Title
Learning Naïve Physics Models and Misconceptions

Journal
Proceedings of the Annual Meeting of the Cognitive Science Society, 31(31)

ISSN
1069-7977

Authors
Forbus, Kenneth
Friedman, Scott

Publication Date
2009

Peer reviewed

Learning Naïve Physics Models and Misconceptions

Scott E. Friedman (friedman@northwestern.edu)
Qualitative Reasoning Group, Northwestern University, 2133 Sheridan Rd
Evanston, IL 60208 USA

Kenneth D. Forbus (forbus@northwestern.edu)
Qualitative Reasoning Group, Northwestern University, 2133 Sheridan Rd
Evanston, IL 60208 USA

Abstract

Modeling how intuitive physics concepts are learned from experience is an important challenge for cognitive science. We describe a simulation that can learn intuitive causal models from a corpus of multimodal stimuli, consisting of sketches and text. The simulation uses analogical generalization and statistical tests over qualitative representations it constructs from the stimuli to learn abstract models. We show that the explanations the simulation provides for a new situation are consistent with explanations given by naïve students.

Keywords: Cognitive modeling; conceptual change; misconceptions; naïve physics; qualitative reasoning

Introduction

Many people have intuitive models of physical domains that are at odds with scientific models (Minstrell, 1982; McCloskey, 1983; diSessa, 1993; Brown, 1994). While productive for reasoning about everyday physical phenomena, these naïve models cause patterns of misconceptions. These misconceptions may result from improperly generalizing or contextualizing experience (Smith, diSessa, & Roschelle, 1994) or from incorporating instruction into a flawed intuitive framework (Vosniadou, 1994). Understanding how such intuitive models come about is an important problem for understanding how people learn physical domains (Forbus & Gentner, 1986).

We believe it is important for computational models of domain learning and conceptual change (e.g. Ram, 1993; Esposito et al., 2000) to encompass the learning of the initial intuitive concepts. This paper describes a simulation of learning intuitive physics models from experience. Experiences are provided as combinations of sketches and natural language, which are automatically processed to produce symbolic representations for learning. The system identifies and encodes instances of the concepts to be learned and constructs qualitative representations of behavior across time. Analogical generalization is used with a statistical criterion to induce abstract models of typical patterns of behavior, which constitutes our representation of intuitive models. These models can be used to make predictions and perform simple counterfactual reasoning. We compare its explanations to those of human students on a simple reasoning task (Brown 1994).

We next briefly summarize the relevant aspects of qualitative process theory and structure-mapping theory used in the simulation. Then we describe how our stimuli are represented and encoded, motivated by results and ideas from the cognitive science literature (diSessa, 1993; Talmy, 1988; Zacks, Tversky, & Iyer, 2001). The learning process itself is described next, followed by how the learned models are used in reasoning. We show that the simulation's explanations of a situation where a book is at rest on a table are compatible with student explanations (Brown, 1994). We close with other related work and future work.

Background

People's intuitive physical knowledge appears to rely heavily on qualitative representations (Forbus & Gentner, 1986; Baillargeon, 1998). Consequently, we use qualitative process theory (Forbus, 1984) as part of our model. In QP theory, physical processes are the mechanism of causality for changes in dynamic systems. However, the learning we are describing here is what provides the foundation for ultimately learning physical processes; in the framework of (Forbus & Gentner, 1986), we are modeling the construction of *protohistories* from experience, and building on those a *causal corpus* consisting of causal relationships between those typical patterns of behavior. To model these patterns of behavior, we use the concept of *encapsulated history* (EH) from QP theory.

An encapsulated history represents a category of abstracted behavior, over some span of time. It can include multiple qualitative states and events. Encapsulated histories are used when a learner does not yet understand how to reduce a behavior to physical processes. Encapsulated histories are a type of schema, and consequently have variables. The *participants* are the entities that an EH is instantiated over. The *conditions* are statements which must be true for an instance of the EH to be *active.* When an instance of an EH is active, the statements in its *consequences* are assumed to be true. Encapsulated histories are a form of explanatory schema: When instantiated, they provide an explanation for a behavior via recognizing it as an instance of a typical pattern, and furthermore can provide causal explanations, if there is causal information in the consequences.

