

Quantitative Explorations of Category Learning with Symbolic Concept Acquisition

Robert E. Wray (wray@soartech.com)

Soar Technology, Inc. 3600 Green Court Suite 600 Ann Arbor, MI 48105 USA

Ronald S. Chong (rchong@gmu.edu)

Department of Psychology, George Mason University, Fairfax, VA 22030 USA

Abstract

We are striving to create cognitive models that produce veridical human behavior for complex tasks in real time. This paper details the application of an existing model of human category learning, Symbolic Concept Acquisition, to model category learning within a complex, real-time, perceptual-motor task. The chief modeling challenge was quantitatively matching human data within the constraints imposed by theory and its implementation in an integrative cognitive architecture. We describe how we improved aggregate fits to human learning results by considering the significant differences in the learning trajectories of individual subjects.

Introduction

This paper explores the quantitative application of an existing model of human category learning, Symbolic Concept Acquisition (SCA) (Miller, 1993; Miller & Laird, 1996), to model rule-based category learning in a complex, real-time perceptual-motor task. The task is a simplified representation of air-traffic control (ATC). This paper provides the first report of quantitative comparisons of SCA models to human data.

SCA is an exemplar category learning model and has been shown to be consistent with a range of phenomena in human category learning such as typicality effects, response time effects for typicality and practice, comparable learning rates for linearly separable and non-separable categories, and base-level effects (Miller, 1993). SCA was also previously shown to be sensitive to the complexity of category type for rule-based classification, although prior to the publication of quantitative results (Nosofsky, Gluck, Palmeri, McKinley, & Glauthier, 1994).

We integrated the SCA model with task performance knowledge within the EPIC theory of perception and action (Kieras & Meyer, 1997) and the Soar cognitive architecture (Chong & Laird, 1997; Newell, 1990). Long-term, this approach provides us a means with which to explore interactions between learning and performance. For this paper, we focus on a task in which performance and learning were interleaved rather than simultaneous. Interleaving allows us to focus on the reproduction of human learning results using SCA. In the future, this approach will also enable us to assess

empirically the impact of more tightly integrated learning and performance as well as attempt to predict these effects with the ATC model.

The remainder of the paper describes the learning task and outlines the learning results obtained from human subjects. It reviews SCA and presents initial quantitative comparisons to the human data. Because our initial results provided poor fits to the aggregate human data, we examined the human individual data closely and discovered widely varying subject learning trajectories. In response, we created a number of model variants to better reflect the range of human responses. This population of models provides much better fits to the aggregate data. Thus, poor initial fits and constraints of theory led ultimately to a richer analysis and multiple models.

ATC Task & Human Experimental Results

The simplified air-traffic control (ATC) task requires that human subjects respond to a variety of events on a graphical “radar” display by moving and clicking a mouse pointer to select iconic aircraft on a computer monitor screen and select “buttons” on the display that perform specific tasks. For the performance task, subjects must “accept” new aircraft entering the controller’s airspace and “hand-off” aircraft exiting the airspace by clicking on the aircraft, clicking a message button on the display and clicking a “send” button. Although simplified in comparison to actual air-traffic control, the task is representative of the real-time decision making and visual-perceptual skill required of human controllers in air-traffic control (and many other monitoring and control tasks).

In addition to the performance task, subjects were instructed that their primary task was to learn to respond correctly to an “altitude change request” from aircraft. As in the performance task, a response consisted of choosing a button on the GUI, either an “allow” or “reject” button for the altitude change request. Positive or negative feedback (a green or red circle next to the aircraft) was presented immediately after a response. Although the performance task was included in the human experiments and a performance task model was integrated with the learning model, the human learning results showed no effect across several workload

conditions in the performance task. Thus, we limit the presentation below to the learning task alone.

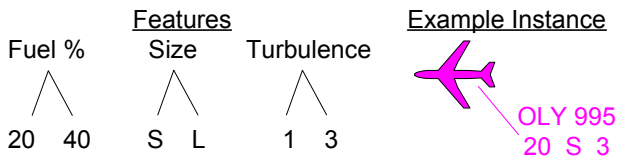


Figure 1 Instances: Features, values and an example

For each altitude change request, three values appeared adjacent to the aircraft icon on the screen. These values represented the attributes of percentage of fuel remaining, the size of the aircraft, and the value of turbulence. Each attribute could take two values. Figure 1 shows the possible values of the three inputs and a representation of one of the eight unique instances as it appeared on the simulator display.¹ The instance representations and responses were designed to map to the experimental methodology used for rule-based classification by Shepard, Hovland, and Jenkins (1961) and, subsequently, Nosofsky, Gluck et al. (1994). The eight unique stimuli were divided equally into four positive and four negative examples. Each subject was presented with instances representing one of three category structures: Type 1 (a single dimensional rule can be used to make all the classifications), Type 3 (a single dimensional rule plus exceptions describes all the instances), and Type 6 (all features must be considered; the exclusive-or problem). Other category types were not used in the ATC experiments.

