation. That is, the perceptual system must deliver, or learn to deliver, an invariant response to instances of the relation to be learned. Second, the system must isolate those invariants from the other properties of the objects engaged in the relation to be learned. That is, given an activation vector delivered by the perceptual system in response to a stimulus (e.g., a visual scene wherein a table is larger than a hammer), the system must isolate those properties that code for the relation to be learned (e.g., the system must be able to separate the invariant features of *larger* from those encoding the rest of the scene). Third, the system must learn a *predicate* representation of the critical invariant properties—that is, it must come to represent those invariant properties as an explicit entity that can be bound

to arbitrary and novel arguments while remaining independent of those arguments.

Some models have attempted to account for the origins of abstract symbolic representations without positing innate sets of structured representations. In particular, Lu et al.'s (2012) BART model and Doumas et al.'s (2008) DORA model have been proposed as accounts of how relational representations might be learned from experience, rather than pre-specified.

BART (Lu et al., 2012) begins with feature lists generated by human subjects or via crawls of corpora. BART is then given pairs of items representing a particular relation (e.g., *bigger*). BART learns what amounts to second-order probability distributions over the features of the objects involved in particular roles (e.g., the *larger* and *smaller* roles of *bigger* (x,y)) using Bayes rule and some very general learning priors. Essentially, BART finds properties associated with items in the world that embody particular relations.

The representations that BART learns are sufficient to solve a wide range of analogy problems, but the system struggles with some important edge cases. In particular, because the model represents relations as weights in feature space and not in terms of any specific invariant properties, it has difficulty with full relational reasoning. For example, the model struggles to predicate that an atom can be *bigger* than something when the system has never experienced an instance in which an atom was bigger than something else. The model also has some limitations in reasoning about counterfactuals, and reasoning outside of training range (Lu et al., 2012). However, while the representations that the model learns have some limitations compared to more powerful structured systems (e.g., graph grammars; Kemp, 2012), it makes a serious effort to account for the development of analogy making with minimal assumptions about the starting representations of the learning system.

In a similar vein, the DORA model (Doumas et al., 2008) provides an account for how structured representations (i.e., predicates) are acquired from unstructured representations (i.e., feature vectors). DORA begins with representations of objects as flat feature vectors. DORA uses a process of comparison-based feature extraction and self-supervised learning routines to learn structured representations of any invariant properties from the learning instances that it receives. For example, given a number of examples of instances where one object is larger than another object, DORA can extract and learn a structured representation of the invariant properties of all those larger things (e.g., it learns a predicate for *larger* from the properties that are invariant across instances of one larger and one smaller

object). The resulting representations support successful reasoning in a wide range of relational tasks (see, e.g., Doumas & Hummel, 2010; Doumas et al., 2008; Martin & Doumas, 2017; Morrison, Doumas, & Richland, 2011; Son, Doumas, & Goldstone, 2010).

While the representations that DORA learns do not have the limitations of BART's representations (Doumas et al., 2008), DORA does have the important limitation that it simply assumes a capacity to detect a set of invariant features that underlie the abstract concepts that it learns, although more recent versions of the model have begun to address this limitation by learning representations of relations from pixel images (e.g., Doumas, Hamer, Puebla, & Martin, 2017).

In the following we provide an overview of the DORA model. We focus on how the model learns structured representations from examples. We then review a number of simulations that demonstrate the symbolic capacity of the resulting representations, as well as how the representations and the learning trajectory that produces them allow the model to account for a wide range of phenomena from the literature in child and adult reasoning.

3. Overview of the DORA model

DORA is a model of how structured (i.e., functional predicate) representations can be learned from unstructured representations without assuming an a priori set of structured predicates, or an innate explicit formal language. That is, DORA provides an account of how structured, symbolic, representations can be learned from scratch.

DORA is descended from Hummel and Holyoak's (1997, 2003) LISA model. LISA is a highly successful model of many high-level cognitive phenomena, and DORA is a model of how the powerful representational currency upon which LISA's reasoning is based (LISAese) can be learned in the first place. As DORA learns, it eventually becomes a version of the LISA model (i.e., once it has learned a set of representations, it operates like a version of LISA with those representations). To date, LISA and DORA have been used to account for over 50 empirical phenomena in domains such as analogy, memory retrieval, inductive generalization, structured representation learning, concept development, the development of analogical reasoning, and the development of object recognition (e.g., Doumas & Hummel, 2010; Doumas et al., 2008; Hummel & Holyoak, 1997, 2003; Livins, Spivey, & Doumas, 2015; Martin & Doumas, 2017; Morrison et al.,

2011, 2004; Son et al., 2010). In the following section we cover DORA's basic operations in broad strokes. Details of the model can be found in Doumas et al. (2008).

3.1 Representation in DORA

DORA begins with representations of objects coded as flat feature vectors (see Fig. 3A). These representations are similar to representations in traditional connectionist architectures where elements are coded by distributed collections of units. For example, DORA might represent a toy ball with a node connected to a set of features (Fig. 1A).[c] In short, DORA begins with objects coded conjunctively by flat feature vectors. (In terms of cortical computation, feature nodes can be thought of as aggregate units, perceptual representations, or activation states over networks.)

DORA learns representations of a form we call LISAese (Fig. 1B and C). Full propositions in LISAese are coded by layers of units in a connectionist computing framework (Fig. 1B). At the bottom of the hierarchy, semantic (or feature) nodes (small circles in Fig. 1B) code for the featural properties of represented instances in a distributed manner. At the next layer, localist predicate and object units (POs; triangles and large circles in Fig. 1B) conjunctively code collections of semantic units into representations of objects and roles. At the next layer localist role-binding units (RBs; rectangles in Fig. 1B) conjunctively bind object and role POs into linked role-filler pairs. Finally, proposition units (Ps; ovals in Fig. 1B) link RBs to form whole relational structures.

As an example, consider a LISAese representation of the proposition *bigger* (cup, ball), as depicted in Fig. 1B. PO units representing the relational roles *larger* and *smaller*, and the fillers cup and ball, are connected to semantic units coding their semantic features. At the next layer of the network, RB units conjunctively connect a specific role to a specific filler. Specifically, one RB unit conjunctively connects cup to *bigger*, and one conjunctively connects ball to *smaller*. At the top of the hierarchy, a P unit links the RBs

[c] While we use labels for semantic units, the specific content of the units coding for a property are unimportant to DORA (in fact, the model is actually unaware of the labels given to semantic units). So long as there is something common across the units representing a set of objects, DORA can learn an explicit representation of this commonality. That is, for the purposes of DORA's learning algorithm, all that matters is there is something invariant across instances of a *container* (which there must be for us to learn the concept), and that the perceptual system is capable of responding to this invariance (which, again, there must be for us to respond similarly across instances of containment in the world; for a more complete discussion of the role of invariance in perception see, e.g., Biederman, 1987; Kellman, Burke, & Hummel, 1999).

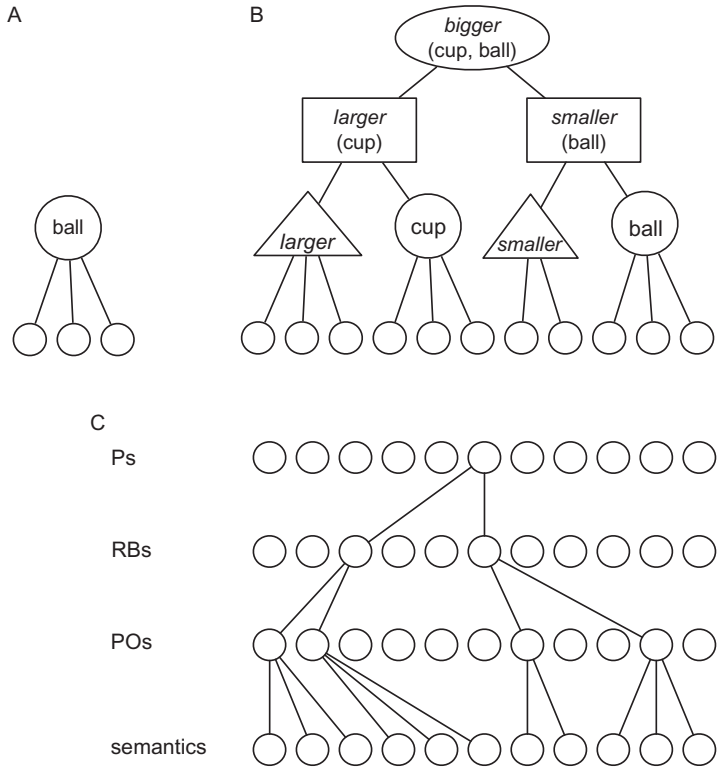

Fig. 1 Representations in DORA. (A) DORA's starting state. DORA begins with representations of objects connected to lists of their features. (B) LISAese representation of the proposition *bigger* (cup, ball). DORA learns full LISAese representations from examples of representations like those in (A). We use larger circles, triangles, rectangles, and ovals for the purposes clearly differentiating units in different layers of the network. (C) More conventional depiction of a LISAese proposition. The proposition is instantiated in layers of bidirectionally connected nodes.

representing *larger*+cup and the RB representing *smaller*+ball to form a whole relational structure. The entire hierarchy of units then encodes the relational proposition *bigger* (cup, ball).

