Theory Acquisition and the Language of Thought

Charles Kemp
ckemp@cmu.edu
Department of Psychology
Carnegie Mellon University

Noah D. Goodman & Joshua B. Tenenbaum
{ndg,jbt}@mit.edu
Department of Brain and Cognitive Sciences
Massachusetts Institute of Technology

Abstract

Everyday knowledge about living things, physical objects and the beliefs and desires of other people appears to be organized into sophisticated systems that are often called intuitive theories. Two long term goals for psychological research are to understand how these theories are mentally represented and how they are acquired. We argue that the language of thought hypothesis can help to address both questions. First, compositional languages can capture the content of intuitive theories. Second, any compositional language will generate an account of theory learning which predicts that theories with short descriptions tend to be preferred. We describe a computational framework that captures both ideas, and compare its predictions to behavioral data from a simple theory learning task.

Any comprehensive account of human knowledge must acknowledge two principles. First, everyday knowledge is more than a list of isolated facts, and much of it appears to be organized into richly structured systems that are sometimes called intuitive theories. Even young children, for instance, have systematic beliefs about domains including folk physics, folk biology, and folk psychology [10]. Second, some aspects of these theories appear to be learned. Developmental psychologists have explored how intuitive theories emerge over the first decade of life, and at least some of these changes appear to result from learning.

Although theory learning raises some challenging problems, two computational principles that may support this ability have been known for many years. First, a theory-learning system must be able to represent the content of any theory that it acquires. A learner that cannot represent a given system of concepts is clearly unable to learn this system from data. Second, there will always be many systems of concepts that are compatible with any given data set, and a learner must rely on some *a priori* ordering of the set of possible theories to decide which candidate is best [5, 9]. Loosely speaking, this ordering can be identified with a simplicity measure, or a prior distribution over the space of possible theories.

There is at least one natural way to connect these two computational principles. Suppose that intuitive theories are represented in a "language of thought:" a language that allows complex concepts to be represented as combinations of simpler concepts [5]. A compositional language provides a straightforward way to construct sophisticated theories, but also provides a natural ordering over the resulting space of theories: the *a priori* probability of a theory can be identified with its length in this representation language [3, 7]. Combining this prior distribution with an engine for Bayesian inference leads immediately to a computational account of theory learning. There may be other ways to explain how people represent and acquire complex systems of knowledge, but it is striking that the "language of thought" hypothesis can address both questions.

This paper describes a computational framework that helps to explain how theories are acquired, and that can be used to evaluate different proposals about the language of thought. Our approach builds on previous discussions of concept learning that have explored the link between compositional representations and inductive inference. Two recent approaches propose that concepts are represented in a form of propositional logic, and that the *a priori* plausibility of an inductive hypothesis is related to the length of its representation in this language [4, 6]. Our approach is similar in spirit, but is motivated in part by the need for languages richer than propositional logic. The framework we present is extremely general, and is compatible with virtually any representation language, including various forms of predicate logic. Methods for learning theories expressed in predicate logic have previously been explored in the field of Inductive Logic Programming, and we recently proposed a theory-learning model that is inspired by this tradition [7]. Our current approach is motivated by similar goals, but is better able to account for the discovery of abstract theoretical laws.

The next section describes our computational framework and introduces the specific logical language that we will consider throughout. Our framework allows relatively sophisticated theories to be represented and learned, but we evaluate it here by applying it to a simple learning problem and comparing its predictions with human inductive inferences.

A Bayesian approach to theory discovery

Suppose that a learner observes some of the relationships that hold among a fixed, finite set of entities, and wishes to discover a theory that accounts for these data. Suppose, for instance, that the entities are thirteen adults from a remote tribe (a through m), and that the data specify that the spouse relation ($\mathtt{S}(\cdot,\cdot)$) is true of some pairs (Figure 1). One candidate theory states that $\mathtt{S}(\cdot,\cdot)$ is a symmetric relation, that some of the individuals are male ($\mathtt{M}(\cdot)$), that marriages are permitted only between males and non-males, and that males may take multiple spouses but non-males may have only one spouse (Figure 1b). Other theories are possible, including the theory which states only that $\mathtt{S}(\cdot,\cdot)$ is symmetric.