Since EHs can include multiple qualitative states, they can be used for learning causal relationships between behaviors and properties of the world. In naïve mechanics, for example, the models of movement, pushing, and blocking learned by the simulation are represented by encapsulated histories. Figure 1 shows an EH learned by the simulation. This can be read as: *P1* pushes *P2* while *P1* and *P2* touch; the direction from the pusher *P1* to the pushed *P2* matches the direction of the push; and pushed *P2* consequently moves (*M1*) in the direction of the push. When given a novel test scenario, the EHs learned by the system are checked to see if there are entities that match the participants. If so, instances of those EHs are created. Each EH instance is active only if the statements in its conditions hold in the scenario. If the consequences fail to hold, that is a prediction failure for the EH.

```
define-encapsulated-history Push05
Participants:
Entity(?P1), Entity(?P2), PushingAnObject(?P3),
Direction(?dir1), Direction(?dir2)
Conditions:
providerOfMotiveForce(?P3, ?P1), objectActedOn(?P3,
?P2), dir-Pointing(?P3, ?dir1), touches(?P1, ?P2),
dirBetween(?P1, ?P2, ?dir1), dirBetween(?P2, ?P1,
?dir2)
Consequences:
causes-SitProp(Push05,
                (exists ?M1
                  (and MovementEvent(?M1),
                       objectMoving(?M1, ?P2),
                       dir-Pointing(?M1, ?dir1)))
```

Figure 1: **An encapsulated history relating pushing and movement.**

Our hypothesis is that people use analogical generalization to construct encapsulated histories. To model this process, we use SEQL (Keuhne et al., 2000). SEQL is grounded in structure-mapping theory (Gentner, 1983), and uses the Structure-Mapping Engine, SME (Falkenhainer et al 1989) as a module. Given two representations, a base and a target, SME computes a set of mappings that describe how they can be aligned (i.e. correspondences), candidate inferences that might be projected from one description to the other, and a structural evaluation score that provides a numerical similarity score. SEQL uses SME as follows. SEQL maintains a list of exemplars and generalizations. Given a new exemplar, it is first compared against each generalization. If the score is over the *assimilation threshold*, they are combined to update the generalization. Otherwise, the new exemplar is compared with the unassimilated exemplars. Again, if the score is high enough, the two exemplars are combined to form a new generalization. Otherwise, the exemplar is added to the list of unassimilated exemplars. The combination process maintains a probability for each statement in a generalization, based on how frequently it occurred in the exemplars generalized within (Halstead & Forbus, 2005). These probabilities are used in our simulation for doing statistical tests.

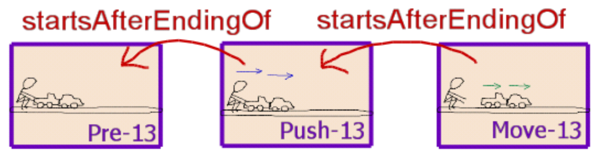

Figure 2: A sketched behavior

Multimodal Stimuli

To reduce tailorability, we provide experiences to the simulation in the form of sketches accompanied by natural language text. This serves as an approximation to what learners might perceive and hear in the world. The sketches are created in CogSketch[1] (Forbus et al., 2008), an open-domain sketch understanding system. In CogSketch, users draw and label *glyphs* to link the content of the sketches to concepts in CogSketch's knowledge base[2]. CogSketch automatically computes qualitative spatial relations between the glyphs such as topological relations, relative size, and positional relationships (e.g., above). Behaviors are segmented according to qualitative differences in behavior, such as changes in contact and actions of agents. This is common practice in qualitative reasoning research, and seems psychologically plausible (Zacks, Tversky, and Iyer, 2001). Each distinct state is drawn as a separate sketch. The sequential relationships between them are drawn as arrows on the *metalayer*, where other sub-sketches are treated as glyphs, as shown in Figure 2. Figure 3 shows a close-up of one of the sketched states. The child, truck, and car are glyphs in the sketch. The two right-pointing arrows are *pushing* annotations.

