Learning from Examples using CIBRE: a Cognitive Instance-Based Rule Engine

Bradley J. Best and Nathan Gerhart

Adaptive Cognitive Systems, LLC, Boulder, CO

For Further Information

Visit our website at www.AdCogSys.com or contact us at Admin@AdCogSys.com

Introduction

Cognitive model behavior can be produced directly from human performance data, using case-based reasoning, instance-based modeling (Aha, 1991), or learning from examples (Simon & Zhu, 1988; Simon & Gobet, 1996). The ACT-R cognitive architecture (Anderson et. al., 2004) is often used as a platform to develop models using this methodology (e.g., Taatgen & Wallach, 2002; Gonzalez, Lerch, & Lebiere, 2003; Best & Lovett, 2006; Best, Dixon, & Speed, 2008).

The strengths of instance-based learning include simplicity, rapid initial learning, and its robust performance in noisy environments (Aha, 1991). Instance-based methods often require substantially more storage than corresponding rule-based methods, and are thus limited by efficiency of storing and retrieving instances. Even the most effective pruning methods, which reduce storage, do so by less than an order of magnitude (Wilson & Martinez, 1997; Brighton & Mellish, 2002). Further, while retrieval methods have been optimized using mechanisms such as multiresolution kd-trees (Deng & Moore, 1995), these approaches provide little gain over serial search when dimensionality exceeds 10. Thus, issues of efficiency may overwhelm instance-based approaches. For example, the ACT-R architecture has exhibited difficulty processing the volume of instances produced in a real-time driving environment (Best et. al., 2008). While other researchers have investigated large-scale memories using ACT-R (e.g., Douglass, Ball, & Rogers, 2009), our focus is on speed, where the system must execute well within a 100ms cycle time. We have implemented a Cognitive Instance-Based Rule Engine (CIBRE) to meet this requirement.

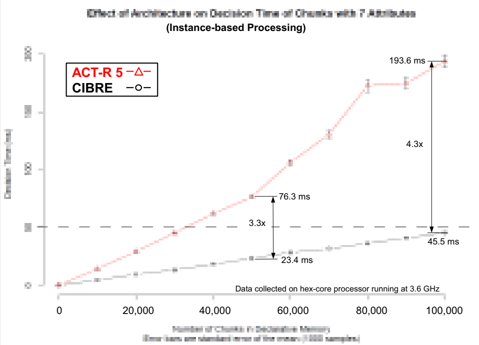

Figure 1: A comparison of Decision Times between ACT-R 5 and CIBRE

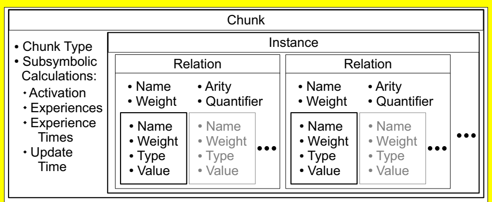

Figure 2: The structure of CIBRE chunks and instances

CIBRE Architecture

The CIBRE architecture is a light-weight instance-based learning system implemented in the LISP programming language. CIBRE depends on a forward- and backward-chaining rule engine for interactively achieving goals, and an HTN planner, which allows for evaluating potential futures prior to committing to any course of action. These high-level mechanisms provide the skeleton of cognitive behavior using CIBRE, while the instance-based methods provide the meat. CIBRE's core is an activation-based retrieval model, where activation follows a logarithmic decay over time, as in ACT-R. Unlike ACT-R, CIBRE includes support for cue-learning (using ***incremental*** rather than ***iterative*** methods), which increases accuracy, on-the-fly type definitions, which allows for learning in novel environments, and memory partitioning, which increases efficiency. CIBRE also includes an incremental pruning mechanism that dramatically reduces storage requirements.

CIBRE learns from a data set of instances and uses similarity matching to predict a response based on the values of attributes contained in the current context. Instances may contain multiple attributes and responses, and these items may be typed (e.g., values that are continuous or discrete). Classes are dynamically defined for instance types, allowing for flexibility in adding or deleting attributes from instance bases, and decreasing memory requirements (the name and type of an attribute are only stored once).