Ninety undergraduates participated in the study, divided into three groups of thirty. Each group was presented with learning trials of one of the three category types interleaved with the performance task. Each learning trial was represented by a magenta-colored aircraft icon and the remainder of the screen was dimmed out to ensure subjects attended to the learning task. Each subject performed eight blocks of 16 learning trials and saw each instance twice in each block.

Figure 2 illustrates the aggregate human learning results, as well as the initial model results described below. These results show that simpler category types (i.e., those requiring the consideration of fewer dimensions) are learned more quickly than more complex ones, and qualitatively reproduce the results of Nosofsky, Gluck et al (1994).

¹ Researchers at BBN Technologies (Cambridge, MA) designed the experiment, implemented the simplified ATC task software, and collected half the subject data. Researchers at the University of Central Florida ran the remainder of human subjects. Our role was creating and running the SCA model and comparing the results to the human data.

Symbolic Concept Acquisition

One of us had previously developed a model of the ATC performance task (Chong, 2000), developed within the EPIC-Soar cognitive architecture (Chong & Laird, 1997).² EPIC-Soar integrates the perceptual-motor components of EPIC and the cognitive processor of Soar. Soar has been shown to model a wide range of human learning with a single, symbolic learning mechanism termed chunking. Because Soar provides the learning component and cognitive processor critical to this work, we focus on Soar in this discussion and discuss EPIC's role only where it impacts the results.

Using Soar introduces two important constraints for the category learning model. First, because chunking is the architecture's sole learning mechanism, we limit ourselves to this mechanism alone. For example, the original SCA model included a production-based algorithm for simulating frequency effects. Because frequency effects are not an architectural component, we excised them from the model.

Second, working within an architectural theory requires model reuse and cumulation (Newell, 1990). Because SCA is the only model of classification learning in the Soar framework, we chose to use it. One benefit of following these architectural constraints is that we realize another benefit of reuse: it lowers the cost of producing new models. Although developed over a decade ago, we were able to use the previously existing SCA model (including the production code). We introduced only monotonic changes (other than the removal of the frequency effect computations). As mentioned previously, this model had only been used for qualitative comparisons to category learning results. It was not clear how important the frequency effect simulation played in those results. Thus, an open question was to determine if this existing model could produce results that quantitatively matched human learning within the constraints of the architecture.

Table 1 presents a high-level representation of the SCA learning and prediction algorithm. When tasked to predict the category of an instance without feedback, SCA performs a specific-to-general search over prediction rules. It first attempts to recall prediction rules for all features (2), and then progressively abstracts³ features (3) to search for less specific prediction rules. The prediction algorithm is also used for learning. When feedback is present, SCA specializes the retrieved prediction rule (6) with the last feature abstracted from the instance (remembered at 4).

² A cognitive architecture is a software system that makes task-neutral commitments to cognitive (and, in the case of EPIC, perceptual and motor) mechanisms and processing.

³ We adopt the terminology used by Miller (1993). "Abstraction" here refers to the process of removing or ignoring a feature in the training instance.

It stores this specialized rule as a new prediction rule (7). Over multiple learning trials, this learning results in a general-to-specific search over the feature space. First, SCA learns rules sensitive to one feature, then to two, and so on. The concept representation becomes more specific as more features (and combinations of features) are incorporated into learned prediction rules.

Table 1 also presents an SCA learning example. The example assumes that the model has previously learned a prediction rule (Rule 3) that indicates an instance with a fuel percentage of 20 should be accepted. A new positive instance (S,3,20) is presented. For this example, assume the abstraction order is size, then turbulence, then fuel. There are no matching prediction rules for the input instance. SCA abstracts size from the instance, leaving (3,20). Again, it looks for prediction rules for this instance, and, finding none, abstracts the turbulence value, leaving (20). Rule 3 matches this instance. SCA now specializes Rule 3, adding the last abstracted feature and value (turbulence 3). The new Rule 4 indicates that (3,20) instances should be accepted. Had the example instance been negative, SCA would have learned a prediction rule that indicated instances with fuel values of 20 should be rejected. In this case, given the previously learned prediction rule (i.e., fuel 20 ⇒ accept), the model would have come to recognize that fuel values of 20 could not be used, by themselves, to make category predictions.