While we use different shapes (e.g., large circles, triangles, rectangles, and ovals) to indicate nodes at different layers, these are not different types of nodes. We use different shapes solely for the purpose of clarifying different layers of nodes. The same proposition can be represented in a more traditional format as layers of bidirectionally connected nodes (Fig. 1C).

3.2 Computational macrostructure

Propositions in DORA are divided into four mutually exclusive sets (Fig. 2): the *driver*, the *recipients*, *long-term-memory* (LTM), and the *emerging recipient* (EM). Each set consists of a layered network coding for POs, RBs, and Ps (i.e., there are specific layers coding for POs, RBs, and Ps in the driver, and another set of layers coding for POs, RBs, and Ps in the recipient). Semantic units are common across all networks (i.e., driver, recipient, LTM, and EM units are connected to the same pool of semantic units).

An *analog* in DORA is a complete story, event, or situation. Analogs are represented by a collection of token (P, RB and PO) units that together represent the propositions in that analog. While token units are not duplicated within an analog (e.g., within an analog, each proposition that refers to Don connects to the same "Don" unit), separate analogs have non-identical token units (e.g., Don will be represented by one PO unit in one analog and by a different PO in another analog). All analogs are connected to the same pool of semantic units. The semantic units thus represent general type information and token units represent instantiations of those things in specific analogs (Hummel & Holyoak, 1997, 2003). For example, if in some analog, the token (PO) unit "Fido" is connected to the semantics "animal," "dog," "furry" and "Fido," then it is a token of an animal, a dog, a furry thing and of the particular dog Fido.

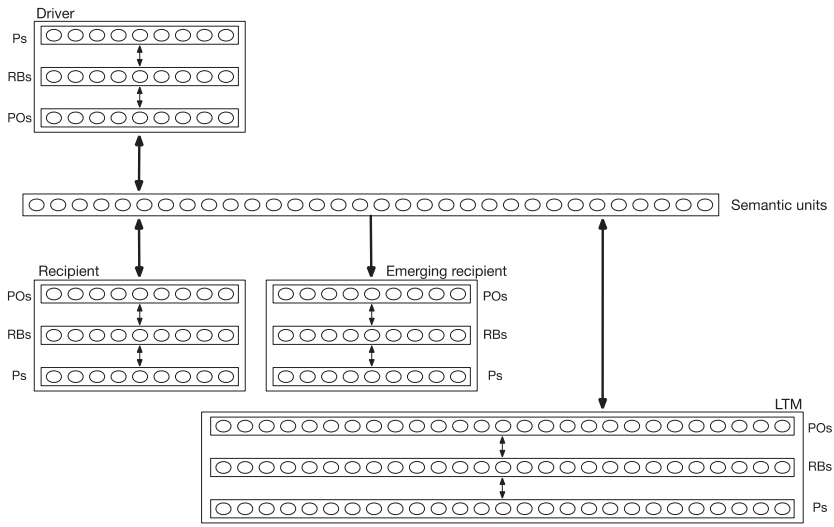

Fig. 2 DORA's computational macrostructure.

3.3 Basic processing

The driver controls the flow of activation in DORA and corresponds to DORA's current focus of attention. Units in the driver pass activation to the semantic units. Because the semantic units are shared by propositions in all sets, activation flows from the driver to propositions in the other three sets. All of DORA's operations (i.e., *retrieval*, *mapping*, *predicate learning*, *relation formation*, *schema induction*, and *generalization*) proceed as a product of the units in the driver activating semantic units, which in turn activate units in the various other sets (as detailed below). During *retrieval*, patterns of activation generated on the semantic units by units in the driver activate representations in LTM, which are retrieved into the recipient. Propositions in the recipient are available for *mapping* onto propositions in the driver. Active driver units activate corresponding (i.e., semantically similar) units in the recipient, allowing DORA to learn mapping connections between them. Mappings between units in the driver and recipient are the basis of DORA's ability to *learn new predicate representations* and to *form higher arity structures* from lower arity structures. Finally, DORA can *learn schemas* from mapped propositions in the driver and recipient, which are encoded into the EM, and may subsequently be encoded into LTM and later enter the driver or recipient (we discuss all of these processes in much more detail below).

When a proposition in the driver becomes active, role-filler bindings must be represented dynamically on the units that maintain role-filler independence (i.e., POs and semantic units; see Doumas & Hummel, 2005; Doumas et al., 2008; Hummel & Holyoak, 1997, 2003). In DORA, roles are dynamically bound to their fillers by systematic asynchrony of firing. DORA maintains an asynchrony of firing at either the level of POs, which we term *asynchronous binding*, or the level of RBs, which has been termed *synchronous binding*. During asynchronous binding, as a proposition in the driver becomes active, bound roles and objects fire in direct sequence (e.g., with roles firing directly before their fillers; see Fig. 2A). For example, to bind *bigger* to cup and *smaller* to ball (and so represent *bigger* (cup, ball)), the units corresponding to *larger* fire (Fig. 3A[i]) directly followed by the units corresponding to cup (Fig. 3A[ii]), followed by the units for coding *smaller* (Fig. 3A[iii]) followed by the units for ball (Fig. 3A[iv]).

In brief, bound role-filler pairs fire as couplets, and role-filler sets from the same proposition fire in sequence. For instance, to represent both *bigger* (cup, ball) and *bigger* (circle star), the units representing *larger* would fire, followed by the units representing cup, then the units representing *smaller*

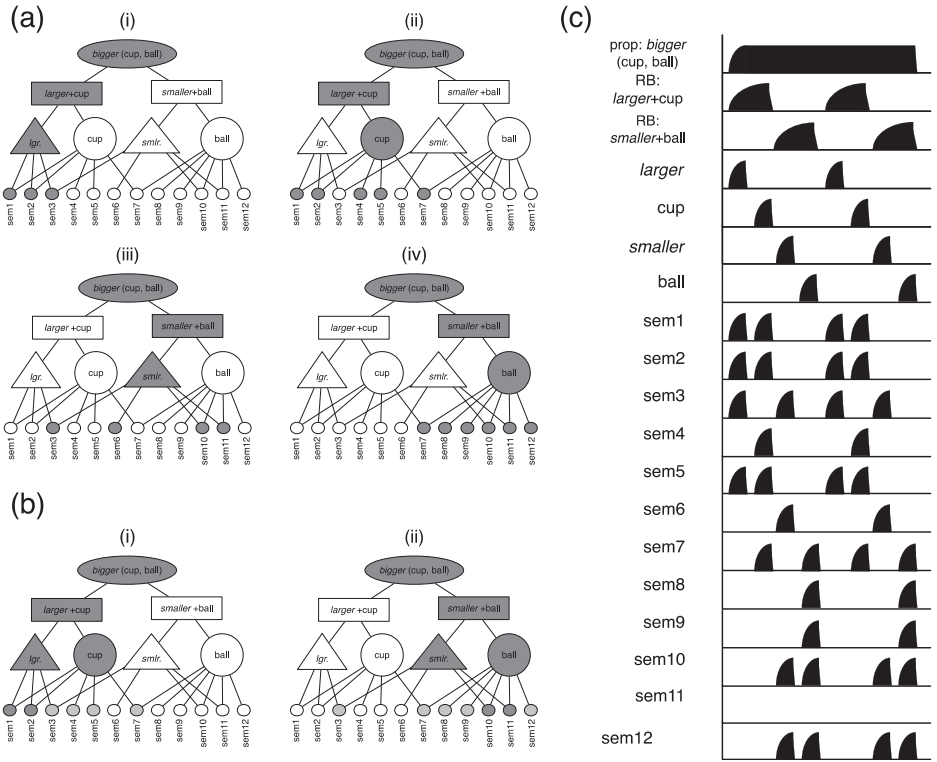

Fig. 3 Binding in DORA. Binding in DORA occurs via systematic asynchrony of firing. The asynchrony is maintained at either the level of POs (A) or RBs (B). When the asynchrony is maintained at the level of POs, to represent the binding between a role and a filler, units coding for the role fire directly in sequence with units coding for a filler, and out of synchrony with other role-filler sets. Gray units are active. (A[i–ii]) The binding of *larger* to cup is carried by the sequence of firing of the units representing *larger* A[i] followed by the units representing cup A[ii]. (A[iii–iv]) The binding of *smaller* to ball is carried by the sequence of firing of the units for *smaller* A[iii] followed by the units for ball A[iv]. When the asynchrony is maintained at the level of RBs, to represent the binding between a role and a filler, units coding for one role-filler pair fire out of synchrony with the units coding for another role-filler pair. The binding of *larger* to cup is carried by the firing of the units coding for *larger* and cup firing together (B[i]) and out of sequence with the units coding for *smaller* and ball (B[ii]). (C) A time series illustration of the firing of units during asynchrony-based binding (as in (A)).

fire, followed by the units representing ball. Next, the units representing *larger* fire, followed by the units representing circle, then the units representing *smaller* fire, followed by the units representing star (see Fig. 4). In binding by systematic asynchrony, binding information is carried by when units fire. Role-filler bindings are dynamic and represented explicitly, while role-filler independence is maintained (see, e.g., Doumas et al., 2008). Consequently, there is no need to use different types of units (as in SME; Falkenhainer et al., 1989; Forbus, Gentner, & Law, 1995; or STAR; Halford et al., 1998) or different sets of units (as in LISA; Hummel & Holyoak, 1997, 2003) to represent relations/relational roles and objects and their semantic properties. Accordingly, in DORA, roles and objects are both coded by a common pool of semantic units, and, importantly, as detailed below, this capacity allows DORA to learn representations of object properties and relational roles from representations of objects.