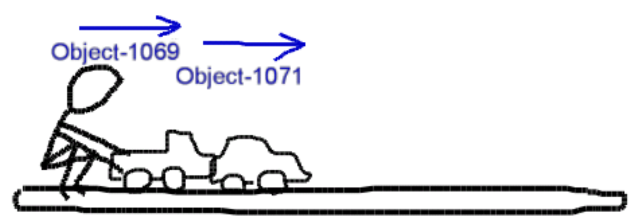

Figure 3: Example state drawn in CogSketch.

Our encoding of the physical phenomena of pushing, movement, and blocking as separate concepts is motivated by two lines of evidence. diSessa (1993) notes that people are unlikely to confuse successful resistance (i.e. a wall blocking a person's push) from nonsuccess (i.e. a ball moving due to tugging a string) in recalling events, and that these phenomena are encoded separately. Talmy (1988) attributes this separation of success and nonsuccess

[1] CogSketch is available online at http://spatiallearning.org/projects/cogsketch_index.html

[2] CogSketch uses a combination of knowledge extracted from OpenCyc (www.opencyc.org) and our own extensions for qualitative, analogical, and spatial reasoning.

encoding to varying language schemata between the two conditions.

For information not easily communicated via sketching, we use simplified English, which is converted to predicate calculus via a natural language understanding system (Tomai & Forbus, 2009). Here is one of the sentences from our example: "The child child-13 is playing with the truck truck-13." The special names **child-13** and **truck-13** are the internal tokens used in the sketch for the child and the truck respectively, so that linguistically expressed information is linked with information expressed via the sketch. This sentence leads to these assertions being added to the exemplar:

```
(isa truck-13 Truck)
(isa play1733 RecreationalActivity)
(performedBy play1733 child-13)
(with-UnderspecifiedAgent play1733 truck-13)
```

If the NLU system finds an ambiguity it cannot handle, it displays alternate interpretations for the experimenter to choose. But again, the choices are created by the NLU system: No hand-coded predicate calculus statements are included in the stimuli.

To be sure, this method of simulation input has limitations: Sketches are less visually rich than images, and they do not provide opportunities for the learner to experiment. Nevertheless, we believe that this is a significant advance over the hand-coded stimuli typically used by other systems, given the reduction in tailorability. CogSketch is being developed in part as a model of human visual perception, so there is independent support for many of its representational choices. Sketching and simplified English are natural human communication methods, so preparation of stimuli is simplified as well.

Learning

The simulation is provided with a set of target phenomena that it is trying to learn, here *pushing*, *movement*, and *blocking*. We assume that for a truly novice learner, words used in contexts of behaviors that they do not understand are clues that there is something worth modeling.

Given a new stimulus, a set of exemplars is produced, one for each occurrence of a target phenomenon. Since an instance of a particular phenomenon may continue across state boundaries, these occurrences can span multiple states. Temporal relationships between these occurrences are derived to support learning of preconditions and consequences. For example, consider a series of states S_1-S_3, where a man is pushing a crate in S_1-S_2 and not in S_3, and the crate moves in S_2-S_3 but not in S_1. The motion would have a `startsDuring` relationship with the pushing. Each stimulus observed by the simulation is automatically temporally encoded into exemplars using this strategy.

Generalizing behaviors

For each target phenomenon, the simulation maintains a separate copy of SEQL, a *generalization context* (Friedman & Forbus, 2008). A generalization context has an entry pattern that is used to determine when an exemplar is relevant. For example, the entry pattern for *pushing* is `(and (isa ?x PushingAnObject) (providerOfMotiveForce ?x ?y) (objectActedOn ?x ?z))`. Figure 4 shows the generalization contexts and their contents after the learning experiment described below. Multiple generalizations exist in Pushing and Moving contexts because certain exemplars are not structurally similar enough to share a generalization. Consequently, each generalization within a context represents a different behavior of the same concept. Our simulation currently operates in batch mode, only constructing models after all stimuli have been processed.