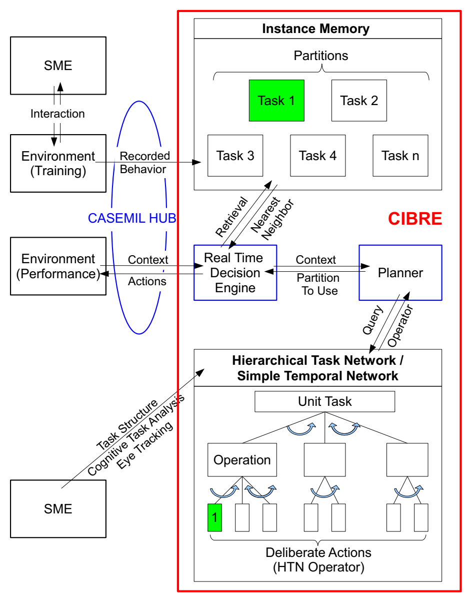

Figure 3: Architectural Diagram of the CIBRE Engine

Methods and Results

Our toolkit directly supports partitioning the instances within the domain into testing and training groups for validation. Using this functionality, we applied the CIBRE instance-based mechanism to 12 different public data sets drawn from the UCI Machine Learning Repository. CIBRE's response prediction accuracy for these data sets appears in Table 1 along with the predictions of two other instance-based machine learning systems (Wilson & Martinez, 1997; Brighton & Mellish, 2002).

CIBRE produces accuracy in both the 1/10th and 1/5th data set training/test split size that exceed the published benchmark results, differing from their performance by less than 3% on average across all databases and training/test split sizes (note that their focus was on ***iteratively*** pruning the data set, while ours was on retaining accuracy while using a faster ***incremental*** method). This consistency with the prediction performance of other published machine-learning algorithms indicates that the basic CIBRE implementation is able to detect and apply the feature-response relationship present within each data set.

	$9/10^{ths}$ predict $1/10^{th}$			$4/5^{ths}$ predict $1/5^{th}$		
Database	***Wilson & Martinez***	**CIBRE**	**% diff**	***Brighton & Mellish***	**CIBRE**	**% diff**
ANNEALING	*93.11*	**98.23**	5.50%	*95.28*	**97.74**	2.58%
BREAST-CANCER	*96.28*	**96.38**	0.10%	*95.76*	**96.55**	0.82%
CREDIT	***83.62***	81.88	-2.08%	***82.32***	81.59	-0.88%
GLASS	*73.83*	**90.95**	23.19%	*71.43*	**89.05**	24.66%
HEART-CLEV.	*81.19*	**81.33**	0.18%	*77.67*	**79.33**	2.14%
HEART-HUNG.	***79.22***	78.28	-1.19%	*76.55*	**77.24**	0.90%
HEART-SWITZ.	***92.69***	88.33	-4.70%	***92.08***	89.17	-3.16%
HEPATITIS	*80.62*	**84.00**	4.19%	*85.16*	85.16	0.00%
IRIS	*94.00*	94.00	0.00%	***95.00***	94.00	-1.05%
LIVER-DISORDERS	***65.57***	61.18	-6.70%	***59.71***	59.42	-0.49%
PIMA	***73.56***	66.32	-9.85%	***69.54***	66.41	-4.51%
WINE	*94.93*	**97.65**	2.86%	*84.57*	**98.29**	16.22%
AVERAGE	***84.05***	**84.88**	**0.96%**	***82.09***	**84.49**	**3.10%**

Table 1. CIBRE's percent correct response predictions for 12 public databases, compared to two benchmark algorithms. The most accurate algorithm per database is presented in bold.