DORA can also maintain a systematic asynchrony at the level of RBs, such that bound roles fire in synchrony with their fillers (Fig. 3B). This temporal binding signal is the same as that used by LISA (Hummel & Holyoak, 1997, 2003). When role-filler pairs fire in synchrony, their semantic patterns are superimposed. To keep role and filler representations distinct, as is necessary for many forms of learning, including schema induction and relational generalization, and the representation of propositions or sentences (Martin & Doumas, 2017), different pools of units are required to code for the properties of objects and roles, which makes learning relational roles and multi-place relations from object representations much more difficult (see Doumas et al., 2008). However, the resources required to bind role-filler pairs are halved during synchronous binding as opposed to asynchronous binding—specifically, binding a role-filler pair by asynchrony requires that two spikes of unit activation be maintained and kept distinct, but synchrony requires only one spike of unit activation. As such, DORA uses synchronous binding for tasks that do not require that role and filler semantics be differentiated (e.g., retrieval and mapping), and asynchronous binding for tasks that require distinct representations of role and filler semantics (e.g., learning). The model thus makes the prediction that representing a proposition should require more WM (or other processing resources) during learning than during mapping. This prediction appears to hold for humans (see, e.g., Saiki, 2003).

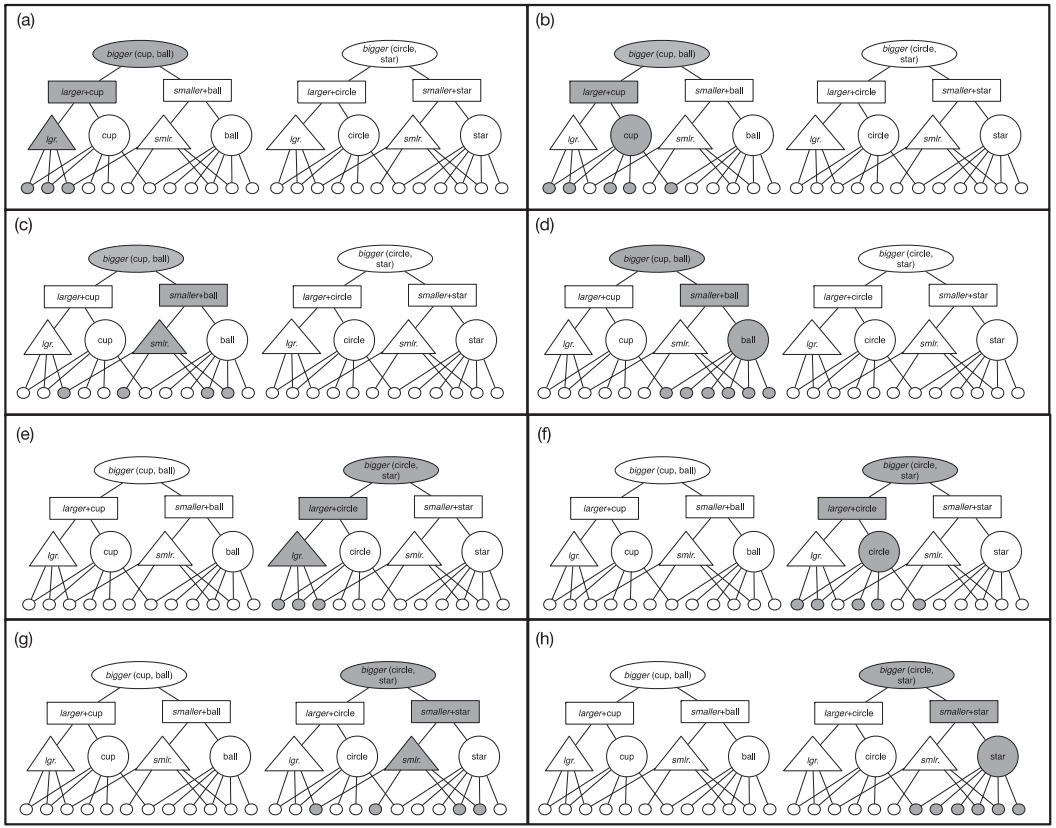

Fig. 4 Binding roles to fillers across multiple propositions in DORA. (A–D) DORA represents *bigger* (*cup, ball*), representing the binding of *larger* to cup (a–b) and *smaller* to ball (c–d). (e–h) DORA represents *bigger* (*circle, star*), representing the binding of *larger* to circle (e–f), and *smaller* to star (g–h).

3.4 Retrieval and mapping

DORA adopts its retrieval and mapping algorithms from LISA (see Doumas et al., 2008; Hummel & Holyoak, 1997, 2003). Retrieval from LTM and analogical mapping are highly related processes. During retrieval, propositions in the driver become active and time-share, due to time-based binding, as described above. Patterns of semantic activation generated by the driver propositions excite token units in LTM. Analogs are retrieved from LTM into the recipient using the Luce (1956) choice axiom:

$$L_i = \frac{R_i}{\sum_i R_j} \tag{1}$$

where L_i is the probability that analog i will be retrieved into working memory, R_i is the maximum activation of a token unit in i reached during the retrieval phase and j are all other analogs in LTM.

Analogical mapping is the processes of discovering which elements (objects, relational roles, whole propositions, etc.) of one analog correspond to which elements of another. Mapping is similar to retrieval with two important distinctions. First, activation flows from units in the driver, through the semantic units, to units in the recipient (rather than units in LTM); and second, connections (called *mapping connections*) are established between coactive units in driver and recipient. During mapping, propositions in the driver become active as described above and activate their semantic units. Units in the recipient compete via lateral inhibition to respond to the pattern of active semantic units. Units in the recipient that share the most semantic content with the active units in the driver will tend to become the most active. DORA learns mapping connections between co-active token units of the same type in the driver and recipient (e.g., between driver and recipient PO units, and between driver and recipient RB units). As a result, DORA will map items across the driver and recipient that share both relational and object similarity.

For example, consider a simple case where DORA has a representation of *taller* (Bill, Joe) in the driver, and *taller* (Susan, Jill) in the recipient. As the representation of Bill as having more height—i.e., *more-height* (Bill)—becomes active in the driver, it activates any semantic features of *more-height*. These, in turn, activate the representation of *more-height* in the recipient—in this case the *more-height* (Susan) role-binding. When the representations of *more-height* (Bill) and *more-height* (Susan) are coactive across the driver and

recipient, DORA learns mapping connections between them (i.e., it learns that *more-height* (Bill) corresponds or maps to *more-height* (Susan)).

The mapping algorithm has been used to simulate a number of phenomena from the literature on human analogy making (see Doumas et al., 2008; Doumas, Morrison, & Richland, 2018; Hummel & Holyoak, 1997; Morrison et al., 2011).

3.5 Generalization

When augmented with the capacity for self-supervised learning, LISA's mapping algorithm naturally allows for analogical inference. To illustrate, consider how DORA solves an inference problem such as a common geometric analogy problem such as the one in Fig. 5. DORA represents the A and B terms in the driver and the C term in the recipient. As the proposition coding for the A term, *above* (rectangle, square), becomes active in the driver, it activates, and consequently maps to, the units coding for *above* (circle, triangle) in the recipient. Specifically, the units coding for *higher* (rectangle) in the driver activate and map to the units coding for *higher* (circle) in the recipient, and the units coding for *lower* (square) in the driver activate and map to the units coding for *lower* (triangle) in the recipient.

However, when the B term, *above* (square, rectangle) becomes active in the driver, there are no corresponding units in the recipient for it to map to (recall the C term is already mapped to the A term). When units are active in the driver and no units are available for mapping in the recipient, DORA performs analogical inference via a self-supervised learning algorithm. During self-supervised learning, active units in the driver signal DORA to recruit matching units in the recipient. Continuing the example, as units coding for *higher* (square) become active, DORA recruits RB and P units in the recipient to match the active RB in P in the driver. Newly recruited P units in the recipient learn connections to active recipient RB units, and newly recruited RB units learn connections to active PO units (i.e., DORA

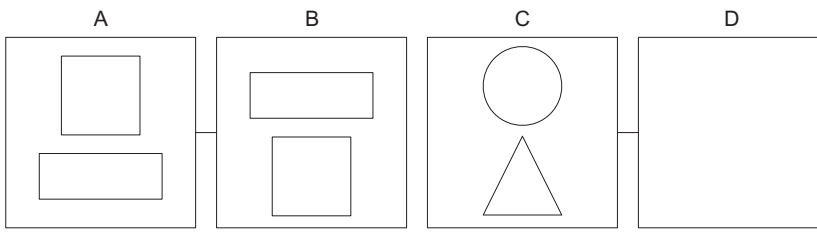

Fig. 5 An example of a geometric analogy problem.

learns connections between the new P and RB units and between the new RB unit and the units coding for *higher* and triangle in the recipient). In other words, DORA infers that *higher* (square) in the driver should correspond to *higher* (triangle) in the recipient. The same happens when *lower* (rectangle) fires in the driver and LISA infers *lower* (circle) in the recipient. Thus, DORA completes the D term in a problem via analogical inference.

