RUTGERS UNIVERSITY
Center for Expert Systems Research

Annual Report:
Empirical Analysis and Refinement of Expert System Knowledge Bases
Contract Number
N00014-87-K-0398
Office of Naval Research

August 31, 1988

Principal Investigators:
Sholom M. Weiss
Casimir A. Kulikowski

Table of Contents

1. Technical Project Summary

Knowledge base refinement is the modification of an existing expert system knowledge base with the goals of localizing specific weaknesses in a knowledge base and improving an expert system's performance. Systems that automate some aspects of knowledge base refinement can have a significant impact on the related problems of knowledge base acquisition, maintenance, verification, and learning from experience. The SEEK system was the first expert system framework to integrate large-scale performance information into all phases of knowledge base development and to provide automatic information about rule refinement. A recently developed successor system, SEEK2 [Ginsberg, Weiss, and Politakis 88], significantly expands the scope of the original system in terms of generality and automated capabilities.

Based on promising results using the SEEK approach, ~~we believe that~~ significant progress can be made in expert system techniques for knowledge acquisition, knowledge base refinement, maintenance, and verification.

2. Principal Expected Innovations

We are proposing to demonstrate a rule refinement system in an application of the diagnosis of complex equipment failure. The candidate application is computer network troubleshooting. The expert system should demonstrate the following advanced capabilities:

- automatic localization of knowledge base weaknesses
- automatic repair (refinement) of poorly performing rules
- automatic verification of new knowledge base rules
- some automatic learning capabilities.

3. Objectives for FY88

These were our objectives for the current year, Fiscal year 88:

1. functioning equipment diagnosis and repair knowledge base, suitable for refinement (in the area of computer networks).
2. initial demonstration of functioning equipment diagnostic system with capabilities of localization of weak rules, automatic refinement, automatic verification.
3. demonstration of initial rule learning capabilities.

4. Summary of Progress

Here are the highlights of progress has been made in meeting our stated objectives for fiscal year 88:

- Dr. Peter Politakis of the Digital Equipment Co. transferred to us DEC's Network Troubleshooting Consultant program that we proposed to use in our system. Dr. Politakis directed the development of this software and serves as our expert in the refinement of the knowledge base. Previously, we circumscribed the knowledge base to the following problem types: line, circuit, or cable problems. During the first half of the fiscal year 88 cited in Section 3, the subset of the knowledge base consisted of 287 observations, 138 hypotheses, and 324 rules. We further revised the knowledge base. At the present time the KB consists of 215 observations, 148 hypotheses, and 390 rules. The purpose of this application is to serve as a vehicle for further experimentation. We expect the knowledge base to remain stable for the remainder of the contract while we develop systems with advanced refinement and learning capabilities. Dr. Politakis obtained 72 real cases of network problems. We will supplement a core group of documented cases with simulated cases derived from verified correct rules in the knowledge base.

- During fiscal year 88, we developed a complete simulation and testing environment for refinement. Both a simulated case generation program, and a random rule basher were developed to enhance rule refinement experimentation.

- Substantial progress was made in our rule induction, i.e. learning system. Several experiments have been underway using data obtained from other researchers who have published results. These include data from Michalski and Quinlan. These efforts are extensions of the procedures we reported at the AAAI-87 conference [Weiss, Galen, and Tadepalli 87]. We note that unlike other fields, it is unusual for AI researchers to re-analyze other researchers data. Complete details of the experimental results have appeared in a technical report entitled *Minimizing Error Rates for Induced Production Rules.* A new technical report [Weiss 88a], summarizing the key ideas of the rule induction procedure (PVM), accompanies this annual report. Some experimental results are briefly reported in the next section.

In terms of the three objectives for fiscal year 88, we have completed the first objective: producing a functioning computer network diagnostic and repair knowledge base suitable for refinement.

The second objective was for an initial demonstration of functioning equipment diagnostic system with capabilities of localization of weak rules, automatic refinement, automatic verification. We believe the current system has these capabilities. However, the knowledge base we have produced is already quite accurate and therefore has limited potential for further refinement. While additional topics could be covered by adding many new rules, this is a not a principal objective. We have embarked on a *novel* approach to testing the system. Because the current knowledge base is considered correct, we have developed the following tools for experimentation:

- A case generator that randomly generates cases for given hypothesis from a correct knowledge base. This allows us to gather many more *simulated* cases than is otherwise possible.

- A rule modifier that randomly changes rules in a given knowledge base. In effect, it introduces random errors into the rules.

These tools will allow us to randomly modify a correct knowledge base and see whether the refinement system can recover from the errors. Figure 4-1 illustrates the experimental components that have been added to the usual refinement environment. These tools were completed during the 4th quarter, and the second fiscal year 88 objective was fully met.