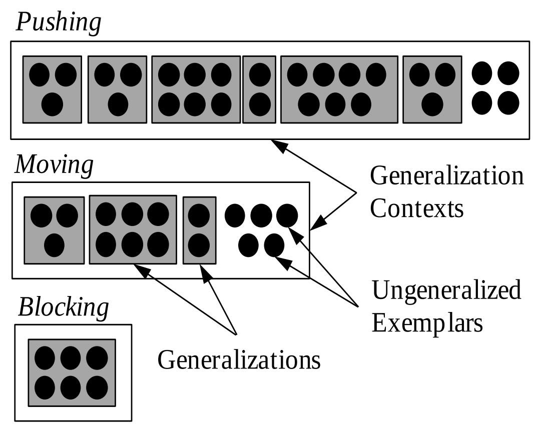

Figure 4: Generalization contexts after learning

Constructing intuitive models

The simulation creates encapsulated histories from protohistories in two steps: (1) Statistics are used to determine which generalizations are worth modeling with encapsulated histories, and (2) worthwhile generalizations are parameterized to create encapsulated histories. We discuss each step in turn.

Filtering generalizations

Not all analogical generalizations lead to useful encapsulated histories. If the conditions are too broad, inaccurate predictions will result. The probability information constructed during generalization provides an important filter. We assume a probability threshold t (here, 0.9) for correlation. That is, if any target phenomenon p is in a generalization with probability $P(p) \geq t$, then p is considered a *correlated phenomenon* within that generalization's context. A generalization is *decisive* if the binary entropy $H(P(p)) \leq H(t)$, for all phenomena p correlated with its generalization context. Entropy is the appropriate criterion to use because it measures information gain (i.e., low entropy implies high gain). All decisive generalizations are parameterized into encapsulated histories.

Extracting Causal Models from Generalizations

The system creates one encapsulated history per decisive generalization. Expressions whose probability is lower than the probability threshold *t* are filtered, thus reducing contingent phenomena. Expressions that remain are analyzed to determine what role they should play in the encapsulated history. An expression is held to be either (a) a *cause* of the state, (b) a *consequence* of the state, or (c) a *condition* that holds during the state, based on analyzing the temporal relationships involved. If the expression occurs before the start of the current state and persists until or throughout the current state, it is a possible cause. If an expression temporally subsumes or coincides with the state, it is a possible condition. If it begins during, with, or immediately after the end of the current state, it is a possible consequence.

Probabilities and temporal relationships are used to hypothesize causal relationships. For instance, in one generalization, movement starts *with* a pushing event with P=0.5, and starts *after* a pushing event with P=0.5. In this case, movement is not a likely condition for pushing because it only satisfies the temporal requirement half the time. Conversely, movement is a likely consequence, because starting *with* and starting *after* are both permissible temporal relations of consequences.

After the causes, conditions, and consequences are determined, the simulation defines an encapsulated history by introducing variables for the entities that appear in the conditions, existence statements for the entities that only appear in the consequences, and using the attribute information in the generalization to construct the participants information. Figure 1 and 6 illustrate. Notice that, while the learning process removes most irrelevancies, in `Block00` the entity `?P1` is included even though it is not causally relevant. It is there because the examples involving pushing all involve the pushing agent standing or sitting on a surface – so to the simulation, blocking must involve touching something else.

Reasoning with Encapsulated Histories

Given a new scenario, the simulation attempts to make sense of it by instantiating its encapsulated histories. For each EH, it finds combinations of entities that satisfy its `Participants` and `Conditions` constraints. When these constraints are completely satisfied for a set of entities, an instance of that EH is considered to be active, meaning that the statements in its `Consequences` are assumed to hold. As shown in Figure 1, this can include predicting new phenomena. When some of these constraints are violated, or some of the consequences are not satisfied, the EH instance can be used for generating counterfactual explanations, as explained below.

To illustrate, consider a scenario used by Brown (1994) and others, illustrated in Figure 5. The sketch shows a book on a table. Gravity pushes down on the book and the table. The scenario description includes two occurrences of pushing: gravity pushing the book and gravity pushing the table. The encapsulated history in Figure 6 can be instantiated sufficiently to be considered for inference by the simulation, since the criterion is that all non-event participants be identifiable in the scenario. Event participants need not be identified because these can be instantiated as predictions.