3.6 Learning single-place predicates by comparison

DORA uses comparison to isolate shared properties of objects and to represent them as explicit structures. DORA starts with simple feature-vector representations of objects (i.e., a node connected to set of object features; Fig. 6A). After mapping, corresponding elements in the driver and recipient will fire together. For example, when DORA compares a cup to a ball, the nodes representing the cup and ball will map, and will fire together (Fig. 6A). Any semantic features that are shared by both compared objects (i.e., features common to both the cup and the ball) receive twice as much input and consequently become roughly twice as active as features connected to only one object (here, for example, these features might be "more" and "size"; Fig. 6B).

DORA uses this activation based highlighting to bootstrap the explicit predication of shared properties. Whenever two solitary object units are mapped, DORA recruits a PO unit in the recipient, clamps its activation to 1.0, and learns connections between that unit and active semantics via the proportional Hebbian learning equation:

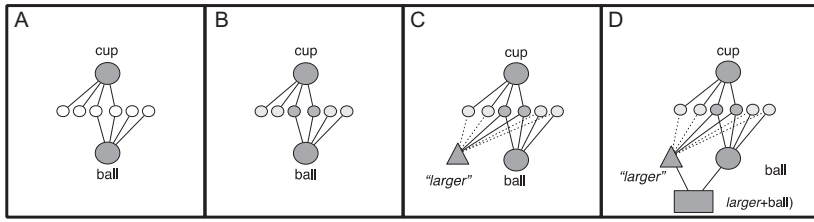

Fig. 6 Comparison-based predication in DORA. DORA learns a representation of smaller by comparing a square that is smaller some object to a triangle smaller some object. Darker gray denotes more active units; lighter gray denotes less active units. (A) DORA compares a cup and a ball. Units representing both become active. (B) Features shared by the cup and the ball become more active than unshared features (darker gray). (C) A new unit learns connections to features in proportion to their activation (solid lines indicate stronger connection weights). The new unit codes the featural overlap of the cup and ball (i.e., the role *larger*).

$$\Delta w_{ij} = a_i\left(a_j - w_{ij}\right)\gamma \tag{2}$$

where Δw_{ij} is the change in weight between the new PO unit, i, and semantic unit, j, a_i and a_j are the activations of i and j, respectively, and γ is a growth rate parameter. By Eq. (2), the weight between the recruited PO and a semantic unit will asymptote to that semantic unit's activation. As semantics shared by the compared objects are roughly twice as active as unshared semantics, and because the strength of connections learned via Hebbian learning is a function of the units' activations, DORA learns stronger connections between the recruited PO unit and the semantic units shared by the compared objects (Fig. 6C). The recruited PO is thus an explicit representation of the featural overlap of the compared objects. In this example, DORA forms an explicit representation of "*larger*" (i.e., the features common to both the cup and ball; Fig. 6D). In addition, DORA recruits a RB unit in the recipient, clamps its activation to 1.0, and learns connections between that unit and any active POs (namely, the recruited PO and the compared PO) via Hebbian learning (Fig. 6D). Importantly, because of DORA's capacity for time-based binding (see above), the new PO will act as an explicit predicate that is dynamically bindable to fillers.

3.7 Predicate refinement

The predicates DORA learns are likely to be initially "dirty" in that they will almost certainly contain extraneous features (e.g., any other features shared by the compared objects). Through repeated iterations of the same learning process, however, DORA forms progressively more refined representations. For example, if DORA learns a representation of *larger* by comparing a cup and a ball, this representation of *larger* might be conflated with the feature "round." Similarly, when DORA learns a representation of *larger* by comparing a red box and a red bag, this representation of *larger* might be conflated the feature "red." By comparing these two "dirty" representations, though, DORA can learn a refined representation of *larger*. When DORA compares the two "dirty" representations of *larger* that it has previously learned, it will map them (e.g., mapping a representation of *larger* (cup) to *larger* (box); Fig. 7A). Using self-supervised learning (Doumas et al., 2008; Hummel & Holyoak, 2003) DORA recruits units in the emerging recipient that correspond to active units in the driver (Fig. 7B). Because of mapping-based inhibition (see Doumas et al., 2008), if any unit, i, in the recipient maps to a unit, j, in the driver, then i will be inhibited all other units, $k \neq j$, in the

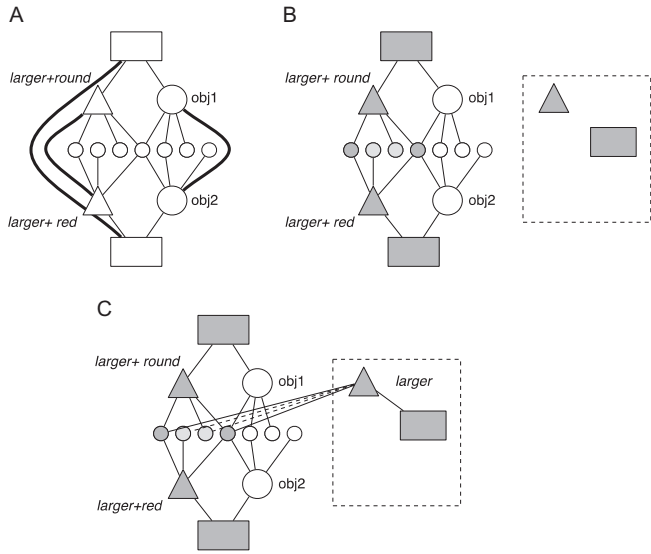

Fig. 7 Predicate refinement in DORA. Gray denotes active units. (A) DORA maps two similar predicates. Heavy solid lines denote mapping connections. Solid lines denote connections between semantic and token units. (B) DORA recruits a PO to respond to the active driver units. (C) DORA learns connections between the new PO and RB, and between the new PO and active semantics, with stronger connections learned to more active units.

driver. When units in the emerging recipient are recruited to correspond to active driver units, mapping connections are established between the corresponding units in driver and emerging recipient. As a result, when a unit, *k*, in the driver maps to no unit in the emerging recipient (or if the emerging recipient is empty), then when *k* fires, it will inhibit all emerging recipient units just as it excites no units. Any such global mapping-based inhibition is a reliable cue that nothing in the emerging recipient analog corresponds to driver unit *k*. Based on this cue, DORA recruits and activates (activation = 1.0) a unit in the emerging recipient to correspond to *k*. As in comparison-based predication, DORA learns connections between active semantics and recruited POs by Eq. (2), and between active corresponding token units (i.e., between POs and RBs, and between RBs and Ps) by simple Hebbian learning (Fig. 7C). So, when DORA compares *larger* (cup) to *larger* (box), the refined representation of the predicate *larger* will have a connection weight of 1.0 to the shared semantics of the two *larger* predicates (i.e., the semantic features

corresponding to "larger"), and connections to all extraneous semantics will be roughly halved (e.g., the semantics "round" and "red"). Applied iteratively, this process produces progressively more refined representations of the compared predicates—and eventually whole multi-place relations, as described below—ultimately resulting in "pure" representation of the property or relational role, free of extraneous semantics.

3.8 Learning multi-place relations

DORA learns representations of multi-place relations by linking sets of constituent role-filler pairs into whole relational structures. DORA exploits the temporal dynamics of binding using time to bootstrap this process. Continuing the previous example, when DORA attends to a cup that is bigger than a ball, and a bowl that is bigger than keys, it will map the representation of *larger* (cup) to *larger* (bowl) and of *smaller* (ball) to *smaller* (keys) (Fig. 8A). This process results in a distinct pattern of firing over the units composing each set of propositions: namely, the RB unit coding *larger*+cup fires out of synchrony with the RB unit coding *smaller*+ball, while the RB unit coding *larger*+bowl fires out of synchrony with the RB unit coding *smaller*+keys (Fig. 8B–D). This distinct pattern emerges in the model only under two conditions: First, when the model maps sets of role-filler pairs that have already been linked into multi-place relations (i.e., after DORA has learned multi-place relations and happens to map them; e.g., when solving an analogy problem); and second, when DORA encounters multiple structurally similar sets of co-occurring role-filler pairs (i.e., when DORA encounters and then maps similar sets of roles that are co-occurring in the world). Consequently, the pattern serves as a reliable signal that DORA exploits to bootstrap learning multi-place relations. When this diagnostic pattern emerges, DORA recruits a P unit in the recipient and clamps its activation to 1.0 (Fig. 8E). DORA then learns connections between the newly recruited P unit and RB units in the recipient as they become active, via Hebbian learning. Continuing the above example, as the RB unit coding *larger*+bowl becomes active, DORA learns a connection between that unit and the newly recruited P unit (Fig. 8F and G). Similarly, when the RB unit coding *smaller*+keys becomes active, DORA learns a connection between that unit and the newly recruited P unit (Fig. 8H and I). The result is a P unit linking the RBs in the recipient to form a whole multi-place relational structure, *bigger* (bowl, keys) (Fig. 8I).