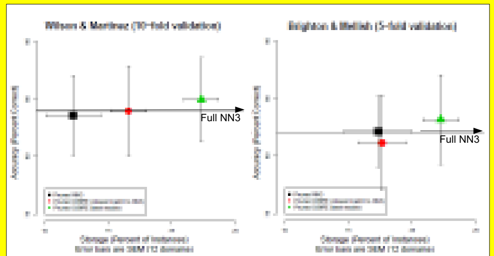

Figure 4: Comparison of Accuracy and Storage for Pruned Machine Learning Implementations. Pruned NN3 uses an iterative method of reducing storage, while CIBRE uses a faster, incremental method without sacrificing accuracy. The closest match (red) represents the CIBRE settings that most closely approximate the results of the two Pruned NN3 algorithms, while the best results (green) represent the CIBRE settings that yielded the best accuracy – storage trade-off.

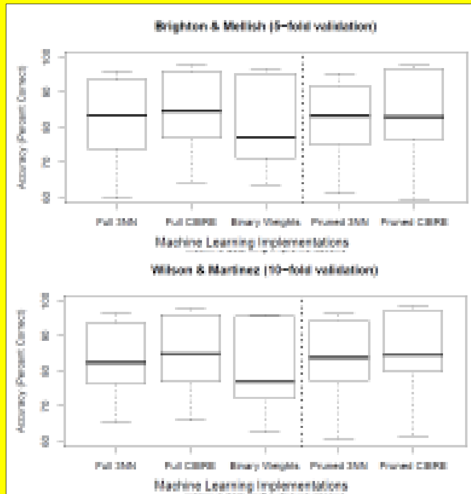

Figure 5: Comparison of Accuracy for Several Machine Learning Implementations. Box plots represent the mean, inter-quartile range, and range of accuracy across the 12 databases listed in Table 1. Implementations to the left of the dotted line stored all instances, while implementations to the right stored as few instances as possible, while maintaining accuracy.

References

Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y . (2004). An integrated theory of the mind. *Psychological Review,* 111, (4). 1036-1060.

Best, B. J., Dixon, K. R. & Speed, A. (2008). Modeling an Unstructured Driving Domain: A Comparison of Two Cognitive Frameworks. In *Proceedings of the 2008 Conference on Behavioral Representation in Modeling and Simulation.*

Best, B. J. & Lovett, M. (2006). Inducing a cognitive model from examples provided by an optimal algorithm. In *Proceedings of the Seventh International Conference on Cognitive Modeling* (pp. 56-61). Trieste, Italy.

Bellman, R. E. (1957). *Dynamic programming.* Princeton University Press.

Brighton, H., & Mellish, C. S. (2002). Advances in instance selection for instance-based learning algorithms. *Data Mining and Knowledge Discovery,* 6, 153-172.

Deng, K., & Moore, A. W (1995). Multiresolution Instance based Learning. In *Proceedings of IJCAI, 1995.* Morgan Kaufmann

Douglass, S., Ball, J., & Rodgers, S. (2009). Large declarative memories in ACT-R. In *Proceedings of the 9th International Conference of Cognitive Modeling* (paper 234), Manchester, United Kingdom.

Gonzalez, C., Lerch, F. J., & Lebiere, C. (2003). Instance-based learning in real-time dynamic decision making. *Cognitive Science* 27 (4), 591-635.

Simon, H. A., & Gobet, F. (1996). Templates in chess memory: A mechanism for recalling several boards. *Cognitive Science,* 31, 1–40.

Simon, H. A., & Zhu, X. (1988). Learning mathematics from examples and by doing. In *Models of thought* (Vol. II, pp. 167–184). New Haven, London: Yale University Press.

University of California Irvine Machine Learning Repository. http://archive.ics.uci.edu/ml/

Wilson, D.R., Martinez, T.R. (1997). Instance Pruning Techniques. In *Proceedings of the Fourteenth International conference on Machine Learning*, pp. 403-411.

Acknowledgments

The work presented here was funded in part by the National Science Foundation (NSF SBIR Phase I Award #0808857) and the Naval Air Warfare Center Training System Division (NAWCTSD)
Contract #: N61339-10-C-0048.

Additional thanks to Dr. Randy Wilson and Dr. Henry Brighton for their data and correspondence regarding their implementations.