Accounts of theory learning should distinguish between at least three kinds of entities: theories, models, and data. A *theory* is a set of statements that captures constraints on possible configurations of the world. For instance, the theory in Figure 1b rules out configurations where the spouse relation is asymmetric. A *model* of a theory specifies the extension

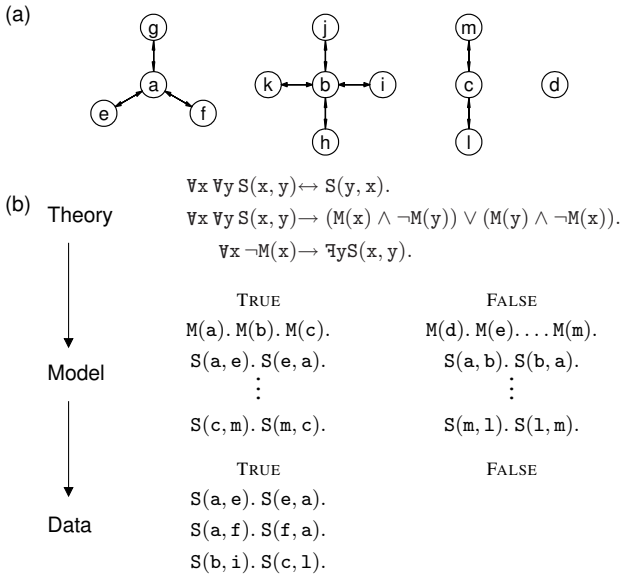

Figure 1: (a) A graph representing marital ties among a group of 13 people. (b) A hierarchical Bayesian framework for theory discovery. The theory is a collection of statements in some logical language, the model specifies the extensions of all predicates mentioned in the theory, and the data represent the information observed by the learner.

of each predicate in a way that is consistent with the theory. Note that *model* is a technical term that we have inherited from standard treatments of formal logic [2]. Figure 1b includes one model of the theory described already. The theory includes one unary predicate $\mathtt{M}(\cdot)$ and one binary predicate $\mathtt{S}(\cdot,\cdot)$, and the model specifies whether or not $\mathtt{M}(\cdot)$ is true for each individual, and whether or not $\mathtt{S}(\cdot,\cdot)$ is true for each pair of individuals. The actual state of the world is captured by just one model, but typically there will be many models consistent with a given theory. The *data* in Figure 1b represent the information available to the theory-learner. Often a learner will have partial information about the state of the world, and the data will capture only some of the information specified by the underlying model.

A hierarchical Bayesian approach allows us to transform the diagram in Figure 1b into a formal account of theory learning. The theory T that best accounts for data set D is the theory that maximizes the posterior probability

$$P(T|D) \propto P(D|T)P(T) = \sum_M P(D|M)P(M|T)P(T)$$

where we have expanded $P(D|T)$ as a sum over all possible models M of theory T. To complete our framework we need to specify a prior on theories $P(T)$, along with distributions $P(M|T)$ and $P(D|M)$ that specify how models are generated from theories and how data are generated from models.

$P(T)$: A prior distribution on theories

The learning framework described in the previous section is extremely general, and can be combined with many different proposals about how theories are mentally represented. We work with the idea that theories are mentally represented in some language, and that the prior probability of any theory is inversely related to the length of its description in this language. As a starting point, we work with a language that is closely related to function-free first-order logic. The theory in Figure 1 is expressed in this language, and many additional examples are shown in Figure 2.

Our language includes symbols representing predicates of different arities (e.g. $\mathtt{M}(\cdot)$ and $\mathtt{S}(\cdot,\cdot)$) variable symbols (e.g. $\mathtt{x}$, $\mathtt{y}$ and $\mathtt{z}$), and Boolean connectives which capture negation ($\neg$), conjunction ($\wedge$), disjunction($\vee$), material implication ($\rightarrow$), and biconditional implication ($\leftrightarrow$). Four quantifiers are included: for all ($\forall$), there exists at least one ($\exists$), there exists one or fewer ($\nexists$), and there exists precisely one ($\exists!$). The language includes the identity symbol ($=$), and we also include an operator $\mathcal{T}(\cdot,\cdot)$, where $\mathcal{T}(\mathtt{R},\mathtt{C})$ indicates that the transitive closure of $\mathtt{R}(\cdot,\cdot)$ is $\mathtt{C}(\cdot,\cdot)$ (in other words, that $\mathtt{C}(\mathtt{x}_1,\mathtt{x}_n)$ is true if and only if there is a set $\{\mathtt{x}_1,\mathtt{x}_2,\ldots,\mathtt{x}_n\}$ such that $\mathtt{R}(\mathtt{x}_1,\mathtt{x}_2)$ is true, $\mathtt{R}(\mathtt{x}_2,\mathtt{x}_3)$ is true, and so on).