<ol style="list-style-type: none"> 1. instance = features and values /* from EPIC */ 2. while (no matching prediction rule for instance) 3. abstract (remove/ignore) feature from instance 4. remember most recently abstracted feature 5. if (no feedback) return prediction else 6. restore most recently abstracted feature to instance 7. store new prediction rule for instance
<p>SCA Learning Example:</p> <p>Available prediction rules:</p> <ul style="list-style-type: none"> Rule 1. (null) ⇒ accept Rule 2. (null) ⇒ reject Rule 3. fuel 20 ⇒ accept <p>New instance:</p> <ul style="list-style-type: none"> – size S, turbulence 3, fuel 20 – category accept – Abstraction order: size, turbulence, fuel <p>New Prediction Rule:</p> <ul style="list-style-type: none"> Rule 4. altitude 20, turbulence 3 ⇒ accept

Table 1 A pseudocode representation of the SCA algorithm (top) and an example learning trial (bottom).

SCA’s learning rate will vary based on the order of feature abstraction. Rather than simulate frequency effects, we considered three possibilities for abstraction order: (1) random abstraction order (slowest learning), (2) systematic abstraction order (fastest learning), and

(3) “relevant feature detection.” This last possibility adds knowledge to the model that recognizes when an ignored (i.e., abstracted) feature leads to an incorrect prediction. It then considers the feature that was ignored a “relevant feature” and abstracts it last in the future. This relevant feature detection allows the model to learn Type 1 (single dimension) categories very quickly. The relevant feature will generally change as the model is run, so that a Type 6 model subject will continue to try different features as relevant as classifications are learned.

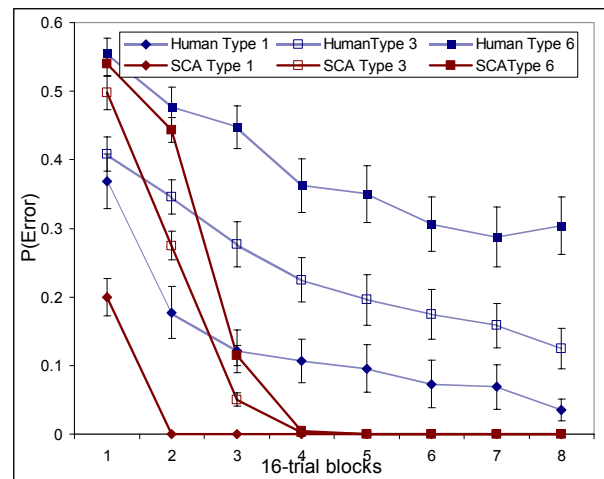


Figure 2 Initial SCA model results in comparison to human aggregate learning for the ATC Learning task

Figure 2 illustrates the initial results obtained from the SCA model with random abstraction order and relevant feature detection (30 model runs per category type). The model assumed a one-to-one mapping between relevant feature values from the display and those used by the model (e.g., a fuel percentage of “20” was represented as a single feature). These results reflect a poor fit to the human data; the G^2 aggregate fit statistic is 674.⁴ While there is a category and block effect (duplicating the qualitative results of Miller (1993)), the SCA mean learning rate for each category type was much faster than that of the human subjects.

Model Variations & Populations

Given the constraints imposed by the methodology and architecture, we turned to the human data to help us better understand what humans were learning during task performance and to provide guidance in adapting SCA to better model the human results. Again, a simpler solution might be to introduce new learning algorithms. However, because we are embracing the

⁴ G^2 , or deviance, is a measure of goodness of fit and often provides results similar to χ^2 . It is a sum of the logarithm of the ratio of observed to expected values; thus, the smaller the deviance, the better the fit.

constraints of the architectural approach, our model-building strategy is to first explore the possibilities of explaining the results within the context of the existing architecture (and learning model) before abandoning it for other approaches (Newell, 1990).

