# Analogies versus Rules in Cognitive Architecture

**Kenneth D. Forbus** FORBUS@NORTHWESTERN.EDU
**Thomas Hinrichs** T-HINRICHS@NORTHWESTERN.EDU
**Maxwell Crouse** MAXWELLCROUSE2020@U.NORTHWESTERN.EDU
**Joseph Blass** JOEBLASS@U.NORTHWESTERN.EDU
Qualitative Reasoning Group, Northwestern University, 2133 Tech Drive, Evanston, IL, 60208 USA

## Abstract

The use of rules to encode procedural knowledge is a common modeling assumption used in cognitive architectures. In many architectures reasoning is viewed as just another skill, and the same rule mechanisms are used to implement it. This uniformity has its attractions, but we suspect it is an oversimplification. We argue that the reasoning that rules perform in many cognitive architectures might be better performed using analogical reasoning. The advantages of analogical reasoning include that it can provide effective reasoning even without complete or correct domain theories, including handling incomplete knowledge. This paper explores the analogy/rules tradeoff through the lens of two tasks, commonsense reasoning and natural language interpretation.

## 1. Introduction

How to encode and learn procedural knowledge is a core design decision in any cognitive architecture. This usually entails rules, often in the form of production systems (e.g. SOAR (Laird 2012), ACT-R (Anderson, 2009)). Other architectures encode procedural knowledge using a variety of rules and plan operators (e.g. ICARUS (Choi & Langley, 2017)). Historically one of the signature phenomena explained by cognitive architectures is skill learning, notably modeling the power law of learning. Even now, subtleties in skill learning are leading to finer-grained decompositions of what rules are in some architectures (e.g., ACT-R's PRIMS, SOAR's PROPs). The Companion cognitive architecture focuses instead on higher levels of cognition: the Rational and Social bands (Newell, 1990), ranging from minutes to months. For us, modeling textbook problem solving, conceptual change, and learning by reading are signature phenomena (Forbus & Hinrichs, 2017). The Companion architecture, following Gentner's (2003) suggestion that analogy is central to human cognition, treats analogy as a core operation. But our architecture also includes rules and HTN plans as essential mechanisms, and we are continually exploring the tradeoffs between analogy and rules. This paper summarizes progress and questions.

## 2. Background and Framing

We begin by briefly summarizing how rules and analogies are used in cognitive architecture, then summarize the tradeoffs in an abstract way, to set the stage for the task examples.

### 2.1 Rules in Cognitive Architecture

A variety of forms of rules have been used in cognitive architectures. The two most common architectural choices are production rules, which operate via forward-chaining over chunks (structured representations of data), and logic-like rules, which can operate with different control strategies over data encoded in predicate-calculus style statements. In both cases, it is assumed that there can be potentially many rules, leading to the knowledge deployment problem. In SOAR and ACT-R, rule and chunk selection are handled via spreading activation mechanisms intended to model human memories. In the Companion architecture, Horn clause rules are backward chained, similar to Prolog, but without any imposed ordering across rules. All relevant rules are run with respect to the *logical environment* of a query, i.e. a Cyc-style microtheory and what it inherits from. Computation only operates on data and rules visible within the logical environment.

Functionally, rules can play many roles. One is *encoding*, translating from input representations available to the system into forms more amenable for subsequent processing.[1] Another is *reasoning*, drawing conclusions from other information. A third is *acting* through predicates that, when concluded, are tied to the system's effectors. In architectures which include planners as an explicit module, acting is often restricted to primitive plan operators, and plans provide a procedural language for organizing multi-step operations. Thus, when control structures are needed, plans tend to be used (often HTN plans), although in other architectures control structures are provided by operators implemented via rules (e.g. SOAR).

### 2.2 Analogy in Cognitive Architecture

We use Gentner's (1983) structure-mapping theory of analogy, where analogy and similarity are modeled via a comparison process that aligns elements of structured representations. We have developed computational models of analogical matching, retrieval, and generalization that have been used to model existing psychological phenomena and have made successful psychological predictions.[2] The Structure-Mapping Engine (SME) provides a computational model of analogical matching. Analogical retrieval is modeled via MAC/FAC which uses a two-stage map/reduce process for scalability. Analogical generalization is modeled via SAGE, which uses SME and MAC/FAC to construct probabilistic structured representations incrementally from a stream of examples. These models have also been used in performance-oriented systems, and are used in the Companion architecture, via procedural attachments to predicates, which invoke these models on the representations denoted by their arguments.