Many aspects of this language are inherited from standard treatments of first-order logic, but several representational choices deserve some attention. First, note that the language includes no symbols for constants. We are especially interested in how abstract theories might be learned, and it will be convenient to restrict our attention to laws that do not refer directly to individual objects. Second, the language includes two quantifiers ($\nexists$ and $\exists!$) that are missing from most logical languages, in part because statements that rely on these quantifiers can be rewritten as statements that use the familiar existential quantifier and the identity symbol instead. These conversions, however, often produce long and unwieldy statements, and our language is based on the hypothesis that $\nexists$ and $\exists!$ are no more complex psychologically than the familiar quantifier $\exists$.

The transitive closure operator $\mathcal{T}(\cdot,\cdot)$ represents our greatest departure from familiar first-order logic. Unlike the quantifiers $\nexists$ and $\exists!$, the operator $\mathcal{T}(\cdot,\cdot)$ cannot be defined within a first-order language. The concept of transitive closure, however, seems psychologically natural, and is probably one of the most important concepts that cannot be formulated in a first-order language. First-order logic provides a simple starting point for investigations of the language of thought, but there is no reason to think that mental representations are limited to a first-order language. We believe that attempts to formalize the language of thought will eventually need to draw on the expressive resources of higher order logics, and the operator $\mathcal{T}$ is a preliminary step in this direction.

Given any theory T expressed in our language, the prior probability $P(T)$ is determined by the number of symbols in T: $P(T) \propto \lambda^{|T|}$, where λ is a parameter between 0 and 1. For all applications in this paper we set $\lambda = 0.9$.

Once we have committed to a representation language and a setting of λ, the prior distribution $P(T)$ is unambiguously specified. This prior, however, depends critically on the language chosen, and our goal is to work towards a language