Individual Data & Models of Individuals

Figures 3, 4, and 5 plot the human individual data for Types 1, 3 and 6 category learning in the ATC task (subsequent sections explain the second set of SCA results also presented in these figures). While these figures are admittedly jumbled, they convey -- in a way that error bars fail to convey -- the large variance in human subject learning. For example, for Type 1, four subjects failed to recognize the relevant feature by the end of the final block; their error accounts for most of the deviation from zero error. For Type 6, one subject memorized the instances by block 2, while almost a third of the subjects were still performing at chance in the 8th block. In order to suggest some of this variance more clearly, we have highlighted the subjects with the highest and lowest average cumulative probability of error for each category type (dark lines, circled points).

In the SCA model used to generate Figure 2, there is no difference in knowledge from one simulated subject to the next and there are no learning parameters other than the (constant) subject knowledge. Hence, this model is more like a model of an individual than a population of subjects. Although SCA's learning rate appeared much too fast in comparison to the aggregate, the SCA results for the three category types are within the human brackets for each category type; thus, the SCA model provided results within the bounds of the fastest and slowest learners. Individual SCA learning curves also qualitatively matched the shape of some individual learning curves, while the aggregate data curve does not reflect the learning curve of any individual learners. SCA also provided an exact fit to nine Type 1 subjects and to at least one subject for each type when the first block (essentially random guessing) is ignored.

While the limitations of creating models that match aggregate data alone are widely recognized (Estes, 2002), this analysis at the level of individual learners provided evidence that the SCA model was not nearly as "wrong" as one-dimensional fits to the aggregate data suggest. This result provided the impetus to continue using SCA to model the ATC learning task. We now discuss improving the aggregate fit by introducing more complex feature mappings and introducing knowledge differences in subjects.

More Complex Feature Mappings

One major difficulty in modeling learning experiments is that models tend to focus only on a restricted set of features, those to which the experiment instructed the subjects to attend. However, the brain interprets its

environment, notices features, and learns continuously. Subjects (consciously or unconsciously) are likely detecting, and possibly using, all sorts of features aside from the ones instructed in the experiment.

Figure 3 Type 1 probability of error results by block.

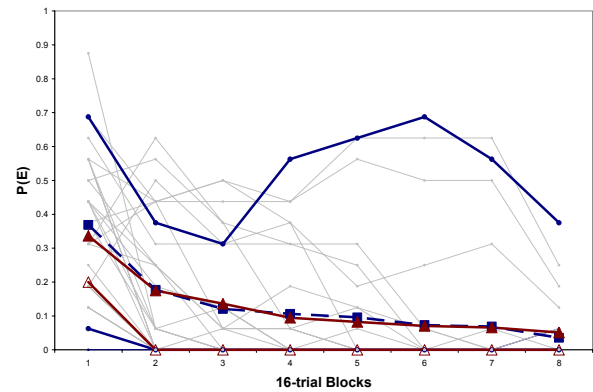


Figure 4 Type 3 probability of error results by block.

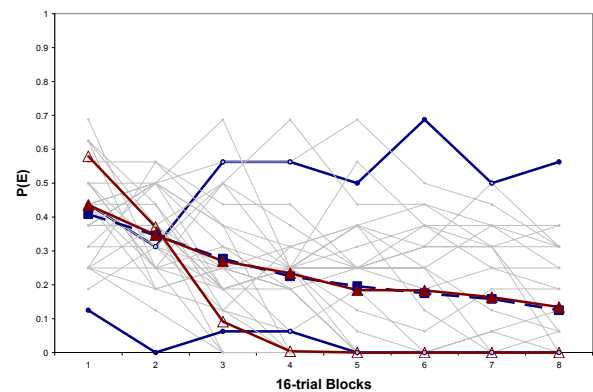


Figure 5 Type 6 probability of error results by block.

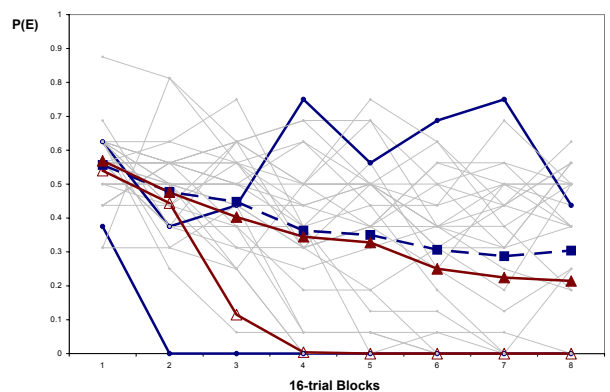


Figure Legend: Individual human subjects (grey), human subject brackets (●), human mean (■), original SCA mean (△), and final SCA mean (▲).