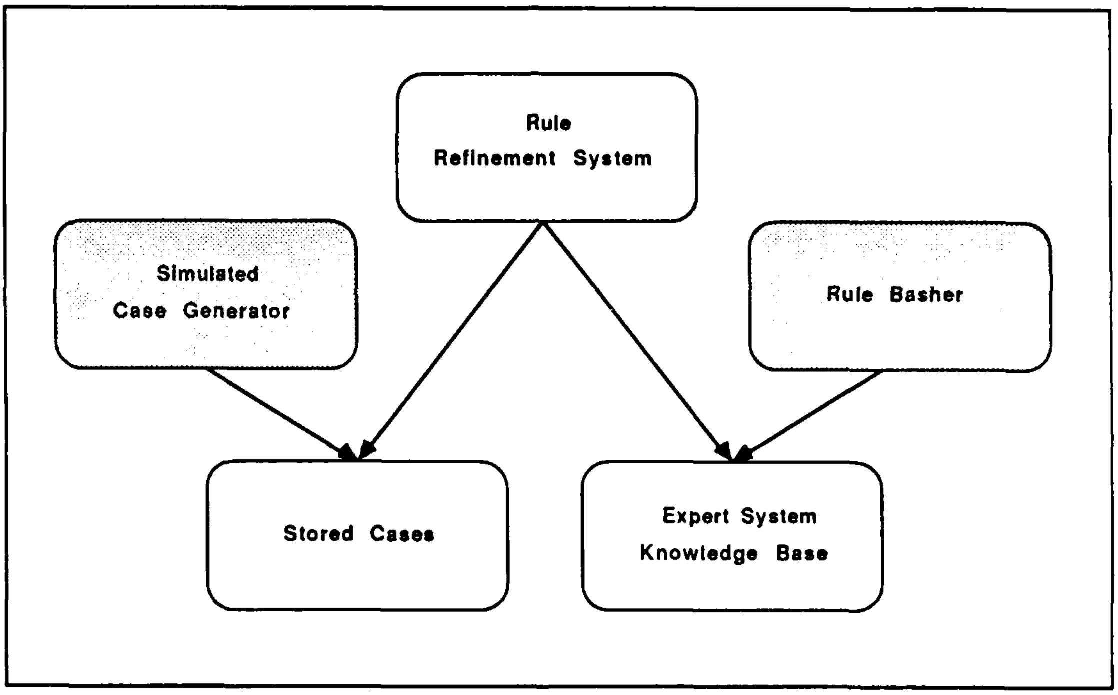

Figure 4-1: Enhanced Operational Environment for the Refinement System

The third fiscal year 88 objective is a demonstration of initial rule learning capabilities. The original SEEK2 refinement system did not add rules to a knowledge base. During the fourth quarter, we added a new heuristic to the refinement system. This allows the system to add a component to a rule in order to specialize the rule. This will make a rule fire less often. The precise details of this form of learning will be described in an upcoming technical report. In Section 5, we briefly review some of these procedures used to enhance the refinement system.

The addition of this heuristic to the refinement system meets the third objective of fiscal year 88: to demonstrate a knowledge based system that has some rule learning capabilities.

The work reported in the next section and in our technical report [Weiss 88b], further amplifies on a new approach to pure rule induction. For applications where a relatively short rule is required or can provide a good solution, our Predictive Value Maximization (PVM) procedure appears superior to other rule induction procedures reported in the literature. PVM is an autonomous induction system that learns rules in restricted situations. During the contract period we expect to integrate this procedure into the overall knowledge base refinement system. During the 4th quarter, we developed heuristics and procedures that can immediately produce a learning capability within the context of the current SEEK2 refinement system.

Progress in Rule Induction Techniques

During the current quarter, we completed our comparative experiments on rule induction. We have issued a technical report entitled *Minimizing Error Rates for Induced Production Rules*. We reproduce the abstract and a few of the key results below.

Abstract: Empirical techniques for induction of decision rules have evolved from procedures that cover all cases in a data base to more accurate procedures for estimating error by train and test sampling. Procedures that prune a set of decision rules and the components of these rules have been successful in increasing the performance of an induced rule set on new test cases. Recently, we reported on a technique for learning the single best decision rule of a fixed length. In this paper we show how resampling techniques for estimating error rates, can be integrated into this procedure for induction of decision rules. Superior results are reported on data sets previously analyzed in the AI literature.