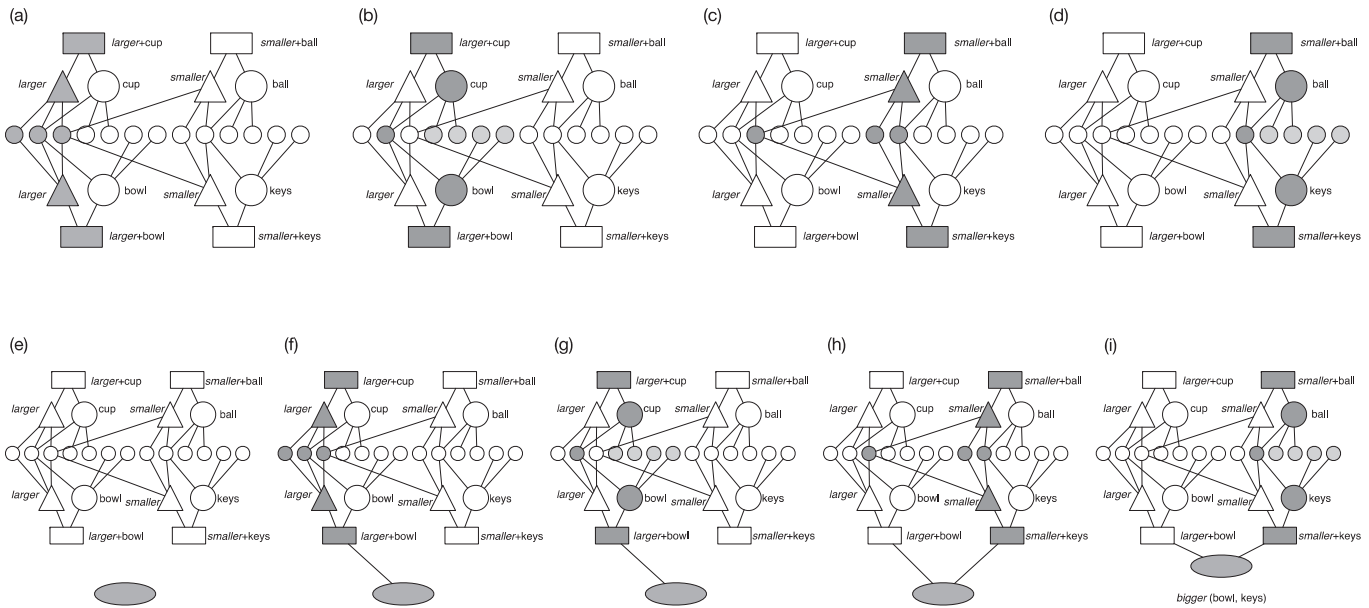

Fig. 8 DORA learns a representation of the whole relation *bigger* (bowl, keys) by mapping *larger* (cup) to *larger* (bowl) and *smaller* (ball) *smaller* (keys). Units above the semantic units (small circles) are driver units. Units below the semantic units are recipient units. Gray denotes active units. (A) The units coding *larger* in the driver and recipient fire; (B) the units for cup and bowl fire; (C) the units for *smaller* in the driver and recipient fire; (D) the units for ball and keys fire. (E) DORA recruits a P unit in the recipient. (F–G) DORA learns a connection between the recruited P unit and the active RB unit (the RB coding for *larger* + bowl). (H–I) DORA learns a connection between the recruited P unit and the active RB unit (the RB coding for *smaller* + keys). The result is a structure coding for *bigger* (bowl, keys).

3.9 DORA as an account of cognitive processing

The capacity to learn structured representations is powerful, and this power manifests in the wide range of phenomena a model that learns structured representations and reasons relationally can account for. In fact, as noted above, together DORA and LISA—which DORA grows into after it learns structured representations—account for over 50 phenomena from the literature on child and adult cognition, including human analogical reasoning, the relational shift, relational categorization, and the individual difference effects of working memory on relational learning (e.g., Doumas et al., 2008, 2018; Livins & Doumas, 2015; Livins, Doumas, & Spivey, 2015; Livins, Spivey, et al., 2015; Morrison et al., 2011; Sandhofer & Doumas, 2008; Son et al., 2010).

One of the most interesting aspects of DORA's behavior is that the trajectory of its representation learning directly mirrors the trajectory of representations learned by human children. Smith (1984) conducted an elegant experiment in which children aged 2, 3, and 4 performed a follow-the-leader task with two experimenters. Specifically, the children and each of the two experimenters had three objects in front of them. The first experimenter would select two objects from their three, and then the second experimenter would do the same, and the child's task was to select two items from her three that matched the items selected by the two experimenters. The experimenters always selected their items by one of three rules. Using the *identical match rule*, the items selected by the first and second experimenter were identical to items in front of the child. For example, the first experimenter might select two red balls, and the second experimenter would then select two identical red balls, and finally, the child had to select two red balls from her pile of three items. Using the *feature match rule*, the items selected by the first and second experimenter matched on some feature. For example, the first experimenter might select two red items (e.g., two red balls), the second experimenter would then select two different red items (e.g., two red blocks), and the child's task was to select two red items from her pile (e.g., two red toy cars). Finally, in the *relational match rule*, the items selected by the first and second experimenters matched on some relation (e.g., being the same color, or same size). For example, the first experimenter might select two red items from her pile, and the second experimenter might select two green items from her pile, and the child had to select the two blue items from her pile.

Smith found a very clear trajectory in the capacities of children at various ages to perform this task. Particularly, 2-, 3-, and 4-year-old children all successfully completed trials using the identical match rule, 3- and 4-year-old children all successfully completed trials using the feature match rule, and 4-year-old children all successfully completed trials using the relational match rule. In other words, children initially made matches based on whole-object similarity, then acquired the capacity to match items based on specific features, while ignoring others, and finally developed the capacity to match items based on shared relations. DORA follows the exact same trajectory as the children in Smith's study (Doumas et al., 2008). DORA begins with representations of objects represented as collections of features, which support making matches based on whole-object similarity. Next, DORA learns representations of single-place predicates (or object properties) that support making matches on specific properties, while ignoring others. Finally, DORA learns representations of whole relations, by concatenating co-occurring sets of single-place predicates (see above), which support making matches on based on relational concepts.

Another interesting aspect of the DORA architecture is that it learns in an unsupervised manner (i.e., without feedback). Specifically, it learns by making comparisons, and the representations that it learns depend on the comparisons that it makes. We have used DORA to learn representations of a number of relational concepts from a collection of objects, simply by randomly selecting and attempting to compare objects. Specifically, DORA has learned representations of spatial relations (e.g., *above*, *wider*, *larger*, Doumas et al., 2008), simple visual configurations (*inside*, *double*, *darker*, Doumas et al., 2008, 2018), and even representations of geons (as in Biederman, 1987; Doumas & Hummel, 2010).

Importantly, DORA also learns without feedback in the same way that human children appear to. Kotovsky and Gentner (1996) conducted an experiment in which 5–6-year-old children attempted to solve a match to sample task with rather complex relational stimuli. The children were presented with a sample triad (three objects) that exemplified a relation (e.g., symmetry—two small shapes flanking a large central shape—or monotonic increase—three squares increasing in size from left to right). They were then presented with two match items (each also composed of three shapes) and asked to match one with the sample. The match items either exemplified the same relation as the sample item, or shared features with

it. So, for example, the sample item might be three circles increasing in size from left to right, and the match items might be three triangles increasing in size from left to right (a relational match) and three circles, with the largest circle in the middle, the smallest on the left, and the medium sized one on the right (the featural match). Additionally, the relation could exist within dimension (e.g., an increase in size of shapes in both the sample and match item) or across dimensions (e.g., an increase in size of shapes in the sample items, but an increase in lightness of color in the match item). This task was very challenging for younger children, and the 5-year-olds performed at chance. However, if the stimuli were arranged such that easier relational matches appeared earlier and progressed to more complex relational matches (termed progressive-alignment by the original authors), children who made successful matches earlier performed well above chance even on harder relational matches, even without any feedback on any of the trials. We performed the same task with DORA (Doumas et al., 2008), and just like the children in the Kotovsky and Gentner study, when DORA successfully completed earlier matches (and learned from the results), it performed with similar to success as human children. When DORA failed to make relational matches earlier, it performed at chance on later trials, also like children who made non-relational matches on earlier trials.

Recently DORA has been extended to capture both formal and implementational phenomena in a related domain of abstract mental representation: language processing. Forming structured, relational, and hierarchical representations from perceptual input that is extended in time is a challenge to the brain for any processing modality, but perhaps none so acutely as for language comprehension from speech. Martin and Doumas (2017) showed that DORA could not only represent the kinds of representations needed to capture the basic formal linguistic structures of a sentence, but that it also emits oscillatory activation during processing that is highly similar to the cortical activity elicited by the linguistic stimuli from a series of neuroimaging studies carried out by Ding, Melloni, Zhang, Tian, and Poeppel (2016).

Ding et al. (2016) presented auditory strings of synthesized Mandarin Chinese syllables to native speaker participants in a magnetoencephalography (MEG) experiment. They manipulated the structural relationship between the units in the auditory string (i.e., the syllables) such that there was either no meaningful relationship between the strings of syllables, or phrases were formed from adjacent syllables, or sentences emerged from the continuous string of syllables. Using this design, they observed peaks

in the MEG-based oscillatory response on the timescale of syllabic rate (4 Hz), phrasal rate (2 Hz), and sentential rate (1 Hz), suggesting that cortical networks track, or perhaps even generate, multiple levels of linguistic structures during speech and language comprehension.