1. $\forall x\, \exists^{\leq 1} y\, R(x, y)$.	At most one outgoing edge per node
2. $\forall y\, \exists^{\leq 1} x\, R(x, y)$.	At most one incoming edge per node
3. $\forall x\, \exists y\, R(x, y)$.	At least one outgoing edge per node
4. $\forall y\, \exists x\, R(x, y)$.	At least one incoming edge per node
5. $\forall x\, \exists^{1} y\, R(x, y)$.	Exactly one outgoing edge per node
6. $\forall y\, \exists^{1} x\, R(x, y)$.	Exactly one incoming edge per node
7. $\exists x\, \exists^{1} y\, R(x, y)$.	At least one node with exactly one outgoing edge
8. $\exists y\, \exists^{1} x\, R(x, y)$.	At least one node with exactly one incoming edge
9. $\exists^{1} x\, \forall y\, \neg R(x, y)$.	Exactly one node with no outgoing edge
10. $\exists^{1} y\, \forall x\, \neg R(x, y)$.	Exactly one node with no incoming edge
11. $\forall x\, \forall y\, R(x, y) \rightarrow \exists^{\leq 2} z\, (\neg y = z \wedge R(x, z))$.	Zero or two outgoing edges per node
12. $\forall x\, \neg C(x, x)$.	No cycles in $R(\cdot, \cdot)$
13. $\exists x\, C(x, x)$.	At least one cycle in $R(\cdot, \cdot)$
14. $\forall x\, C(x, x)$.	$R(\cdot, \cdot)$ has a path between any pair of nodes
15. $\exists x\, \forall y\, \neg x = y \rightarrow C(x, y)$.	At least one node is an ancestor of every other node
16. $\exists^{1} x\, \forall y\, \neg x = y \rightarrow C(x, y)$.	Exactly one node is an ancestor of every other node
17. $\exists y\, \forall x\, \neg x = y \rightarrow C(x, y)$.	At least one node is a descendant of every other node
18. $\exists^{1} y\, \forall x\, \neg x = y \rightarrow C(x, y)$.	Exactly one node is a descendant of every other node
19. $\forall w\, \forall x\, \forall y\, \forall z\, \neg R(w, x) \vee \neg R(x, y) \vee \neg R(y, z)$.	No paths of length three
20. $\forall x\, \forall y\, \forall z\, R(x, y) \wedge R(y, z) \rightarrow R(x, z)$.	$R(\cdot, \cdot)$ is transitive
21. $\forall x\, \forall y\, \forall z\, R(x, y) \wedge R(y, z) \wedge \neg x = z \rightarrow R(x, z)$.	$R(\cdot, \cdot)$ is transitive with all self-edges removed.
22. $\forall x\, \forall y\, R(x, y) \rightarrow R(y, x)$.	$R(\cdot, \cdot)$ is symmetric
23. $\exists x\, \exists y\, R(x, y) \wedge R(y, x)$.	At least one symmetric edge
24. $\forall x\, \forall y\, S(x, y) \leftrightarrow R(x, y) \vee R(y, x)$. $\mathcal{T}(S, T)$. $\forall x\, \forall y\, T(x, y)$.	$R(\cdot, \cdot)$ is connected.
25. $\exists w\, \exists x\, \exists y\, \exists z\, \neg w = x \wedge \neg w = y \wedge \neg w = z \wedge \neg x = y \wedge \neg x = z \wedge \neg y = z \wedge$ $\forall u\, \forall v\, R(u, v) \leftrightarrow (p = w \wedge q = x) \vee (p = x \wedge q = y) \vee (p = y \wedge q = z)$.	Exhaustive description of the chain
26. (law not shown)	Exhaustive description of the tree
27. (law not shown)	Exhaustive description of the transitive relation
28. (law not shown)	Exhaustive description of the pair of cliques
29. (law not shown)	Exhaustive description of the ring
30. (law not shown)	Exhaustive description of the random relation

Figure 2: Thirty laws used to approximate the predictions of our model. Relation $C(\cdot, \cdot)$ is the transitive closure of $R(\cdot, \cdot)$.

that captures the language of thought as closely as possible. Our current language does not achieve this goal: note, for instance, that all of the Boolean connectives are assumed to produce the same cognitive burden, but it is well known that conjunctions are easier to learn than disjunctions in some contexts [1]. Although our current language is little more than a starting point, it allows us to demonstrate our theory-learning framework in action, and to show how proposals about knowledge representation can guide and be guided by proposals about theory learning.

$P(M|T)$ and $P(D|M)$: Generating models and data

Since we are working within a finite domain, the number of models for any given theory is finite. For any theory T, we use a distribution $P(M|T)$ which assigns uniform probability to any model M that is consistent with T, and zero probability to all remaining models.

A model M unambiguously specifies which statements are true, but often only some of these statements will be available to a theory-learner. Figure 1b, for instance, shows a case where a learner must reason about the social structure of a tribe given only a limited sample of positive examples. The distribution $P(D|M)$ should capture the assumptions that were used to generate the data D. In some cases, the data will include both positive ($S(a, e)$) and negative examples($\neg S(a, b)$), but in other cases a learner may observe only positive examples. In some cases, the data D will be known to be uncontaminated by noise, but in other cases the distribution $P(D|M)$ should acknowledge that some parts of the data D may not accurately represent the underlying model M. Our framework can handle all of these cases, but we will focus on a simple setting where $P(D|M)$ is 1 if each observation in D is true according to M, and 0 otherwise.

Now that all components of our model have been specified, we can combine them to discover the theory T that best accounts for a given data set D. Our framework makes two psychological contributions: our language for representing theories is a proposal about the language of thought, and our model predicts which theories people will infer when presented with a given data set. Note, however, that our approach does not attempt to capture the psychological mechanisms that might allow human learners to identify the theory T that maximizes $P(T|D)$.

Learning abstract relational laws

In principle, the formal framework we described can be used to study the acquisition of rich and complex theories. Here, however, we explore one of the simplest settings where abstract theoretical laws must be discovered. We consider problems where a learner observes a single binary relation $R(\cdot, \cdot)$ and must discover a theory that explains the structure of this relation. Working in this relatively simple setting allows us to provide a transparent demonstration of our model in action, and to test some of its behavioral predictions.

