

Reply to “A Review of *Creating a Memory of Causal Relationships*”

Michael Pazzani

Department of Information and Computer Science

University of California

Irvine, CA 92717

pazzani@ics.uci.edu

In Pazzani (1990), I present a theory of learning that integrates explanation-based, theory-driven, and similarity-based methods. The goal of this research is to qualitatively model how differences in prior knowledge can account for differences in learning rates and to explain how this prior knowledge might be acquired and stored in memory (cf. Schank 1982). Cohen (this issue) does a commendable job of summarizing *Creating a Memory of Causal Relationships*. Here, I would like to clarify some issues that were raised in the review and one that was not mentioned. In particular, this reply will focus primarily on three points:

- OCCAM qualitatively models the data on human learning rates, but no quantitative comparisons are made between OCCAM’s learning rates and human rates.
- OCCAM is tested on hand-crafted problems, rather than natural problems.
- OCCAM uses explanation-based methods, but does not address the utility problem.

1 Qualitative computational models of human learning

Since OCCAM runs on a variety of problems, including problems whose training data corresponds to stimuli on which I have run experiments on human subjects, a quantitative comparison between OCCAM’s learning rate and that of human learners would be possible. However, I believe that it would not be a meaningful comparison. Instead, I have performed a qualitative comparison. In such a comparison, there is typically one dependent variable that

usually corresponds to the presence or absence of some prior knowledge. In computational models, it is easy to add or delete prior knowledge. In human subjects, prior knowledge is manipulated by including additional background knowledge in the instructions or by asking subjects to perform different learning tasks with the same stimuli¹. This dependent variable is manipulated in both a group of human subjects and in the computational model to produce four experimental conditions:

- Computational model with some (additional) prior knowledge
- Computational model without some (additional) prior knowledge
- Human subjects with some (additional) prior knowledge
- Human subjects without some (additional) prior knowledge

In a qualitative comparison, a computation model would be considered successful if a factor that changes the learning rate of the computational model produces a corresponding change in the human subjects. A quantitative comparison would require that the computational and human control group start have identical learning rates, and the control manipulated would produce an identical change. Human learning is quite complex, and there are a variety of known factors (e.g., attention, fatigue, motivation, cue salience) that influence learning that are not taken into account by OCCAM. In addition, although there is a correspondence between the input to OCCAM and the stimuli presented to human subjects, I think it is safe to assume that a human subject does not build a representation that contains exactly the same information as that encoded in the training data. Until all factors that affect human learning are identified and modeled, and the information that humans encode from stimuli is completely understood, it is premature to make quantitative comparison on complex learning tasks. However, a qualitative comparison does not require such a close correspondence, and allows one to evaluate models on how well they account for some factors of interest.

1. For example, one group of subjects has to identify which photographs belong to the category “alpha” and another group of subjects has to learn whether the person pictured in the photograph will be able to inflate a balloon. The prior knowledge of the “alpha” subjects is irrelevant to the task while the “inflate” subjects can use prior knowledge of the factors that influence the inflatability of balloons.

The qualitative evaluation of OCCAM is also appropriate since OCCAM is intended as one instantiation of the theory described in the book. This theory is a framework (or architecture) for integrating a number of learning methods. Of course, in order to implement this framework, a number of concrete decisions had to be made, including the choice of appropriate algorithms for clustering and generalizing observations. Improvements on any of these components (e.g., a different algorithm for unsupervised clustering of relational data-- an important topic worthy of further study) might result in different quantitative behavior on particular problems. In some cases, compromises on representations or processes were made to allow three different learning methods, as well as natural language understanding, question answering, and natural language generation systems to operate on the same memory structures. If each component of OCCAM were replaced by a different algorithm that performs the same function, I would accept that system as an alternate instantiation of the framework proposed in *Creating a Memory of Causal Relationships* provided that the qualitative findings are unaffected.

2 Research Methodology: A time for exploration and a time for consolidation

The problem addressed in *Creating a Memory of Causal Relationships* differs from that commonly studied in machine learning. It is designed to approximate the type of learning situation that young children and adults encounter when learning predictive relationships between actions and their consequences. The problem may be summarized as:

Given:

- A set of time intervals
- Each time interval includes
 - A set of actions
 - A set of state changes

Learn:

- Schemata that predict the state changes that occur when an action occurs.