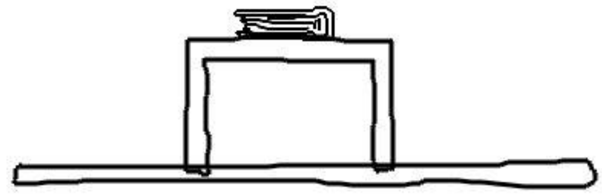

Figure 5: An example from Brown (1994) for testing learned knowledge

Specifically, activating `Block00` to explain gravity pushing the book requires assuming two additional events: (1) the book ?*P2* pushes some adjacent object ?*P3* in the direction *?dir1* of the initial push, and (2) the entity ?*P3* blocks the book ?*P2*. The table alone satisfies the constraints on *?P3*, binding the last of the non-event participants. This is sufficient grounds for the simulation to instantiate new pushing and blocking events, binding them to *?P6* and *?P7*, respectively.

```
define-encapsulated-history Block00
Participants:
Entity(?P1), Entity(?P2), Entity(?P3), Entity(?P4),
PushingAnObject(?P5), PushingAnObject(?P6),
Blocking(?P7)

Conditions:
providerOfMotiveForce(?P5, ?P2), objectActedOn(?P5,
?P3), providerOfMotiveForce(?P6, ?P3),
objectActedOn(?P6, ?P4), doneBy(?P7, ?P4),
objectActedOn(?P7, ?P3), dir-Pointing(?P5, ?dir1), dir-
Pointing(?P6, ?dir1), dirBetween(?P2, ?P3, ?dir1),
dirBetween(?P3, ?P4, ?dir1), dirBetween(?P3, ?P2,
?dir2), dirBetween(?P4, ?P3, ?dir2), touches(?P2, ?P3),
touches(?P3, ?P4), touches(?P2, ?P1)
```

Figure 6: An encapsulated history relating pushing and blocking phenomena

The simulation has two strategies for answering questions about a scenario. If the question concerns a phenomenon that is predicted by the EH instances it has created for the scenario, it answers based on that information, including any causal argument provided as part of the EH. If the question concerns some phenomenon that is not predicted, it assumes that phenomenon occurs and looks at what new EHs could be instantiated to explain it. The instantiation failures for those EH instances are provided as the reasons for the phenomenon not occurring, as shown below.

Experiment

To test whether this model can learn psychologically plausible encapsulated histories from multimodal stimuli, we compare explanations it provides for a question from Brown's (1994) assessment of student mental models. We start by summarizing Brown's results, and then we describe the conditions used for the simulation and compare its results. If the explanations used by the students and the simulation are compatible on the same reasoning task, then the simulation has learned psychologically plausible intuitive models.

Brown's results

A question about the scenario in Figure 5 was asked of students: *Does the table exert a force against the book?*

Brown reported that 33 of 73 students agreed that the table exerts an upward force on the book, because it must, in order to counteract the downward force of the book. This is the scientifically correct answer. However, the 40-student majority denied that the table exerted this force. Their reasons fell into five categories:

1. Gravity pushes the book flat, and the book exerts a force on the table. The table merely supports the book (19 students)
2. The table requires energy to push (7 students)
3. The table is not pushing or pulling (5 students)
4. The table is just blocking the book (4 students)
5. The book would move up if the table exerted a force (4 students)

We query our simulation similarly, to determine whether it can reproduce some of the reasons that students gave.

Simulation setup

Our simulation was implemented using the Companions Cognitive Systems architecture (Forbus et al., 2008). We used 17 multi-state sketches as stimuli, using examples motivated by the mental models literature cited earlier. This set did not include the test scenario. The SEQL assimilation threshold was set to 0.5 and the encapsulated history probability threshold was set to 0.9. The temporal encoding step resulted in 28 pushing exemplars, 16 moving exemplars, and 6 blocking exemplars. These exemplars produced ten generalizations across the three generalization contexts, as illustrated in Figure 4. Six of these generalizations were decisive, leading to the push$\rightarrow$move model of Figure 1, the push$\rightarrow$block model in Figure 6, and four additional models.

