Studies of the XCSI Classifier System on a Data Mining Problem

Chunsheng Fu
NuTech Solutions, Inc.
28 Green ST
Newbury, MA 01951
Chunsheng.fu@nutechsolutions.com

Stewart W. Wilson
Prediction Dynamics
30 Lang ST
Concord, MA 01742
Wilson@prediction-dynamics.com

Lawrence Davis
Nutech solutions, Inc.
28 Green ST
Newbury, MA 01951
David.davis@nutechsolutions.com

Abstract

Several machine learning techniques have been applied to the Wisconsin Breast Cancer (WBC) database, a publicly available data mining benchmark problem. Among these, a version of the classifier system XCSI achieved performance comparable to the best published results (Wilson 2000). This paper describes some modifications to the XCSI algorithm and to the parameter settings of XCSI that improved its performance noticeably on that problem. The modifications are robust in the presence of noise and appear to reduce the algorithm's tendency to overtrain.

1 INTRUDUCTION

XCS is a recently developed classifier system (Wilson 1995). XCSI is Wilson's extension of XCS to carry out classification for problems with inputs that are vectors of integers rather than bits (Wilson 2000). The Wisconsin Breast Cancer database (WBC), donated by Prof. Olvi Mangasarian, is a database of real-world data collected by Dr. William H. Wolberg to serve as a test case for classification data mining systems (Blake 1998). This paper will compare and contrast the modifications and parameter settings of this version of XCSI with Wilson's version on the WBC database.

2 EXPERIMENTS AND CONCLUSIONS

- On the WBC database, performance is slightly improved with an equal mix of choosing actions randomly and choosing the best action.
- Experiment showed that termination at 40,000 training steps is good for the XCSI described here. It is possible that the requirement of 100% training performance causes overtraining.
- Our version of XCSI here is relatively immune to random noise. The performance of the system with 5% random noise is about 0.5% lower than without the noise.
- This version does not use average population values as the initial values for parameters such as prediction, error, fitness, experience, last time in GA, and action set size for new classifiers created by the GA. Instead, these values are each set to initial values. Experiments on WBC database showed that resetting the variables to the average values of population decreases the performance.
- Extensive experiments showed that Wilson's default parameters are pretty good in general. The following modifications slightly increase the system's performance. The learning rate is 0.25 instead of 0.20. Using error threshold as 0.5% of the payoff range is optimal. Performance was severely degraded when both payoff and error threshold (e0) are too small.

Wilson's application of XCSI to the WBC database showed that integer-based classifier systems could be competitive in the real-world data-mining arena. This paper's modifications and extensive tests further confirm that XCSI does extremely well on the WBC database.

References

Blake, C. and C. Merz (1998). UCI repository of machine learning databases.

http://www.ics.uci.edu/~mlearn/MLRepository.html

Wilson, S. W. (1995). Classifier Fitness Based on Accuracy. *Evolutionary Computation*, 3(2), 149-175

Wilson, S. W. (2000). Mining Oblique Data with XCS. In: Technical Report No. 2000028, Illinois Genetic Algorithms Laboratory, University of Illinois at Urbana-Champaign.

Relating Two Cooperative Learning Strategies to the Features of the Found Concept Descriptions

Filippo Neri
Marie Curie Fellow at Unilever Research Port Sunlight
Quarry Road East, Bebington, Wirral, CH363JW, UK, Tel: +44 0151 6411314 and
Researcher at University of Piemonte Orientale, Italy
email: neri@di.unito.it

1 Extended Abstract

Concept learning [Dietterich and Michalski, 1983] is the task of finding a rule (in a wide sense) that discriminates between positive and negative instances of a given concept. The relevance of concept learning is well characterized by the variety of its fielded applications like prediction of mutagenetic compounds, and management of computer systems and networks [Lee et al., 1998, Neri, 2000]. Learning concepts means searching large hypothesis spaces. So, the capability to take advantage of effective search becomes a plus.

Approaches based on Genetic Algorithms proved their potentialities on a variety of concept learning tasks [De Jong et al., 1993, Giordana and Neri, 1995].

From these efforts it emerged that the main disadvantage of using GAs, with respect to alternative approaches, stays in their high user waiting time and in their high computational cost. A possible way of reducing GA computational cost is to use distributed computation efficiently: possibly by promoting cooperation among the simultaneous evolving populations. This approach is known as cooperative evolution or co-evolution [Husbands and Mill, 1991, Potter, 1997].

In co-evolution, a complex problem is decomposed into simpler subproblems at runtime, then the evolution of several species, each one oriented to a subproblem's solution, is promoted. Periodically, a candidate solution for the problem is assembled from the species' best individuals and evaluated. Finally, the solution evaluation is backpropagated to the existing species through a new problem decomposition that affects their further evolution.

Two cooperative learning strategies have been investigated. They show a different behavior with respect to the features of the found concept descriptions. We believe that a (distributed genetic base) learner able to exploit both cooperative strategies may acquire satisfactory concept descriptions across a wide range of applications. Further research to investigate this claim is in progress.

References

[De Jong et al., 1993] De Jong, K. A., Spears, W. M., and Gordon, F. D. (1993). Using genetic algorithms for concept learning. *Machine Learning*, 13:161–188.

[Dietterich and Michalski, 1983] Dietterich, T. and Michalski, R. (1983). A comparative review of selected methods for learning from examples. In Carbonell, J., Michalski, R., and Mitchell, T., editors, *Machine Learning, an Artificial Intelligence Approach*. Morgan Kaufmann.

[Giordana and Neri, 1995] Giordana, A. and Neri, F. (1995). Search-intensive concept induction. *Evolutionary Computation*, 3 (4):375–416.

[Husbands and Mill, 1991] Husbands, P. and Mill, F. (1991). A theoretical investigation of a parallel genetic algorithm. In *Fourth International Conference on Genetic Algorithms*, pages 264–270, Fairfax, VA. Morgan Kaufmann.

[Lee et al., 1998] Lee, W., Stolfo, S., and Mok, K. W. (1998). Mining audit data to build intrusion detection models. In *Knowledge discovery in databases 1998*, pages 66–72, Fairfax, VA.

[Neri, 2000] Neri, F. (2000). Comparing local search with respect to genetic evolution to detect intrusions in computer networks. In *Congress on Evolutionary Computation 2000*, pages 512–517, IEEE Press.

[Potter, 1997] Potter, M. (1997). *The Design and Analysis of a Computational Model of Cooperative Coevolution*. PhD thesis, Department of Computer Science. George Mason University, VA.