The initial SCA model for the ATC task employed only three binary-valued features to represent the feature

space for the learning task. Because SCA performs a refinement search over feature space when learning, its learning rate is sensitive to the number of features; they define the size of the search space. We empirically explored the sensitivity of SCA to the number of features. For example, for 4 features, the learning rate is not significantly affected. For 6 features, however, in the 8th block, probability of error is only slightly better than chance for Type 6 subjects. Thus, introducing additional features could help SCA improve the fit to the aggregate human data.

One possible source of additional features could be the additional information available on the screen. As Figure 1 shows, within the immediate vicinity of the instance features are the iconic representation of the aircraft (pointed in a particular compass direction), a text string representing the airline name, and a three digit flight number. Subjects were given explicit instructions to ignore all but the instance values but a few human subjects reported being influenced by such factors in a post-test questionnaire. Further, because participants likely perceived this information, it is still possible that it influenced their categorization processes even if not reported. Because this visual information could make its way into visual working memory in the model, the irrelevant visual information could also be incorporated into the learning decisions in the model.

The relevant features themselves also may be represented as more than binary, mutually exclusive values. For example, in the case of the fuel, the value is represented by 2 digits (“20” or “40”). Each value is also a feature with meaning; “1” is a single digit but also a scalar representing the degree of turbulence. Some co-occurring features have similar shapes which may require additional discrimination (e.g., “S 3” and “L 1”). Features can also be constructed from combinations of the relevant features. For example, some human subjects reported that during the learning experiment, they focused on “all high” and “all low” learning instances. Thus, it is plausible that more than just a single feature can be associated or constructed from an individual input value.

One of the positive consequences of our approach is the constraint of the architecture and model led us to consider these issues, because the architecture lacks other parameters that might mask these effects. While we would much prefer to be able to make *a priori* predictions of the features based on the representation, estimating the feature space is consistent with the decisions of other modelers. For example, in an icon search task model, Fleetwood & Byrne (2002) indicate that the features for their task, which include very simple shapes as well as more complex icons, were estimated from human data. Currently, models of

perception such as EPIC and ACT-R/PM do not inform or constrain the number of features needed for any particular percept (in part because such results are not available in the human factors or cognitive science literature).

Populations of Models

Looking at the individual data also suggested creating a population of models rather than a single model. We observed that for Types 3 and 6, some human subjects exhibited steady progress to zero error, some subjects made little improvements after repeated trials, and some “regressed,” exhibiting decreasing error for a number of blocks and then suddenly increasing. Interestingly, these patterns corresponded qualitatively to the three possible options for abstraction order outlined previously. A model with a fixed or “systematic” abstraction order will converge relatively quickly to zero error, even when critical features are abstracted early in the abstraction process, because less of the total feature space needs to be examined. A random abstraction order results in relatively slow progress because a much greater portion of the feature space will be examined. This “unsystematic” strategy leads to much slower progress when the number of features (and thus the feature space) grows larger. Finally, for Types 3 and 6, relevant feature detection can lead to increases in the error. Even when the relevant feature is incorrect, it will stabilize abstraction order for a time and a consistent portion of the feature space will be examined, leading to a decrease in the error. However, when the model recognizes another relevant feature candidate, it changes the abstraction order and moves to a different part of the feature space. This move can increase the error because the model may have learned few prediction rules in the new area of the feature space.

We re-ran the SCA model using a population of models with additional features and one of strategies (systematic, random, and relevant feature detection). Feature vectors ranged from zero to three extra features (i.e., from three to six total features). Extra features were represented as random, binary values. This provides twelve basic models (4 feature sets x 3 strategies). Having no guidance for choosing a distribution, model instances were chosen randomly from a uniform distribution. We ran 30 subjects for each category type using the identical uniform distribution for each category.

The lines with filled triangles in Figures 3, 4, and 5 illustrate the results for Types 1, 3 and 6 respectively. These results provide excellent fits for Type 1 and 3, and a reasonable fit for Type 6. The G^2 statistic for the aggregate fit is 9.96. Qualitative fits improve as well. Plots of individual model data look similar to the individual human results, and, for Types 1 and 3, the model generates comparable numbers of “perfect learners” (those that reach zero error by the final block).