Unlike similarity operations involving vectors, matrices, or tensors, which only produce a numerical estimate of similarity, SME produces *mappings* to describe a comparison. A mapping contains a numerical estimate of similarity, plus two other kinds of information. First, it provides *correspondences* that indicate which aspects of a description align with another. These explain the ways in which descriptions are alike. Second, it provides *candidate inferences* that describe

[1] ARCADIA (Bridewell & Bello, 2016) uses components intended to model aspects of visual processing which are flexibly combined based on attention to handle encoding; Companions use CogSketch for high-level visual perception, which operates at a coarser level of modeling and without an attention model.

[2] Space limitations preclude model citations, see (Forbus & Hinrichs 2017) for details.

how information from one description can be projected into the other using the correspondences. These provide a form of pattern completion, a way to conjecture new knowledge.

Because candidate inferences project information, analogical matching can be used to perform deduction, abduction, and induction (e.g. Forbus 2015). Consider for example an `implies` statement which is a candidate inference. If the mapping includes the antecedents for the implication, and the consequent is new, the inference is a deduction. If the consequent is known but one or more of the antecedents is not, the inference is an abduction. If the relationship in a candidate inference is not a connective, then it is an inductive hypothesis. Several other cognitive architectures have included analogy in some way as a component, e.g. some versions of PRODIGY included a derivational analogy system (Veloso & Carbonell, 1993) and AMBR (Kokinov & Petrov, 2001) implements analogy using micro-agents that also perform deduction and induction.

### 2.3 Tradeoffs in using analogies versus rules

It has been argued (Forbus & Gentner, 1997; Gentner & Medina, 1998) that there is a continuum of representations used in human learning, with the most concrete being specific examples, applied by analogy to new situations, and the most abstract being rule-like structures, constructed via analogical generalization. These sparse relational descriptions produced by SAGE are retrieved via MAC/FAC and applied to new situations via SME, providing an analogical mechanism for rule-like reasoning. This provides an attractive uniformity of mechanism. That said, Companions still use Horn clause rules for encoding, constraint checking, various forms of "glue" inference, and grammar rules for syntactic parsing. Whether this is an engineering approximation or something reflecting an architectural mechanism is an open question. Is there a transformation step, where sufficiently robust generalizations are converted into rules with explicit variables that are retrieved and applied via mechanisms besides analogy? Or can rules be eliminated? The analogy-based models shown for the two tasks in the rest of this paper illustrate the power to be found in some of those intermediate levels of representation.

## 3. Task 1: Commonsense Reasoning

Commonsense reasoning is one of the most difficult challenges in AI (Davis & Marcus, 2015). Massive amounts of knowledge are needed, but approaches vary in how it is encoded. Cyc, for example, focuses on axioms in a highly expressive logic. While we view this as a valuable approach, it leaves out experiential knowledge, the everyday experiences people often use in reasoning. Recent neural network approaches distill massive amounts of text into language models and attempt to provide commonsense predictions. For example, Yang et al. (2018) demonstrate that ordinal judgments about everyday properties can be derived from a carefully designed language model. This seems worth pursuing, although the outputs from such models currently can be quite noisy (e.g. Bosselut et al. 2019). Our hypothesis is that language-level models are too low-precision to handle the range of human commonsense reasoning, but general axioms omit experience. What role do such episodic memories play in commonsense reasoning and what is their structure?