To simulate Ding et al.'s experimental procedure we allowed DORA to process Ding et al.'s English sentences one at a time. Representations of the sentence structures entered the driver (i.e., were attended to). DORA processed the sentences as it normally would (i.e., the units fired to represent and encode binding information; see above). We tracked firing rate of all the nodes in the driver as DORA processed the sentences. Because of the controlled length and structure of the sentences, DORA, like the participants in the Ding et al. experiments, took the same amount of time to process each sentence. Strikingly, DORA closely mirrored the patterns observed by Ding et al. (2016). Specifically, just like the cortical signals, DORA showed an activation burst that lasted throughout the processing of the sentence (i.e., firing in the 1 Hz range), activation bursts at twice the rate of the whole sentence burst (i.e., firing in the 2 Hz range), aligned with phrase-level processing, and activation bursts at four times the rate of the whole sentence burst (i.e., firing in the 4 Hz range), corresponding to the word-level processing.

It is also important to note that the DORA model can learn all of the representations used in Martin and Doumas (2017) from experience; DORA can learn explicit symbolic representations of verb structures like *give*, *rubs*, or *chases*, and of single-place modifiers like *dry*, *new*, or *golden* from experience with objects in the world involved in those relations or with those features (e.g., Doumas & Hummel, 2010; Doumas et al., 2008; Lim, Doumas, & Sinnett, 2014; Sandhofer & Doumas, 2008).

Martin and Doumas (2017) provided a mechanism for how the brain might parse discrete, hierarchical, and compositional structured representations from continuous unstructured input, as in language comprehension from speech. The simulations highlighted how a key principle from DORA, i.e., exploiting the fact that time carries information, or using asynchrony of population or neuronal assembly firing, can be used to represent information at multiple timescales such that hierarchical and compositional structures emerge from naturally occurring signals in distributed computing systems. Indeed, systematic temporal asynchrony might be the computational mechanism that gives rise to the generative and compositional representational hierarchies that underlie human language processing.

4. Discussion

How a system represents information tightly constrains the kinds of problems it can solve. Humans routinely solve problems that appear to require structured representations of stimulus properties and relations. As such, answering the question of how we acquire these representations has central importance in an account of human cognition.

DORA uses Hebbian learning, time-based binding, comparison-based intersection discovery, and analogical mapping to learn and refine effective single-place predicate representations of object properties and relational roles, and to form whole multi-place relational structures from sets of co-occurring single-place predicates, and then further refine those relational representations. The result is a system that learns structured representations of relations from unstructured flat feature vector representations (i.e., traditional connectionist distributed representations) of objects with absolute properties.

A question that might arise is whether the role-filler representational system that DORA employs is sufficient for supporting all the relations that humans learn. Again, the short answer is, at least in principle, yes. Formally, any multi-place relation is representable as a linked set of single-place predicates (Mints, 2001). Therefore, a role-filler system can be used to represent the relations that humans represent. The more pressing question, however, is whether a role-filler system fits with what we know about actual human relational representations. Again, the answer seems to be that it does. Certainly, models that employ role-filler type representations have successfully accounted for a large number of phenomena from human analogy making, relational learning, cognitive development, and learning (e.g., Doumas & Hummel, 2010; Doumas et al., 2008, 2018; Hummel & Holyoak, 1997, 2003; Lim, Doumas, & Sinnett, 2012, 2014; Livins, Spivey, et al., 2015; Martin & Doumas, 2017; Morrison et al., 2011, 2004; Sandhofer & Doumas, 2008; Son et al., 2010). Moreover, access to and use of role-based semantic information is quite automatic in human cognition, including during memory retrieval (e.g., Gentner, Rattermann, & Forbus, 1993; Ross, 1989), and analogical mapping and inference (Bassok & Olseth, 1995; Krawczyk et al., 2008; Kubose, Holyoak, & Hummel, 2002; Ross, 1989). Indeed, the meanings of relational roles influence relational thinking even when they are irrelevant or misleading (e.g., Bassok & Olseth, 1995; Ross, 1989). Role information appears to be an integral part of the mental

representation of relations, and role-filler representations provide a direct account for why it is so.

Additionally, Livins, Doumas, et al. (2015) and Livins, Spivey, et al. (2015) have shown that we can affect the direction of the relation by manipulating which item you look at first. Livins et al. showed participants images depicting a relation that could be interpreted in different forms (e.g., *chase/pursued-by*, *lift/hang*). Before the image appeared on screen, though, a dot appeared on the screen drawing the participant's attention to a location that one of the objects involved in the relation would appear. For example, the image might show a monkey hanging from a man's arm, and the participant might be cued to the location where the monkey would appear. The relation that the participant used to describe the image was strongly influenced by the object that they attended to first. That is, if the participant saw the image of the monkey hanging from the man's arm, and she was cued to the monkey, they would describe the scene using a *hanging* relation. However, if the participant was cued to the man, she would describe the scene using a *lifting* relation. This result follows directly from a system based on role-filler representations wherein complementary relations are represented by a similar set of roles, but the predicate, or role, that fires first defines the subject of the relation.

4.1 Learning things we don't already know

As noted above, Bayesian models of concept learning generally follow a learning-by-hypothesis-testing framework (e.g., Goodman et al., 2011; Kemp & Tenenbaum, 2009; Lake et al., 2015; cf. Lu et al., 2012). In these models, the system starts with a large set of representations and rules for combining them, and then learns combinations of these elements that best fit a given set of data. In other words, the model might learn a particular configuration of symbols to solve a problem, but this representation can be generated in the model before any actual learning occurs. In fact, a configuration must be generated in order to enter as a candidate hypothesis in the learning algorithm.

The DORA model starts with representations of objects represented as flat feature vectors. Before learning, there are no functional predicate representations anywhere in the model. That is, at time $t(0)$, the model has one type of representation, and not another. During learning, DORA learns new representations. Some of these new representations function like single-place—and eventually multi-place—predicates and can be bound

to arguments. That is, at time $t(n)$, after learning, the model has another type of representation, one that is of a qualitatively different type from any representation present anywhere in the model at time $t(0)$. The result is that the expressive power of the system has increased between $t(0)$ and $t(n)$. Of course, trivially, the capacity to *learn* these representations is present in the model (as it must be). There are architectural assumptions made (detailed in the main text and in Doumas et al., 2008), but these assumptions simply produce the capacity to learn predicate type representations. The representations themselves are specified nowhere within the model, and as a result, after learning, the model has a capacity to *represent* things that it simply did not have before learning. It is only after learning—including learning both representations that function like single-place predicates and multi-place relations—that the structures necessary to represent relational propositions are present in the model. This change stands in stark contrast to models like those of Lake et al. (2015) and Kemp and colleagues (Kemp, 2012; Kemp & Tenenbaum, 2009), wherein the models start with explicit representations of all the primitive elements and data types necessary to express what is present in the training dataset or environment.

It is certainly possible that models like those proposed by Lake et al. (2015) and Kemp and colleagues (Kemp, 2012; Kemp & Tenenbaum, 2009) might be augmented with routines to generate the representations that they assume at the onset of learning. Perhaps solutions like DORA and BART might provide the means by which such useful representations can be generated on the first place. At the very least, Lake et al. and Kemp and Tenenbaum's models might serve as very useful tools for addressing questions about what humans do with the representations that we have learned after they have acquired them.

Fodor (1975) provides a well-known argument for radical nativism, which has come to be known as "Fodor's puzzle." The argument—in its simplest form—progresses as follows: (1) concept learning is a process of hypothesis formulation and testing; (2) formulating a hypothesis about some x requires having a concept of that x; (3) therefore, concepts are represented before they are learned, i.e., they are innate. In short, the argument holds that because the expressive power of the system at some time t determines the hypotheses that can be expressed and tested, the expressive power of a cognitive system is, primarily, fixed at onset. A cognitive system can learn to express combinations of existing concepts and can learn to weight different combinations as more or less useful, but the constituent elements of these

concepts and the rules for combining them are present in the system a priori. This argument has had a great deal of influence in the cognitive science community.

There have been some attempts to address Fodor's problem. Generally, these accounts take umbrage with the statement that concept learning must be a process of hypothesis formation and testing. Specifically, the argument goes that we can test the utility of concepts via hypothesis testing, but we learn new concepts, by some other means. Such approaches usually propose some process for learning new primitives. While a system might begin with some set of initial representational primitives and means of combining them, by learning new primitives the system can formulate new concepts, which, in turn, can be tested for their utility. There are many such approaches in the literature including tuning (Landy & Goldstone, 2005), Quineian bootstrapping (Carey, 2009), or random generation of representational elements.

An alternative idea is to extend the expressive power of a system by learning new data structures or data types. For example, if a system that has representations of objects, but no predicate representations, when that system learns a predicate representation—even if that predicate is composed of primitives that already exist in the system—then the expressive power of that system has necessarily increased. Now the system can represent novel statements as a function of being able to represent instances wherein a predicate is *about* objects. The current proposal falls into this later camp.

Finally, it is worth noting that DORA achieves much of what it does by exploiting naturally occurring "neural" oscillations as it learns and processes. Being sensitive to how information is carried in time in a neural system means that the system can also learn from its dynamics. This capacity is noteworthy not only for its computational power (e.g., being able to learn from past states and learn relations over multiple timepoints and states) but also for the explanatory parsimony this mechanism may offer in linking theories of neural computation to formal accounts of cognition and to the wealth of emerging neuroscientific data about the function of neural oscillations in cognition. We believe computing with neural oscillations represents a fundamental formal and neurophysiological alignment between how human-like representations can be achieved in a system that learns, and how distributed neural computing systems, including cortical assemblies, might process information (see Martin & Doumas, 2017 for a discussion). Neural oscillations have long been implicated as the indices of neural

information processing (e.g., Buzsáki, 2006). Learning symbolic structure from signals that naturally occur in distributed computing systems offers a promising approach whereby the computational principles that can yield the highest forms of the human mind (e.g., relational reasoning, formal and natural language processing) can also be realized in systems based on the computational primitives of neurophysiology.