Although other combinations of parameters improve the fit further, uniform distributions provide a good fit to the data with minimal additional assumptions.

Discussion and Conclusions

The original SCA model, representing a single strategy and subject, provided learning within the brackets of the fastest and slowest human learners and matched the learning of some individual subjects qualitatively and quantitatively. The SCA Soar model provided these fits *a priori*. We achieved these results by reusing an extant model, following constraints imposed by theory, and without introducing non-task knowledge or parameters. This methodology follows the one outlined in Newell (1990) as necessary for the cumulation of results for an architectural theory.

Achieving improved aggregate fits required analyzing individual data and developing a population of models with different strategies (reflected in the different methods for determining abstraction order) and different feature vectors. This approach begins to approximate the demand for more sophisticated models of human learning that match not only the aggregate data but also match the learning trajectories of individual subjects and successfully predict performance on transfer stimuli (Estes, 2002). We have not yet attempted a detailed comparison between individual SCA model runs and individual subjects; in general, however, correspondence between human and model runs improves over the original SCA model. We also have preliminary but encouraging transfer task data.

Because we introduced parameters not addressed by the EPIC or Soar theories, we will also evaluate whether we could match any aggregate learning curve. We would prefer to identify and quantify parameters *a priori* and show that we can use them in additional models.

The more significant reservation about introducing non-relevant features in SCA is that SCA's abstraction process is deliberate. Thus, even if irrelevant features were perceived, SCA, as currently conceived, should ignore them due to their irrelevance (as defined by the task instructions). While our immediate goal was to model this task within the existing model of SCA, we are investigating an alternative formulation of SCA that will use episodic indexing (Altmann & John, 1999). In this model, abstraction will occur as a consequence of attention and recall, rather than via a deliberate abstraction process. Preliminary results suggest this model will provide similar learning results but avoid the use of a deliberate abstraction procedure, which is the most psychologically questionable component of SCA.

Finally, this model is a first step towards exploring the interaction of learning and performance in perceptual-motor tasks. As learning and performance are more

tightly integrated, we expect the two will interfere with each other. The current model provides a potential foundation to understand and predict such interactions, as well as to model them.

Acknowledgments

This work was supported by the United States Air Force, Contract F33615-01-C-6077. We thank the program sponsors, other members of the AMBR program, Randolph Jones, Anthony Hornof and the anonymous reviewers for feedback and suggestions.

References

- Altmann, E. M., & John, B. E. (1999). Episodic Indexing: A model of memory for attention events. *Cognitive Science*, 23(2), 117-156.
- Chong, R. S. (2000). *Modeling with perceptual and memory constraints: An EPIC-Soar model of a simplified enroute air traffic control task*. (Final Report): USAF/AFRL contract F33615-99-C-6005.
- Chong, R. S., & Laird, J. E. (1997). *Identifying dual-task executive process knowledge using EPIC-Soar*. Paper presented at the 19th Annual Conference of the Cognitive Science Society.
- Estes, W. K. (2002). Traps in the route to models of memory and decision. *Psychonomic Bulletin and Review*, 9(1), 3-25.
- Fleetwood, M. D., & Byrne, M. D. (2002). Modeling icon search in ATC-R. *Cognitive Systems Research*, 3, 25-33.
- Kieras, D. E., & Meyer, D. E. (1997). An overview of the EPIC architecture for cognition and performance with application to human-computer interaction. *Human-Computer Interaction*, 12, 391-438.
- Miller, C. S. (1993). *Modeling Concept Acquisition in the Context of a Unified Theory of Cognition*. University of Michigan, Ann Arbor.
- Miller, C. S., & Laird, J. E. (1996). Accounting for graded performance within a discrete search framework. *Cognitive Science*, 20, 499-537.
- Newell, A. (1990). *Unified Theories of Cognition*. Cambridge, Massachusetts: Harvard University Press.
- Nosofsky, R. M., Gluck, M. A., Palmeri, T. J., McKinley, S., & Glauthier, P. (1994). Comparing models of rule-based classification learning: A replication and extension of Shepard, Hovland, and Jenkins (1961). *Memory & Cognition*, 22(3), 352-369.
- Shepard, R. N., Hovland, C. I., & Jenkins, H. M. (1961). Learning and memorization of classifications. *Psychological Monographs*, 75(13).