Any binary relation $R(\cdot, \cdot)$ can be represented as a graph where there is an edge from node i to node j if and only if $R(i, j)$ is true. The six binary relations that we will consider are shown in Figure 3a. Note that each relation is con-

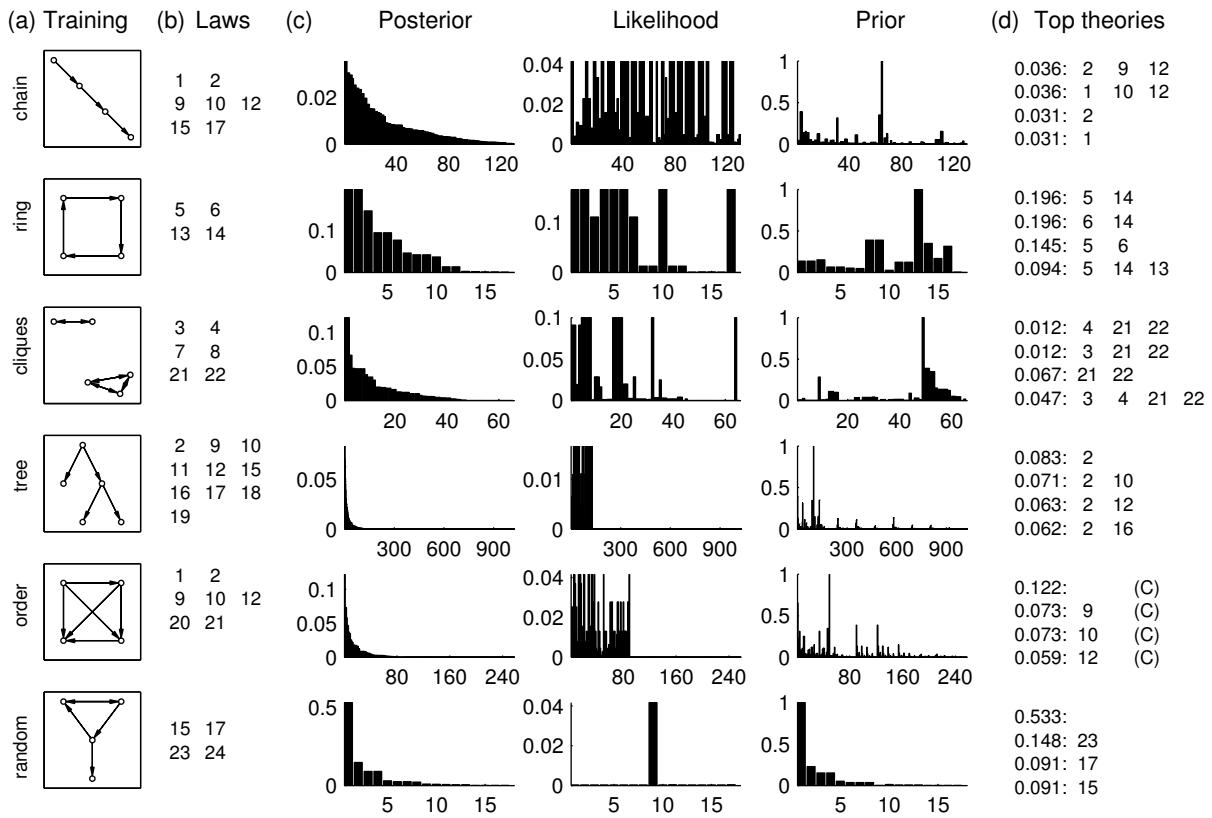

Figure 3: (a) Six binary relations provided as input to our theory learning framework. (b) Laws used to construct candidate theories for each relation (numbers correspond to laws in Figure 2). (c) Posterior distribution, likelihood function, and prior distribution for each relation. Each bar represents a theory, and the theories along the x-axis of each plot are sorted in order of decreasing posterior probability. (d) Top-scoring theories and their posterior probabilities. For instance, the two best theories for the ring have posterior probabilities of around 0.2, and the first of these theories includes laws 5 and 14. Theories marked with C are cases where the observed relation is defined as the transitive closure C of a latent relation R.

sistent with many abstract regularities: for instance, the first graph in Figure 3a is a graph where each node has at most one child, a graph where each node has at most one parent, a graph without cycles, and a connected graph. These and many other regularities can be formulated as abstract theoretical laws, and deciding which laws account best for a given relation is a challenging problem.