4.2 Limitations and future directions

Humans routinely learn structured representations from experience. We offer an account of this fundamental process that is based on minimal standard assumptions of connectionist systems. Our account is, of course, limited in several ways. Below we outline some of the limitations of the current model and propose some means of possibly addressing these limitations in future work.

First, the constraints on learning in our system are likely underdetermined; DORA learns when it can and stores all the results of its learning. We have implemented a crude form of recency bias in our simulations (see above), but future work should focus on development of more principled mechanisms for constraining learning and storage. Such mechanisms might focus on either constraining when learning takes place, or on when the results of learning are stored for future processing. Most likely, though, it will be necessary to account for both.

Constraining when DORA learns amounts, essentially, to constraining when it performs comparison. We have previously proposed a number of possible constraints on comparison such as language (e.g., shared labels) and object salience, and have shown how direction to compare (i.e., instruction) serves as a very powerful constraint on learning (see Doumas & Hummel, 2013; Doumas et al., 2008). These constraints may also serve to limit when the results of learning are stored in memory. DORA might be extended or integrated with existing accounts of language or perceptual (feature) processing in order to implement such constraints (see, e.g., Martin, 2016; Martin & Doumas, 2017).

Perhaps more satisfyingly, both of these limitations might be successfully addressed by refining the control structure of the system. We see evaluating the quality of comparisons and of the representations that DORA learns as important potential constraints that the control process might impose. Reinforcement learning provides a very useful tool for

implementing both of these constraints. Our current work is focused on developing a reinforcement-based control structure in DORA. This structure has two primary focuses: (a) it evaluates the utility of a current comparison based on the reward from learning based on similar comparisons in the past. (b) It scores the utility of propositions in LTM based on their retrieval history and the reward from inferences based on these propositions in the past. This utility metric might then be used to prune representations in LTM.

Second, we lack a full account of how known predicates and relations are recognized during real-time processing. Previous work has shown that directing attention to particular features or particular agents has a pronounced effect on what relations are recognized in a scene (Livins & Doumas, 2015; Livins, Doumas, et al., 2015; Livins, Spivey, et al., 2015). For instance, directing attention to the height dimension (e.g., by having the participant move her head up and down) will drastically increase the chance that the participant will recognize a relation on that dimension (e.g., *above* (*x*,*y*); Livins, Doumas, et al., 2015). Furthermore, drawing a participant's attention to a particular object, unsurprisingly, makes recognizing a relation involving that object more likely, but also increases the probability of recognizing a relation in which that object is the subject (Livins & Doumas, 2015; Livins, Doumas, et al., 2015). It remains an open question, however, how relational recognition is actually implemented in human cognition—although, see Livins, Doumas, et al. (2015) and Livins, Spivey, et al. (2015) for a potential candidate.

The discovery of invariance has relevance beyond the few problems presented here. For example, detecting invariants in speech and language is a defining and unsolved problem in language acquisition and adult speech processing, including in automatic speech recognition by machines. Similarly, whether the generalization of grammatical rules can be fully accounted for in systems that rely on statistical learning alone remains highly contentious. The account of learning invariance from experience offered here, combined with principles like the compression of role information (Doumas, 2005), may present new computational vistas on these classic problems in the language sciences (see Martin, 2016; Martin & Doumas, 2017 for further discussion). Systems with the properties of DORA, augmented by this subroutine and likely others, may offer an inroad to representational sufficiency across multiple domains, built from the same mechanisms and computational primitives.

References

Anderson, J. R. (2009). *How can the human mind occur in the physical universe?* Oxford University Press.

Bassok, M., & Olseth, K. L. (1995). Judging a book by its cover: Interpretative effects of content on problem-solving transfer. *Memory & Cognition*, *23*(3), 354–367.

Bowers, J. S. (2017). Parallel distributed processing theory in the age of deep networks. *Trends in Cognitive Sciences*, *21*, 950–961.

Biederman, I. (1987). Recognition-by-components: A theory of human image understanding. *Psychological Review*, *94*(2), 115.

Bringsjord. (2008). Declarative/logic-based cognitive models. In R. Sun (Ed.), *Cambridge handbook of computational psychology* (pp. 127–169). Cambridge University Press.

Buzsáki, G. (2006). *Rhythms of the brain*. Oxford University Press.

Carey, S. (2009). *The origin of concepts*. Oxford University Press.

Christiansen, M. H., & Chater, N. (2001). Connectionist psycholinguistics: Capturing the empirical data. *Trends in Cognitive Sciences*, *5*(2), 82–88.

Ding, N., Melloni, L., Zhang, H., Tian, X., & Poeppel, D. (2016). Cortical tracking of hierarchical linguistic structures in connected speech. *Nature Neuroscience*, *19*(1), 158.

Doumas, L. A. A. (2005). A neural-network model for discovering relational concepts and learning structured representations (Doctoral dissertation). *UCLA*.

Doumas, L. A., Hamer, A., Puebla, G., & Martin, A. E. (2017). A theory of the detection and learning of structured representations of similarity and relative magnitude. In *The 39th annual conference of the Cognitive Science Society (CogSci 2017)* (pp. 1955–1960). Cognitive Science Society.

Doumas, L. A., & Hummel, J. E. (2005). Approaches to modeling human mental representations: What works, what doesn't and why. In K. J. Holyoak & R. G. Morrison (Eds.), *The Cambridge handbook of thinking and reasoning* (pp. 73–94). Cambridge University Press.

Doumas, L. A., & Hummel, J. E. (2010). A computational account of the development of the generalization of shape information. *Cognitive Science*, *34*(4), 698–712.

Doumas, L. A., & Hummel, J. E. (2012). Computational models of higher cognition. In *Vol. 19. The Oxford handbook of thinking and reasoning*. New York, NY: Oxford University Press.

Doumas, L. A., & Hummel, J. E. (2013). Comparison and mapping facilitate relation discovery and predication. *PLoS One*, *8*(6), e63889.

Doumas, L. A., Hummel, J. E., & Sandhofer, C. M. (2008). A theory of the discovery and predication of relational concepts. *Psychological Review*, *115*(1), 1–43.

Doumas, L., Morrison, R. G., & Richland, L. E. (2018). Individual differences in relational learning and analogical reasoning: A computational model of longitudinal change. *Frontiers in Psychology*, *9*, 1235. https://doi.org/10.3389/fpsyg.2018.01235.

Falkenhainer, B., Forbus, K. D., & Gentner, D. (1989). The structure-mapping engine: Algorithm and examples. *Artificial Intelligence*, *41*(1), 1–63.

Fodor, J. A. (1975). The language of thought. (*Vol. 5*). Harvard University Press.

Fodor, J. A., & Pylyshyn, Z. W. (1988). Connectionism and cognitive architecture: A critical analysis. *Cognition*, *28*(1), 3–71.

Forbus, K. D., Gentner, D., & Law, K. (1995). MAC/FAC: A model of similarity-based retrieval. *Cognitive Science*, *19*(2), 141–205.

Franklin, S. (1999). *Artificial minds*. MIT Press.

Gentner, D. (2010). Bootstrapping the mind: Analogical processes and symbol systems. *Cognitive Science*, *34*(5), 752–775.

Gentner, D., & Forbus, K. D. (2011). Computational models of analogy. *Wiley Interdisciplinary Reviews: Cognitive Science*, *2*(3), 266–276.

Gentner, D., Rattermann, M. J., & Forbus, K. D. (1993). The roles of similarity in transfer: Separating retrievability from inferential soundness. *Cognitive Psychology*, *25*, 524–575.

Goodman, N. D., Ullman, T. D., & Tenenbaum, J. B. (2011). Learning a theory of causality. *Psychological Review*, *118*(1), 110.

Halford, G. S., Wilson, W. H., & Phillips, S. (1998). Processing capacity defined by relational complexity: Implications for comparative, developmental, and cognitive psychology. *Behavioral and Brain Sciences*, *21*(6), 803–831.

Halford, G. S., Wilson, W. H., & Phillips, S. (2010). Relational knowledge: The foundation of higher cognition. *Trends in Cognitive Sciences*, *14*(11), 497–505.

Holyoak, K. J. (2012). Analogy and relational reasoning. In K. J. Holyoak & R. G. Morrison (Eds.), *The Oxford handbook of thinking and reasoning* (pp. 234–259). New York: Oxford University Press.

Holyoak, K. J., & Hummel, J. E. (2000). The proper treatment of symbols in a connectionist architecture. In *Cognitive dynamics: Conceptual change in humans and machines* (pp. 229–263). Lawrence Erlbaum Associates.

Holyoak, K. J., & Thagard, P. (1995). *Mental leaps*. MIT Press.

Hummel, J. E. (2010). Symbolic versus associative learning. *Cognitive Science*, *34*(6), 958–965.

Hummel, J. E. (2013). Object recognition. In D. Reisberg (Ed.), *Oxford handbook of cognitive psychology* (pp. 32–46). Oxford, UK: Oxford University Press.