The four additional models learned by the system were not activated during this test scenario. Three EHs described movement behaviors caused by pushing, with minor variations in the conditions. The fourth EH describes classic "billiard ball" causality, with a push causing motion, which then causes another push and setting another entity into motion.

Comparison with human results

Given these EHs, how does the system perform? Upon receiving the test scenario, the system activates EHs to infer the additional events of the book pushing down against the table and the table pushing down against the ground.

For the query, since the simulation does not have the event of the table pushing the book as a current prediction, it uses the counterfactual strategy. Only the EH of Figure 1 can provide a possible explanation. Assuming this EH is active, the simulation gets a new prediction: The book should move upward as a result of the push. This prediction contradicts the book's lack of motion in the scenario. Consequently, it answers that the table does not push up on the book, because in that case the book would move upward, and it does not. This is essentially the same as answer 5, given by four students.

After the proof by contradiction, the system cites activated EHs in which the book and table jointly participate to explain their behavior in the scenario. Consequently, it uses the EH in Figure 6 to explain that gravity pushes down on the book, that the book pushes down on the table, and that the table blocks the book. This is similar to answer 4, given by four students. This explanation also resembles answer 1, given by 19 students, though the students cite the concept of support, which was not among the simulation's target phenomena. These results support the hypothesis that the models learned by the simulation are like those used by naïve students.

Could the system learn models corresponding to the other answers? If the target phenomena and corpus included the concept of support and energy, it seems likely to us that it could, but this is an empirical question. With a different corpus of examples – perhaps including examples like those used by Camp & Clement (1994) and the rest of Brown (1994) – the simulation may be capable of coming to the correct model. Answer 3 may rest on an interpretation of events being mutually exclusive, i.e., if the table is blocking, then it cannot be doing the other actions. Further experiments should clarify this.

Related Work

The closest simulations are the COBWEB (Fisher, 1986) model of conceptual clustering and INTHELEX (Esposito et al., 2000), which develops and revises prolog-style theories. COBWEB does unsupervised learning of hierarchical relationships between concepts, in contrast with our use of supervised learning (via entry patterns in generalization contexts) of causal models. COBWEB calculated probabilities of features, whereas SEQL provides probabilities of structured relations. INTHELEX uses refinement operators to model multiple steps in a trajectory of learned models, whereas we focus only on one transition, the first. Both COBWEB and INTHELEX used hand-represented input stimuli, whereas ours is derived by the simulation from sketches and natural language. Ram (1993) discusses SINS, a robot navigation system that retrieves cases, adapts control parameters, and learns new associations incrementally. While both our system and SINS develop concepts incrementally from experience, our system learns models of physical behaviors and causal laws, and SINS learns associations between environmental conditions and control parameters.

Lockwood et al. (2005) used CogSketch and SEQL to model the learning of spatial prepositions, using single sketches labeled with words, in contrast to the sequence of sketches labeled with sentences used here.

Discussion & Future Work

We have described how analogical generalization and qualitative representations can be used to model the process of learning initial intuitive models. To reduce tailorability, the simulation inputs were combinations of sketches and simplified English. The resulting explanations resemble a subset of those of given by human students on a scenario.

While we believe that this is a significant first step, there is much more to be done. First, a broader variety of phenomena must be tested, to provide more evidence as to generality. Second, we need to conduct statistical tests to determine how order-sensitive the simulation is, and how the quality of models learned varies with the number of examples provided. By comparing models learned with different numbers of examples, can we find sequences of models that correspond to known developmental trajectories? That will help determine how much of the development of mental models this simulation can explain. Finally, we plan to incorporate these ideas in a larger-scale model of conceptual change, where the quality and content of its predictions guide future learning.

Acknowledgments

This work was funded by the Office of Naval Research under grant N00014-08-1-0040. We wish to thank Dedre Gentner and Jason Taylor for discussions of concept formation with SEQL.

References

Baillargeon, R. (1998). A model of physical reasoning in infancy. *Advances in infancy research*, 3, 305-371.

Brown, D. (1994). Facilitating conceptual change using analogies and explanatory models. *International Journal of Science Education*, 16(2), 201-214.