When considering potential explanations of each data set, our model uses a hypothesis space that includes all theories that can be formulated in the language we have chosen. Implementing this approach raises some formidable challenges, but we can approximate the predictions of our model. Figure 2 shows 30 theoretical laws that are consistent with one or more of the relations in Figure 3a, and we can approximate our model by restricting ourselves to a hypothesis space that includes any theory which corresponds to a set of laws from this list. In practice, even this approximation is challenging to compute, and we make one further simplification. For each relation in Figure 3a, we choose some subset of the laws in Figure 2, and consider all theories that can be created by combining laws from this subset. Future work can attempt to develop better approximations of our model, including approximations that do not rely on a hypothesis space of pre-specified laws. Even the rough approximation considered here, however, can provide some insight into theory learning.

The laws considered for each relation are shown in Figure 3b, and in each case we have tried to include the laws that seem most likely to produce theories with high posterior probability. Some theories include laws (e.g. law 12) that refer to relation C, and to any such theory we add the statement $\mathcal{T}(\mathtt{R}, \mathtt{C})$ which indicates that $\mathtt{C}(\cdot, \cdot)$ is the transitive closure of $\mathtt{R}(\cdot, \cdot)$. For the fifth relation (the order) we include two copies of any theory that refers to relation C: one where the observed relation is assumed to be R and any statement involving C merely places constraints on the extension of R, and one where the observed relation is defined as the transitive closure C of some latent relation $\mathtt{R}(\cdot, \cdot)$. Finally, we add add two extra theories to each hypothesis space: the empty theory which is consistent with any possible model, and a theory which simply enumerates the structure of the observed relation (see laws 25 through 30 in Figure 2).

When computing the posterior probability of a given theory T, the primary challenge is to compute the sum $P(D|T) = \sum_M P(D|M)P(M|T)$. Since the examples in Figure 3a all use relatively small domains, we compute this sum using the *nauty* program [8] to enumerate all models up to isomorphism, and to count the number of models consistent with each isomorphism class. Note that this approach is not in-