Our approach is to organize episodic memory as cases, suitable for analogical reasoning and for distilling more rule-like knowledge via analogical generalization. We have proposed that such cases should be small *common sense units* (CSUs), consisting of just one or two events that are extracted from larger interconnected episodic memories (Blass & Forbus, 2017). These are concrete descriptions applied by analogy to predict or explain situations. Their size makes them compositional, and thus we do *analogical chaining*, treating CSUs like rules, except applied via SME. This approach has provided reasonable performance on a subset of a commonsense test.[3]

CSUs have several advantages over rules. They can be directly generated via natural language microstories, one to three sentences long, e.g. "The airplane malfunctioned. This caused the airplane to crash.", in a process that is far simpler than formulating expressive logical rules. Each CSU can be used for both prediction and explanation, depending on which way the analogy works, e.g. if an airplane malfunctions it might crash, and if an airplane crashes, it could be due to a malfunction. Unlike logical rules, they are not guaranteed to be sound, so using them is more of an evidential process. This can lead to bizarre conclusions at times, and being able to filter via additional experience about what combinations of events are common is one of several methods we are experimenting with to evaluate and guide such reasoning. A benefit of CSUs over logical axioms for abduction is that they are more focused and generate fewer possible explanations.

## 4. Task 2: Semantic interpretation of natural language

High-precision natural language understanding remains an open challenge. Analogy has been proposed as the key mechanism by which people learn language (Tomasello 2003) and we have some computational evidence that structure-mapping can be used to learn and apply constructions (McFate et al. 2017). That said, as an engineering approximation we often use a layered approach in our language experiments, including an off-the-shelf parser with hand-built broad-coverage grammar, linked FrameNet/OpenCyc semantic information, and Discourse Representation Theory to handle higher-order constructs like counterfactuals and quantification. The tradeoff between rules and analogy comes at higher levels of interpretation, where task and context help focus and elaborate the analyses produced by the domain- and task-independent processes.

The rule-based approach consists of finding *narrative functions* (Tomai & Forbus, 2009), which are abductive hypotheses about the role of statements in the ongoing description of the meaning of the text. Similar rules are also used to interpret texts whose semantics can be captured in terms of qualitative models (McFate et al. 2014). While effective, hand-coding rules is labor-intensive, and thus we have sought ways to automatically learn these higher levels of processing.

Our *analogical Q/A training* approach (Crouse et al. 2018) learns *query cases* for natural language question answering. The learning process operates by taking the space of interpretations produced by the NLU system for the questions (and for the answers, if weakly supervised data is given, or logical forms, if the task provides those) and uses connection graph techniques to find paths in the knowledge base that connect possible interpretations. For example, in "What states border Texas", the meaning of state as a geopolitical entity can be linked with the interpretation for "Louisiana", whereas a state of matter cannot. Query cases are automatically constructed as

[3] Limitations in our NLU system have to date prevented us from running Companions on the entire test.

small pieces of this analysis, so that they can be composed when answering questions. For example, having been trained on the example above, the system can already answer questions like "What states border states that border Nevada?" by applying the same query case twice. The application process during question-answering uses MAC/FAC to retrieve relevant query cases, thereby potentially providing scalability. Query cases consist of a consequent plus a set of antecedents. The analogical match of a query case to the interpretation of a new question provides substitutions for these statements, and the consequent is used in the query being constructed if a sufficient number of the antecedents are also included in the match. Analogical Q/A training provides considerable data efficiency over other machine learning techniques (Crouse et al. 2018) and has been used with other ML benchmarks as well as in a deployed system (Wilson et al. 2019). Whether analogical Q/A training can be extended to learn all of the narrative functions needed for broad natural language understanding is an exciting open question.

## 5. Discussion

Our results on analogical chaining and analogical Q/A training suggest that analogy can play a key role in cognitive architectures. Rules can be viewed as just one end of a continuum rooted in experience and refined via analogical generalization into increasingly abstract representations. Are all rules learned this way? Rules for encoding and reasoning seem excellent candidates, whereas action rules seem harder. Is there ever a transition to more traditional rules, complete with pattern variables and simpler pattern-matching strategies? Perhaps, and adapting chunking mechanisms found in SOAR and ACT-R might provide a path for this transformation.

The techniques used here are knowledge-intensive, e.g. using ontological reasoning to test conclusions, connection graphs over relational knowledge base contents in learning query cases. We expect these tasks only scratch the surface of the ways that cognitive architectures can benefit from large-scale knowledge bases. To help others explore these issues, our NextKB knowledge base, which integrates linguistic, visual, and conceptual knowledge, is publicly available under a Creative Commons Attribution-Only license, and the current version of SME is available as open-source software.[4] We encourage you to think about how your own cognitive architecture experiments might benefit from analogy and ample knowledge.