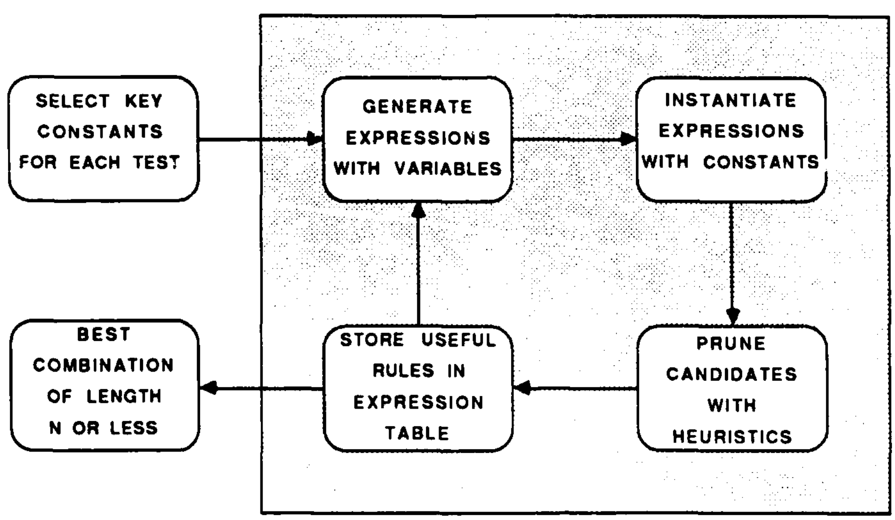

Figure 4-2: Overview of Heuristic Procedure for Best Test Combination

In 1987, we reported on a technique for learning the *single* best decision rule of a fixed length [Weiss, Galen, and Tadepalli 87]. In contrast to other methods of rule induction, the PVM rule induction procedure does not generate and prune a complete set of decision rules. Instead, this method is an approximation to exhaustive generation of all possible rules of a fixed length. While a true exhaustive search is not feasible in most applications, a small number of heuristics reduce the search space to manageable proportions. Figure 4-2 illustrates the key steps of the heuristic procedure. Experiments were performed on two sets of data for which published studies are available. The results are summarized in Figures 4-3 and 4-4.

Method	Variables	Rules	Error Rate
AQ15	7	2	32%
PVM	2	1	23%

Figure 4-3: Comparative Summary for AQ15 and PVM on [Michalski, Mozetic, Hong, and Lavrac 86] Data

Method	Variables	Rules	Errors (1985)	Errors (1986)
C4 pruned rules	8	2	31	43
PVM random resampling	8	2	17	30

Figure 4-4: Comparative Summary for C4 and PVM on [Quinlan 87] Data

We re-analyzed data that had been analyzed using prominent machine learning techniques. We showed that superior rules could be induced from these data sets. In the case of the Michalski data, a simple two variable rule produces better results than the more complex rules cited in the literature. While Quinlan's original data analysis produced excellent results, we showed that somewhat better rules could be induced than those he cited in his reports on thyroid disease.

For our analysis, we used classical resampling techniques [Lachenbruch and Mickey 68, Stone 74, Efron 82] to estimate error rates for nonparametric classifiers. These techniques can be time-consuming, but can lead to better induction results. Because PVM induces rules for a fixed, relatively short length, resampling procedures are a natural extension of the basic method. The major advantage is that error estimates can be derived, while essentially the complete data sample may be used for classifier design. While resampling is a natural fit to PVM, its use with other induction techniques is feasible [Breiman, Friedman, Olshen, and Stone 84].

We do not claim that PVM is always superior to other empirical rule induction procedures. Unlike AQ15 or C4, in practice PVM is limited to the induction of single short rules. However, if a good solution exists in the form of a single short rule, PVM has a decided advantage. Unlike

incremental empirical induction procedures that select one test at a time, PVM examines combinations of tests with varying constants. There are many applications, such as expensive instrument testing, where a short rule that limits the number of tests to be performed is a requirement.

In the upcoming fiscal year, we expect to report on comparative studies of rule induction, statistical pattern recognition and neural net techniques for classification.

5. Notes on Technical Progress

In this section, we briefly review some important issues in the development of the refinement system.

Case Generation

A simulated case generator has been built. Under the assumption that some rules are given to be correct, the system will randomly generate cases from these rules. For a given rule and conclusion, the system traces all observations that are related to the rule. These are observations that appear in the rule, or observations that lead to intermediate hypotheses that in turn appear in the rule. With numerical findings and intermediate hypotheses, there may be very large numbers of potential instantiations of a single rule.

From the implied set of all potential observations of a rule, observations are randomly generated until the rule is satisfied. Cases may be randomly generated for a given rule or hypothesis.