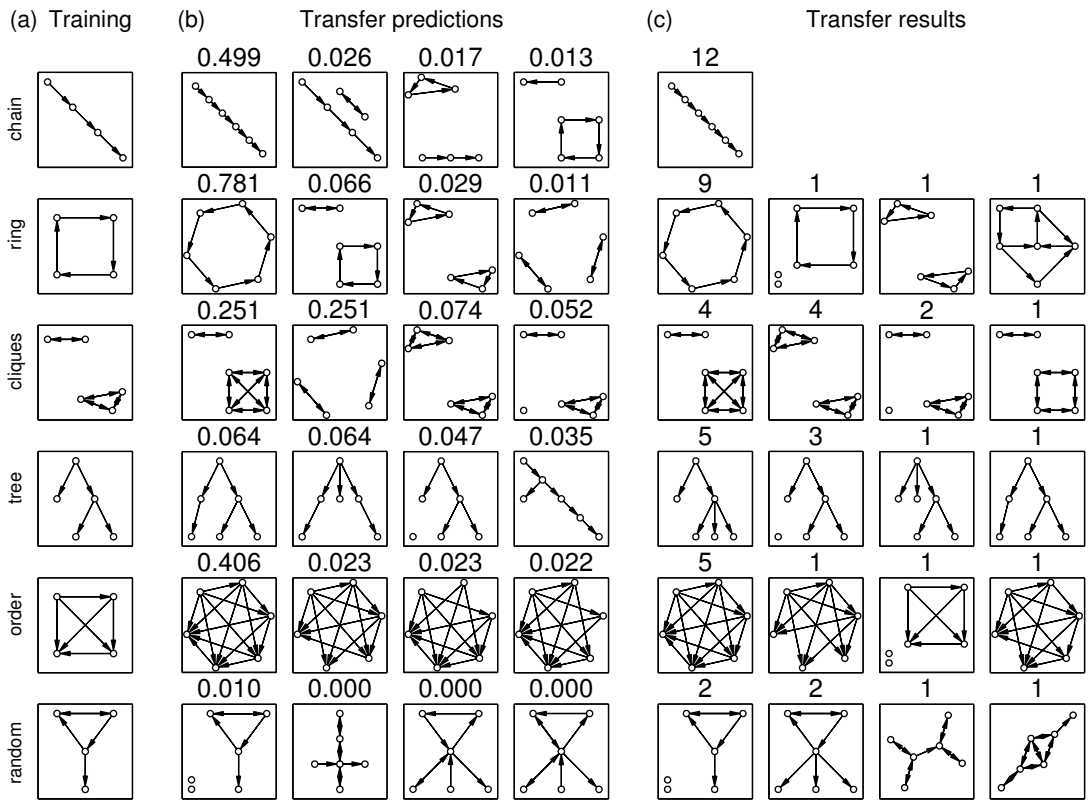

Figure 4: (a) Six training relations. (b) Transfer predictions generated by our hierarchical framework. Each prediction is labeled with its posterior probability. (c) Experimental results. The number of participants who made each prediction is shown.

tended as a model of psychological processing, and is merely a convenient way to compute the predictions of our model.

Figure 3c shows the prior distribution $P(T)$, the likelihood function $P(D|T)$ and the posterior distribution $P(T|D)$ for the six problems we consider. The candidate theories for each relation are arranged along the x-axis of each plot in order of decreasing posterior probability. The plots confirm that the prior and the likelihood both play important roles in our framework. Among all theories consistent with the data, the best are those that are consistent with few other possible data sets (i.e. theories with high likelihood) and that can be concisely described (i.e. theories with high prior probability).

Figure 3d identifies the top four theories for each problem. For instance, the equal top theory for the chain relation indicates that each node has at most one incoming edge (law 2), that there is exactly one node with no outgoing edges (law 9), and that there are no cycles (law 12). Some reflection should confirm that the only relations consistent with these constraints are chains. For each of the first five relations, the best theory according to our framework captures some of the regularities that are apparent in the data, but the best theory for the random relation is the empty theory.

Inductive predictions

Theory acquisition supports at least two kinds of inductive predictions. In previous work we have explored inferences about new or sparsely observed elements from a known domain [7]. After observing marriage ties within tribe T, for instance, a learner who concludes that "Bob tends to have lots of spouses" can predict that a new member of tribe T is fairly likely to end up married to Bob. Here we focus on inductive transfer, or predictions about domains that are entirely novel. After learning about tribe T, for instance, a learner may form expectations about the likely structure of novel tribe U. For instance, a learner who concludes that "exactly one person has lots of spouses" should expect that exactly one member of U will have many spouses.

Each relation $\mathtt{R}(\cdot,\cdot)$ in Figure 3a is defined over a domain that includes four or five entities. Given experience with one of these relations, our model makes predictions about the structure of novel domains. The posterior distributions $P(T|D)$ in Figure 3c provide the basis for these predictions: $P(R_{\text{new}}|D) = \sum_T P(R_{\text{new}}|T)P(T|D)$ where D represents the data observed for the original domain, and R_{new} is the version of relation R defined over the novel domain. For instance, if the posterior distribution $P(T|D)$ assigns high probability to theories which state that R is symmetric, then our framework predicts will predict that R_{new} is very likely to be symmetric.

Figure 4 shows the top transfer predictions for each of the six relations. In each case, we asked our model to make predictions about a novel domain with six entities. Even though this inductive problem is highly underconstrained, our framework makes predictions that seem relatively intuitive, and we tested these predictions in a behavioral experiment.

Experiment

We trained participants on the six relations shown in Figure 4a and asked them to describe similar relations over novel sets of entities.

Participants. 12 members of the MIT community were paid for participating in this experiment.

Materials and Methods. The experiment included six within-participant conditions that correspond to the six relations in Figure 3a. In each condition, participants learned a single relation (the training relation) then generated similar relations for two novel domains, one with six entities and the other with seven entities.