## Acknowledgements

This research was supported by the Air Force Office of Scientific Research and the Office of Naval Research.

## References

Anderson, J. R. (2009) *How can the Human Mind Occur in the Physical Universe?* Oxford University Press.

Blass, J. & Forbus, K. (2017). Analogical Chaining with Natural Language Instruction for Commonsense Reasoning. *Proceedings of AAAI 2017*.

[4] NextKB at www.qrg.northwestern.edu/nextkb/; SME at http://www.qrg.northwestern.edu/software/sme4

Bosselut, A., Rashkin, H., Sap, M., Malaviya, C., Celikyilmaz, A., & Choi, Y. (2019) COMET: Commonsense Transformers for Knowledge Graph Construction. *ACL*.

Bridewell, W., & Bello, P. (2016). A Theory of Attention for Cognitive Systems. *Advances in Cognitive Systems*, 4:1-16.

Choi, D., & Langley, P. (2017). Evolution of the ICARUS Cognitive Architecture. *Cognitive Systems Research*, 48:25-38.

Crouse, M., McFate, C.J., and Forbus, K.D. (2018). Learning from Unannotated QA Pairs to Analogically Disambiguate and Answer Questions. *Proceedings of AAAI 2018*.

Davis, E. & Marcus, G. (2015). Commonsense Reasoning and Commonsense Knowledge in Artificial Intelligence. *CACM*, 58(9):92-103.

Forbus, K. (2015). Analogical Abduction and Prediction: Their Impact on Deception. *AAAI Fall Symposium Series.*

Forbus, K., & Gentner, D. (1997). Qualitative mental models: Simulations or memories? *Proceedings of the Eleventh International Workshop on Qualitative Reasoning*, 1-8.

Forbus, K.D. & Hinrichs, T. (2017) Analogy and Qualitative Representations in the Companion Cognitive Architecture. *AI Magazine*.

Gentner, D. (1983). Structure-mapping: A theoretical framework for analogy. *Cognitive Science*, 7, 155-170.

Gentner, D. (2003). Why we're so smart. In D. Gentner and S. Goldin-Meadow (Eds.), *Language in mind: Advances in the study of language and thought* (pp.195-235). Cambridge, MA: MIT Press.

Gentner, D., & Medina, J. (1998). Similarity and the development of rules. *Cognition*, 65, 263-297.

Kokinov BN, Petrov AA. Integrating memory and reasoning in analogy-making: the AMBR model. In Gentner D, Holyoak KJ, Kokinov BN, eds. The Analogical Mind: Perspectives from Cognitive Science. Cambridge, MA: MIT Press; 2001, 161–196.

Laird, J. (2012). *The SOAR cognitive architecture*. Cambridge: MIT Press

McFate, C.J., Forbus, K. and Hinrichs, T. (2014). Using Narrative Function to Extract Qualitative Information from Natural Language Texts. *Proc's of 28th AAAI Conf. on AI*, Québec, Canada.

McFate, C.J., Klein, J. & Forbus, K. (2017). A Computational Investigation of Analogical Generalization of Linguistic Constructions. *Analogy 2017*.

Newell, A. (1990) *Unified Theories of Cognition*. Harvard University Press.

Tomai, E. and Forbus, K. (2009). EA NLU: Practical Language Understanding for Cognitive Modeling. *Proc's of the 22nd Int'l Florida AI Research Soc'y Conf.* Sanibel Island, Florida.

Tomasello, M. (2003). *Constructing a Language*. Harvard University Press.

Veloso, M. & Carbonell, J. (1993). Derivational Analogy in Prodigy: Automating Case Acquisition, Storage, and Utilization. *Machine Learning* 10:249-278.

Wilson, J., Chen, K., Crouse, M., C. Nakos, C., Ribeiro, D., Rabkina, I., Forbus, K. D. (2019). Analogical Question Answering in a Multimodal Information Kiosk. In *Proceedings of the Seventh Annual Conference on Advances in Cognitive Systems*. Cambridge, MA.

Yang, Yiben, et al. (2018). Extracting commonsense properties from embeddings with limited human guidance." *Procs of the 56th Annual Mtg of the Ass'n for Comp'l Linguistics.*