Hummel, J. E. (2017). Putting distributed representations into context. *Language, Cognition and Neuroscience*, *32*(3), 359–365.

Hummel, J. E., & Holyoak, K. J. (1997). Distributed representations of structure: A theory of analogical access and mapping. *Psychological Review*, *104*(3), 427–466.

Hummel, J. E., & Holyoak, K. J. (2003). A symbolic-connectionist theory of relational inference and generalization. *Psychological Review*, *110*(2), 220–264.

Jordan, M. I. (1986). An introduction to linear algebra in parallel distributed processing. *Parallel Distributed Processing*, *1*, 365–422.

Kellman, P. J., Burke, T., & Hummel, J. E. (1999). Modeling perceptual learning of abstract invariants. In *Proceedings of the Twenty First Annual Conference of the Cognitive Science Society* (pp. 264–269).

Kemp, C. (2012). Exploring the conceptual universe. *Psychological Review*, *119*(4), 685–722.

Kemp, C., & Tenenbaum, J. B. (2009). Structured statistical models of inductive reasoning. *Psychological Review*, *116*(1), 20.

Kotovsky, L., & Gentner, D. (1996). Comparison and categorization in the development of relational similarity. *Child Development*, *67*(6), 2797–2822.

Krawczyk, D. C., Morrison, R. G., Viskontas, I., Holyoak, K. J., Chow, T. W., Mendez, M. F., et al. (2008). Distraction during relational reasoning: The role of prefrontal cortex in interference control. *Neuropsychologia*, *46*(7), 2020–2032.

Kriete, T., Noelle, D. C., Cohen, J. D., & O'Reilly, R. C. (2013). Indirection and symbol-like processing in the prefrontal cortex and basal ganglia. *Proceedings of the National Academy of Sciences of the United States of America*, *110*, 16390–16395.

Kubose, T. T., Holyoak, K. J., & Hummel, J. E. (2002). The role of textual coherence in incremental analogical mapping. *Journal of Memory and Language*, *47*(3), 407–435.

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

Lake, B. M., Ullman, T. D., Tenenbaum, J. B., & Gershman, S. J. (2017). Building machines that learn and think like people. *Behavioral and Brain Sciences*, *40*, e253.

Landy, D., & Goldstone, R. L. (2005). How we learn about things we don't already understand. *Journal of Experimental & Theoretical Artificial Intelligence*, *17*(4), 343–369.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444.

Leech, R., Mareschal, D., & Cooper, R. P. (2008). Analogy as relational priming: A developmental and computational perspective on the origins of a complex cognitive skill. *Behavioral and Brain Sciences*, *31*(4), 357–378.

Lim, A., Doumas, L., & Sinnett, S. (2012). Modeling melodic perception as relational learning using a symbolic-connectionist architecture (DORA). In *Vol. 34. Proceedings of the Cognitive Science Society*. No. 34.

Lim, A., Doumas, L., & Sinnett, S. (2014). Supramodal representations in melodic perception. In *Vol. 36. Proceedings of the Cognitive Science Society*. No. 36.

Livins, K. A., & Doumas, L. A. (2015). Recognising relations: What can be learned from considering complexity. *Thinking & Reasoning*, *21*(3), 251–264.

Livins, K., Doumas, L. A., & Spivey, M. J. (2015). Tracking relations: The effects of visual attention on relational recognition. In *Proceedings of the Cognitive Science Society*.

Livins, K. A., Spivey, M. J., & Doumas, L. A. (2015). Varying variation: The effects of within-versus across-feature differences on relational category learning. *Frontiers in Psychology, 6.* https://doi.org/10.3389/fpsyg.2015.00129.

Lu, H., Chen, D., & Holyoak, K. J. (2012). Bayesian analogy with relational transformations. *Psychological Review*, *119*(3), 617.

Luce, R. D. (1956). Semiorders and a theory of utility discrimination. *Econometrica, Journal of the Econometric Society*, *24*, 178–191.

Marcus, G. F. (1998). Rethinking eliminative connectionism. *Cognitive Psychology*, *37*(3), 243–282.

Marcus, G. F. (2003). *The algebraic mind: Integrating connectionism and cognitive science*. Cambridge, MA: MIT press.

Markman, A. B. (1999). *Knowledge representation*. Mahwah, NJ: Erlbaum.

Martin, A. E. (2016). Language processing as cue integration: Grounding the psychology of language in perception and neurophysiology. *Frontiers in Psychology,* 7. https://doi.org/10.3389/fpsyg.2016.00120.

Martin, A. E., & Doumas, L. A. (2017). A mechanism for the cortical computation of hierarchical linguistic structure. *PLoS Biology*, *15*(3), e2000663.

McClelland, J. L., & Cleeremans, A. (2009). Connectionist models. In T. Byrne, A. Cleeremans, & P. Wilken (Eds.), *Oxford companion to consciousness* (pp. 177–181). New York: Oxford University Press.

Medin, D. L., Goldstone, R. L., & Gentner, D. (1993). Respects for similarity. *Psychological Review*, *100*(2), 254.

Mints, G. E. (2001). Arithmetic, formal. In M. Hazewinkel (Ed.), *Encyclopaedia of mathematics* (pp. 63–64). Berlin, Germany: Springer.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., et al. (2015). Human-level control through deep reinforcement learning. *Nature*, *518*(7540), 529.

Morrison, R. G., Doumas, L. A., & Richland, L. E. (2011). A computational account of children's analogical reasoning: Balancing inhibitory control in working memory and relational representation. *Developmental Science*, *14*(3), 516–529.

Morrison, R. G., Krawczyk, D. C., Holyoak, K. J., Hummel, J. E., Chow, T. W., Miller, B. L., et al. (2004). A neurocomputational model of analogical reasoning and its breakdown in frontotemporal lobar degeneration. *Journal of Cognitive Neuroscience*, *16*(2), 260–271.

O'Reilly, R. C., & Busby, R. S. (2002). Generalizable relational binding from coarse-coded distributed representations. In T. G. Dietterich, S. Becker, & Z. Ghahramani (Eds.), *Vol. 14. Advances in neural information processing systems (NIPS)* (pp. 75–82). Cambridge, MA: MIT Press.

O'Reilly, R. C., Busby, R. S., & Soto, R. (2003). Three forms of binding and their neural substrates: Alternatives to temporal synchrony. In A. Cleeremans (Ed.), *The unity of*

consciousness: Binding, integration, and dissociation (pp. 168–192). Oxford, England: Oxford University Press.
Peirce, C. S. (1879/1903). Logic as semiotic: The theory of signs. In J. Buchler (Ed.), *The philosophical writings of Peirce (1955)* (pp. 98–119). New York: Dover Books.
Penn, D. C., Holyoak, K. J., & Povinelli, D. J. (2008). Darwin's mistake: Explaining the discontinuity between human and nonhuman minds. *Behavioral and Brain Sciences*, *31*(2), 109–130.
Pinker, S., & Prince, A. (1988). On language and connectionism: Analysis of a parallel distributed processing model of language acquisition. *Cognition*, *28*(1), 73–193.
Rogers, T. T., & McClelland, J. L. (2008). Precis of semantic cognition, a parallel distributed processing approach. *Behavioral and Brain Sciences*, *31*, 689–749.
Ross, B. H. (1989). Distinguishing types of superficial similarities: Different effects on the access and use of earlier problems. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *15*(3), 456.
Rumelhart, D. E., McClelland, J. L., & The PDP Research Group, (Eds.), (1986). *In Vol. 1. Parallel distributed processing: Explorations in the microstructure of cognition*. Cambridge, MA: MIT Press.
Saiki, J. (2003). Feature binding in object-file representations of multiple moving objects. *Journal of Vision*, *3*, 6–21.
Sandhofer, C. M., & Doumas, L. A. (2008). Order of presentation effects in learning color categories. *Journal of Cognition and Development*, *9*(2), 194–221.
Smith, L. B. (1984). Young children's understanding of attributes and dimensions: A comparison of conceptual and linguistic measures. *Child Development*, *55*(2), 363–380.
Son, J. Y., Doumas, L. A. A., & Goldstone, R. L. (2010). When do words promote analogical transfer? *The Journal of Problem Solving*, *3*(1), 4.
Spellman, B. A., & Holyoak, K. J. (1992). If Saddam is Hitler then who is George Bush?: Analogical mapping between systems of social roles. *Journal of Personality and Social Psychology*, *62*, 913–933.
Spellman, B. A., & Holyoak, K. J. (1996). Pragmatics in analogical mapping. *Cognitive Psychology*, *31*, 307–346.
Taatgen, N. A., & Anderson, J. R. (2008). ACT-R. In R. Sun (Ed.), *Constraints in cognitive architectures* (pp. 170–185). Cambridge University Press.
Usher, M., & McClelland, J. L. (2001). The time course of perceptual choice: The leaky, competing accumulator model. *Psychological Review*, *108*(3), 550–592.
von der Malsburg, C. (1999). The what and why of binding: The modeler's perspective. *Neuron*, *24*(1), 95–104.
von der Malsburg, C. (1986). Am I thinking assemblies? In *Brain theory* (pp. 161–176). Berlin, Heidelberg: Springer.

Further reading

Usher, M., & Niebur, E. (1996). Modeling the temporal dynamics of IT neurons in visual search: A mechanism for top-down selective attention. *Journal of Cognitive Neuroscience*, *8*(4), 311–327.