The cover story informed participants that they were learning about the organization of several small companies. Each relation in Figure 3a captures information flow within one of the companies. The experiment was carried out on a computer, and during the training phase the interface had a single button labeled "Observe." Upon clicking this button, participants were told about an event corresponding to one of the edges in the current training relation: they might be told, for example, that "John sends an envelope to Bill" (employee names were randomized across participants). After some number of observations, participants were given a test which included a yes/no question about each pair of employees (e.g. "Does John send envelopes to Bill?"). Participants continued to observe edges in the training relation until they were able to answer all of the test questions correctly.

After participants had passed this test, they were told about another company in the same industry with six employees, and were asked to "indicate one way in which the company might be organized." Responses were provided by checking boxes on a screen which included one box for each (directed) pair of employees.

Results. The relations most commonly generated for the six-employee companies are shown in Figure 4. These results confirm that people are able to discover abstract relational regularities given a single training relation. Our model provides a good account of these findings: for all conditions except the tree condition, the top (or equal top) choice is identical to the top (or equal top) choice according to our framework. Some aspects of the data, however, are not captured by our approach. For instance, people often choose a transfer relation that can be generated by connecting new nodes to a copy of the training relation, and this preference might explain why people do not generate a set of three pairs in the cliques condition, even though our model rates this configuration as its equal top choice.

Although our formal framework does not capture all aspects of our results, we know of no other computational framework that is likely to perform better. In particular, approaches (e.g. [7]) that simply search for the shortest description of the data in a first-order language will not be adequate. Note, for instance, that the shortest description of the ring is an exhaustive list of the edges in this relation, and contains no abstract laws that provide a basis for inductive transfer. The hierarchical aspect of our approach is essential for discovering abstract regularities in cases like this, and may capture one of the principles that allow humans to discover abstract theoretical laws.

Conclusion

Proposals about the language of thought and about theory acquisition should be tightly coupled. Knowing how theories are represented should allow us to predict which theories will be readily learned by people, and identifying the theories that people learn readily should provide important clues about the representations that support this ability. We described a computational framework that supports both kinds of investigation. We suggested that theories that are represented in a logical language, and explored the inductive consequences of one such language. Our language is proposed as a very rough approximation to the language of thought, but can certainly be improved in many respects. More important than the specific language we considered is the general framework we described: a framework for simultaneously exploring how theories are represented and how they are acquired.

Many knowledge representation schemes have been proposed by psychologists, computer scientists, and philosophers, including the lambda calculus, many flavors of logic, and several languages for representing semantic networks. Previous work, however, has not always focused on the inductive consequences of these languages. An important direction for future work is to combine some of these proposals with our framework, and to discover which languages lead to inductive predictions that best match human inferences. Probably no existing language will turn out to be entirely satisfactory, but understanding the strengths and weaknesses of existing representation languages should lead to future languages that correspond more closely to the language of thought.

Acknowledgments We thank Pooja Jotwani for helping to design and run our experiment.

References

[1] Bruner, J. A., Goodnow, J. S., and Austin, G. J. (1956). *A study of thinking*. Wiley, New York.

[2] Chang, C. C. and Keisler, H. J. (1973). *Model Theory*. Elsevier.

[3] Chater, N. and Vitanyi, P. (2003). Simplicity: a unifying principle in cognitive science. *Trends in Cognitive Science*, 7:19–22.

[4] Feldman, J. (2006). An algebra of human concept learning. *Journal of Mathematical Psychology*, 50:339–368.

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

[6] Goodman, N. D., Tenenbaum, J. B., Feldman, J., and Griffiths, T. L. (2008). A rational analysis of rule-based concept learning. *Cognitive Science*, 32(1):108–154.

[7] Kemp, C., Goodman, N. D., and Tenenbaum, J. B. (2008). Learning and using relational theories. In *Advances in Neural Information Processing Systems 20*, pages 753–760.

[8] McKay, B. D. (1990). *nauty* user's guide. Technical Report TR-CS-90-02, Department of Computer Science, Australian National University.

[9] Peirce, C. S. (1957). The logic of abduction. In Tomas, V., editor, *Peirce's essays in the philosophy of science*.

[10] Wellman, H. M. and Gelman, S. A. (1992). Cognitive development: Foundational theories of core domains. *Annual Review of Psychology*, 43:337–375.