This learning problem is particularly interesting when acquiring a group of related schemata, rather than a collection of unrelated schemata. Since this was an exploratory research project (Dietterich, 1990), there were no preexisting naturally occurring data sets that correspond to this situation. Therefore, data sets commonly used as test problems (e.g., predicting the party of a member of congress from their voting record) do not fit this problem. Even when there is a group of related concepts (e.g., different diseases of the thyroid) in the same data set, the relationships between the concepts are quite limited. For example, in the thyroid disease database, the concepts are all at the same level of generality and the same grain size. Realistic domains, involving common-sense reasoning, such as those that motivated this research, have concepts at different levels of generality (e.g., coercion vs. kidnapping and economic sanctions) and at different grain sizes (e.g., kidnapping vs. parents having a goal of preserving the health of their children).

To test OCCAM on knowledge intensive learning problems, two sets of problems were coded by hand. These problems are a group of kidnapping cases described in Moorehead (1980) and a group of economic sanction incidents described in Hufbauer and Schott (1985). Although hand-coded by the researcher, this data encodes events from actual kidnapping cases and economic sanctions incidents. However, to test OCCAM's data intensive learning methods, and to demonstrate that the knowledge needed by knowledge-intensive methods can be acquired via data-intensive methods, a variety of examples were constructed and encoded by hand. These include examples of coercion (e.g., "If you won't let me pitch, I'll take my ball and go home") and examples of preservation goals (e.g., a parent comforting a child who fell off a swing).

I would also have preferred to have a large collection of already coded problems on which to test OCCAM. In addition to allowing a more thorough evaluation, time that was allocated to reading original sources and hand-coding data could have been spent on further development and

perhaps resulting in improvements in some of OCCAM's components. However, I do not think that all research activities in machine learning (or any other field) should be focused on mainstream problems for which there is ample existing data and standard procedures for evaluation. Rather, some efforts should be expended on exploration of new problems, followed by more detailed investigation and consolidation of the issues raised during this exploration. After completing the book reviewed, I have embarked on a research plan to more thoroughly investigate issues raised during the development of OCCAM. This more thorough investigation and evaluation of learning methods that take advantage of prior knowledge and evaluation may please some. However, it is not without loss. In particular, individual algorithms running on simplified representations are more amenable to thorough evaluation than architectures. The result is a collection of loosely related articles on the theme of constraining learning with prior knowledge. Each result is obtained at the expense of ignoring certain aspects of the problem studied in OCCAM, such as the organization of memory, the modeling of human cognition, or the concern for detailed computational models. These articles explore the use of prior knowledge in learning from a number of mathematical (Pazzani & Sarrett, in press), psychological (Pazzani, 1991a), computational (Pazzani & Kibler, in press) and engineering (Cain, Pazzani, & Silverstein, 1991) points of view.

In further extensions to OCCAM, I have directly addressed the issue of obtaining data that is not hand-coded. In particular, a simulator was constructed based upon Meehan's (1981) Talespin program. Talespin generates English language stories that describe the actions that actors make to create changes in that state of the world that result in the achievement of the actors' goal. Furthermore, Talespin was modified so that several actions can occur simultaneously and so that actions have nondeterministic effects (e.g., a glass object usually, but not always, breaks when it falls). Since actions and state changes in both Talespin and OCCAM are represented in Conceptual Dependency, Talespin can be used to generate a stream of observations where each observation has a set of actions and a set of state changes. The result is a parametrized example

generator for OCCAM that allows systematic variation of parameters such as the amount of noise in the training data and the number of actions that can occur simultaneously.

This example generator was developed as part of an effort to address an issue that was left for future work in the book: the use of similarity-based methods to acquire the generalization rules² used by theory-driven learning (Pazzani, 1991b). Furthermore, the example generator has allowed more thorough exploration of theory-driven learning, the most novel of OCCAM's learning methods. Although an improvement over hand-coded data, it still represents an oversimplification of the world in which a child learns, particularly because the features from which to learn are unambiguously provided in the input representation (Wisniewski & Medin, 1991).

3. Speed-up learning

Let me close by addressing an issue that is not raised by Cohen, but one which I am often asked: "What are the implications of the OCCAM architecture on the "utility" (Minton, 1988) problem?" After all, OCCAM acquires redundant knowledge, and it is possible that it would take more time to make a prediction by traversing memory to find a very specific complex schema than it would take for an inference process to chain together several general simple schemata. The easy answer is that OCCAM does not address this issue at all. The reason is that the only process for making predictions is instantiating a single schema, so that OCCAM must acquire complex specific schema that are implications of existing simple schemata. An implicit assumption is that schema activation is a fast process while chaining is a slow process. Such an architecture is acceptable for an organism if individual schemata are slightly inaccurate, so that instantiating a schema is likely to produce few errors while chaining together several schemata is likely to produce many errors. By creating new schemata that correspond to observed implications of existing schema, islands of relative certainty can be created around situations that

2. Since writing the book, I have used the more meaningful name "causal pattern" instead of "generalization rule" to describe the knowledge structure used by TDL that encodes general knowledge of causality.

could be predicted to occur and that have actually occurred. Indeed, the A-EBL program (Cohen, in press) and the EBL+CBR (Cain, Pazzani & Silverstein, 1991) program can be viewed as instantiations of this principle.