Rule Basher

Given that one has a correct knowledge base and a correct set of cases, the *rule basher* randomly modifies or *bash*es rules. This leads to incorrect rules, and the goal is to see how well the refinement systems does in return to the previously correct state. The current basher modified rules in a form consistent with potential refinements. Here is a partial list of the types of rules bashes that are performed:

- modify a numerical range (e.g. change age from 40 to 50)
- modify a confidence measure
- modify a choice number (e.g. choose 3 from a list of observations instead of choose 2.)
- add or delete a component of a rule

Preliminary Refinement Results

During the last quarter, we created a simulated case data base of 74 cases. Starting with a correct knowledge base and 74 simulated cases, the knowledge base was bashed with varying numbers of modifications. Preliminary results are listed in Figure 5-1. With a generalized knowledge base, not every bash will result in an erroneous conclusion. Multiple rules may cover the same situation, and some rules may never be invoked because no cases are found for that rule. Figure 5-1 lists the number of bashes, the number of cases correct after the rule bashing, the number of changes made by the refinement system, and the number of correct cases after automatic refinement is completed.

No. of Bashes	Correct Cases	Refinements	Refined Correct
1	74	-	
2	74	-	
4	74	-	
8	72	1	72
16	72	1	72
32	66	4	72
64	66	4	72
128	61	7	72
256	45	11	64

Figure 5-1: Early Results for Simulated Refinement Experiments

We expect to have more extensive experimental results during the next fiscal year (89).

6. Objectives for FY89

Here are our objectives for the next year, fiscal year 89:

- full demonstration of refinement system, using subset of DEC's Network Troubleshooting Consultant (NTC). System will automatically recover from many forms of damage to knowledge base.

- full demonstration of system with capabilities for automatic refinement, and verification of knowledge base consistency. Empirical experiments will be performed and results will be reported.

- demonstration of significant automated rule learning capabilities.

- demonstration of extended system capabilities for alternative control strategies and representations.

- completed comparative studies of empirical techniques for machine learning, statistical pattern recognition, and neural nets.

7. Financial Review

1. Basic contract dollar amount: $536,919 (9/1/87-8/31/89)

2. Dollar amounts and purposes of options: None

3. Total spending authority received to date: $475,000 through 1/31/89

4. Total spending to date: $177,094 through 8/31/88

5. Monthly expenditure rate: We have charged very little in salaries over the academic year in anticipation of funding larger amounts in summer salaries (when we are able to devote major efforts to the research project). We have funded a total of approximately $177,094. This would, therefore, result in an average monthly expenditure rate of $14,758. We do, however, expect the second year of the project to include higher expenditures (predominately in salaries) since we plan to bring on board one or two more graduate students to assist in this research.

6. Major non-salary expenditures planned within this increment of funding: None

7. Date next increment of funds is needed: January, 1989.

I. Technical Report

A technical report, *Maximizing the Predictive Value of Production Rules*, is enclosed with this annual report.

References

[Breiman, Friedman, Olshen, and Stone 84]
Breiman, L., Friedman, J., Olshen, R., and Stone, C.
Classification and Regression Tress.
Wadsworth, Monterrey, Ca., 1984.

[Efron 82] Efron, B.
The Jackknife,the Bootstrap and Other Resampling Plans.
In *SIAM*. Philadelphia, Pa., 1982.

[Ginsberg, Weiss, and Politakis 88]
Ginsberg, A., Weiss, S., and Politakis, P.
Automatic Knowledge Base Refinement for Classification Systems.
Artificial Intelligence :197-226, 1988.

[Lachenbruch and Mickey 68]
Lachenbruch, P. and Mickey, M.
Estimation of Error Rates in Discriminant Analysis.
Technometrics :1-111, 1968.

[Michalski, Mozetic, Hong, and Lavrac 86]
Michalski, R., Mozetic, I., Hong, J., and Lavrac, N.
The Multi-purpose Incremental Learning System AQ15 and its Testing Application to Three Medical Domains.
In *Proceedings of the Fifth Annual National Conference on Artificial Intelligence*, pages 1041-1045. Philadelphia, Pa., 1986.

[Quinlan 87] Quinlan, J.
Simplifying Decision Trees.
International Journal of Man-Machine Studies :in press, 1987.
also Tech. Report 87.4, New South Wales Intitute of T[illegible]ology, School of Computing Sciences.

[Stone 74] Stone, M.
Cross-Validatory Choice and Assessment of Statistical Predictions.
Journal of the Royal Statistical Society 36:111-147, 1974.

[Weiss 88a] Weiss, S., Galen, R., Tadepalli, P.
Maximizing the Predictive Value of Production Rules..
Technical Report ONR-K-0398, Rutgers University, Department of Computer Science, 1988.

[Weiss 88b] Weiss, S.
Minimizing Error Rates for Induced Production Rules..
Technical Report LCSR-TR-106, Rutgers University, Department of Computer Science, 1988.

[Weiss, Galen, and Tadepalli 87]
Weiss, S., Galen, R., and Tadepalli, P.
Optimizing the Predictive Value of Diagnostic Decision Rules.
In *Proceedings of the Sixth Annual National Conference on Artificial Intelligence.*
Seattle, Washington, 1987.
in press.