Brown, D. & Clement, J. (1989). Overcoming misconceptions via analogical reasoning: abstract transfer versus explanatory model construction. *Instructional Science*, 18(4), 237-267.

Camp, C. W., & Clement, J. J. (1994). *Preconceptions in mechanics. Lessons dealing with students' conceptual difficulties.* Dubuque, IA: Kendall/Hunt Publishing Company.

Chi, M., Slotta, J., & De Leeuw, N. (1994). From things to processes: A theory of conceptual change for learning science concepts. *Learning and Instruction*, 4(1), 27-43.

diSessa, A. (1993). Toward an Epistemology of Physics. *Cognition and Instruction*, 10(2-3), 105-225.

Esposito, F., Semeraro, G., Fanizzi, N., & Ferilli., S. (2000). Conceptual Change in Learning Naive Physics: The Computational Model as a Theory Revision Process. In E. Lamma and P. Mello (Eds.), *AI*IA99: Advances in Artificial Intelligence, Lecture Notes in Artificial Intelligence* 1792, 214-225, Springer: Berlin.

Falkenhainer, B., Forbus, K. and Gentner, D. (1989). The Structure Mapping Engine: Algorithm and examples. *Artificial Intelligence*, 41, 1-63.

Fisher, D. H. (1987). Knowledge acquisition via incremental concept clustering. *Machine Learning*, 2,139-172.

Forbus, K. (1984). Qualitative process theory. *Artificial Intelligence*, 24, 85-168.

Forbus, K. and Gentner, D. (1986). Learning Physical Domains: Towards a Theoretical Framework. In Michalski, R., Carbonell, J. and Mitchell, T. (Eds.), *Machine Learning: An Artificial Intelligence Approach.*

Forbus, K., Lovett, A., Lockwood, K., Wetzel, J., Matuk, C., Jee, B., and Usher, J. (2008). CogSketch. *Proceedings of AAAI 2008.*

Forbus, K., Klenk, M. and Hinrichs, T. (2008). Companion Cognitive Systems: Design Goals and Some Lessons Learned. *AAAI Fall Symposium on Naturally Inspired Artificial Intelligence.*

Friedman, S. & Forbus, K. (2008). Learning Causal Models via Progressive Alignment & Qualitative Modeling: A Simulation. *Proceedings of CogSci 2008.*

Gentner, D. (1983). Structure-mapping: A theoretical framework for analogy. *Cognitive Science*, 7(2).

Halstead, D. and Forbus, K. (2005). Transforming between Propositions and Features: Bridging the Gap. *Proceedings of AAAI-2005*. Pittsburgh, PA.

Kuehne, S., Forbus, K., Gentner, D. and Quinn, B. (2000). SEQL: Category learning as progressive abstraction using structure mapping. *Proceedings of CogSci 2000*.

Lockwood, K., Forbus, K., & Usher, J. (2005). SpaceCase: A model of spatial preposition use. *Proceedings of the 27th Annual Conference of the Cognitive Science Society*.

McCloskey, M. (1983). Naive theories of motion. In: D. Gentner and A.L. Stevens, Eds. *Mental models*. Erlbaum, Hillsdale, NJ. 299–324.

Minstrell, J. (1982). Explaining the "at rest" condition of an object. *The Physics Teacher*, 20(1), 10-14.

Ram, A. (1993). Creative conceptual change. *Proceedings of CogSci 1993.*

Smith, J., diSessa, A., & Roschelle, J. (1994). Misconceptions Reconceived: A Constructivist Analysis of Knowledge in Transition. *Journal of the Learning Sciences*, 3(2), 115-163.

Talmy, L. (1988). Force dynamics in language and cognition. *Cognitive Science*, 12(1), 49-100.

Tomai, E. and Forbus, K. (2009). EA NLU: Practical Language Understanding for Cognitive Modeling. *Proceedings of the 22nd International Florida Artificial Intelligence Research Society Conference*.

Vosniadou, S. (1994). Capturing and modeling the process of conceptual change. *Learning and Instruction*, 4, 45-69.

Zacks, J., Tversky, B., & Iyer, G. (2001). Perceiving, remembering, and communicating structure in events. *Journal of Experimental Psychology. General.* 130(1), 29-58.