After first avoiding the utility problem, let me propose an approach for addressing it. There has been a large amount of excellent research on this topic in an attempt to identify the conditions, such as constraints on the representation language (Tambe & Rosenbloom, 1989), under which acquiring redundant knowledge will speed up problem solving. To my knowledge, this research is not intended to address issues in human learning. Psychological experimentation can help machine learning by identifying processes or memory organizations that have evolved to work effectively while permitting storage, retrieval and utilization of some amount of redundant knowledge. For example, I can quickly verify that the product of two odd numbers is odd, but I am also capable of (more slowly) deriving this fact from more basic knowledge of multiplication, addition, and integers. However, until I started to write this sentence, I could not quickly verify that raising an odd number to an odd power results in an odd number, but had to resort to deriving this from basic principles. More careful experimentation (with a stop watch rather than arm chair) could reveal whether or not I have become slower at the question of multiplying odd numbers, as many EBL systems would. Further experimentation might reveal whether this effect exists and if so, whether it is permanent or fades with time perhaps as a function of the lack of use. Research on utility problem in human learners may provide useful insights into possible solutions to this problem in machine learning systems.

4. Conclusions

Progress in machine learning must consist of periods of exploration followed by periods of more thorough careful investigation of issues raised during exploration. The research reported in *Creating a memory of causal relationships* is exploratory in that it addresses a problem that was not previously investigated in the mainstream of machine learning research. However, I

feel that the problem studied was worthy of investigation and is worthy of continued investigation since it corresponds to an important part of the human learning process.

Acknowledgements

The work described here is supported by National Science Foundation Grant IRI-8908260. Dennis Kibler, Patrick Murphy and Ashwin Ram provided valuable comments on an earlier draft and on methodological issues in human and machine learning.

References

Cain, T., Pazzani, M., & Silverstein, G. (1991). Using domain knowledge to influence similarity judgments. *Proceedings of the Case-Based Reasoning Workshop*. Washington, DC: Morgan Kaufmann.

Cohen, W. (in press). Abductive explanation-based learning: A solution to the multiple inconsistent explanation problem. *Machine Learning*.

Cohen, W. (this issue) A review of “creating a memory of causal relationships.” *Machine Learning*.

Dietterich, T. (1990). Editorial: Exploratory research in machine learning. *Machine Learning* 5, 5-9.

Hufbauer, G. C., & Schott, J. J. (1985). *Economic sanctions reconsidered: History and current policy*. Washington, DC: Institute For International Economics.

Moorehead, C. (1980). *Hostages to fortune: A study of kidnapping in the world today*. New York: Atheneum.

Meehan, J. (1981). Talespin. In R. Schank & C. Riesbeck (Eds.), *Inside computer understanding: Five programs plus miniatures*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Minton, S. (1988). *Learning effective search control knowledge: An explanation-based effect*. Ph.D. Dissertation. Computer Science Department. Carnegie Mellon University.

Pazzani, M. J. (1990). *Creating a memory of causal relationships: An integration of empirical and explanation-based learning methods*. Hillsdale, NJ: Lawrence Erlbaum.

Pazzani, M. (1991a). The influence of prior knowledge on concept acquisition: Experimental and computational results. *Journal of Experimental Psychology: Learning, Memory & Cognition*, 17, 3, 416-432.

Pazzani, M. J. (1991b) Learning causal patterns: Deliberately overgeneralizing to facilitate transfer. *Proceedings of the Workshop on Multi-Strategy Learning*, Harper Ferry: WV.

Pazzani, M., & Kibler, D. (in press). The role of prior knowledge in inductive learning. *Machine Learning*.

Pazzani, M., & Sarrett, W. (in press). A framework for average case analysis of conjunctive learning algorithms. *Machine Learning*.

Schank, R. (1982). *Dynamic memory: A theory of reminding and learning in computers and people*. Cambridge University Press.

Tambe, M., & Rosenbloom, P. (1989). Eliminating expensive chunks by restricting expressiveness. *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence* (pp. 731-737). Detroit, MI: Morgan Kaufmann.

Wisniewski, E. & Medin, D. (1991). Harpoons and long sticks: The interaction of theory and similarity in rule induction. In D. Fisher, M. Pazzani, and P. Langley (Eds.). *Concept Formation: Knowledge and experience in unsupervised learning*. San Mateo, CA: Morgan Kaufmann.