Evaluating CELIA: A study of the effect of case acquisition

Michael Redmond
Computer Science
Rutgers University
Camden, NJ 08102
(609) 225-6122
E-mail: redmond@crab.rutgers.edu

Abstract

CELIA [Redmond 1992] is a multi-strategy learner that, among other things, acquires cases through observing an expert. With all other learning turned off, CELIA improves its predictions of the expert's actions dramatically. However, performance does not monotonically increase with more cases since 1) some problems are harder than others, 2) an appropriate case still may not be available, 3) there is the possibility of retrieving the "wrong" cases. We report results from experiments with two different problem sets, both from the domain of automobile diagnosis. We discuss the variation in performance at different levels of experience in the two studies.

Introduction

CELIA [Redmond 1992] is a multi-strategy learner; it learns through observing and interacting with an expert and trying to explain the expert's actions to itself. One of the key learning strategies is acquiring cases; with all other learning (besides learning cases) turned off, CELIA still improves dramatically. In this paper, we discuss our evaluation of the case acquisition portion of CELIA.

It is not surprising that a Case-based reasoner improves as it acquires more cases; having more cases available gives a reasoner more experience to draw from. But performance does not monotonically increase with more cases. We have hypothesized that this may be true for at least three reasons:

1. Some problems may be harder than others.
2. Even as more cases are retained, there still may not be an appropriate case to help with a particular new problem.
3. There is the possibility of retrieving the "wrong" cases, which doesn't necessarily decrease with a larger case base.

We have investigated the improvement in CELIA's performance that can be traced simply to acquiring more cases (for instance, index learning is turned off; retrieval is by a nearest neighbor approach.). The general form of all experiments with CELIA is to measure performance by comparing CELIA's predictions of *each* of the expert's actions to the actions chosen by the expert on the same problems. The diagnosis is only one of the actions being predicted.

We report results from experiments with two problem sets, both from the domain of automobile diagnosis. With the first problem set, there was much more variation in performance than with the second problem set. In this paper, we look at the potential causes of the variation. First, we briefly discuss CELIA.

CELIA

For CELIA to get something out of observed problem solving, CELIA makes an active effort to understand. Then it can make use of what it understands in its later problem solving. Learning is even more effective since CELIA sets up expectations of what the instructor will do. When CELIA's expectations fail, this failure indicates that it must learn something. The understanding process is broken down into three subprocesses:

1. Predict - CELIA predicts the instructor's next reasoning goal, and how it will be carried out.
2. Observe - CELIA observes the expert's actions, comparing to the prediction.
3. Explain - CELIA explains the expert's actions to itself.

Through this process, CELIA comes to understand the expert's problem solving — what the goals are, what goals follow from each other, etc. (CELIA has many other aspects, presented in Redmond[1992].)

Figure 1 shows part of a sequence of the expert's actions in solving an example problem. CELIA retains such an example as a case, though explaining the actions to itself.

Experimental Methods

In evaluating CELIA's learning, we must decide what constitutes the "performance" to be measured. CELIA must do more than categorize the problem into a fault. It needs to be able to take the actions that will gather

Diagnosis Actions (in order presented)

...
11. Hyp - Loose Connected Spark Plug
12. Test - Connected Spark Plug (Pos.)
13. Interpret - Rule Out Loose Spark Plugs
14. Hyp - Malfunction Carburetor
15. Hyp - Lean Idle Mixture
16. Hyp - High Float Level
17. Test - Lean Idle Mixture (Neg.)
18. Interpret - Rule Out Lean Idle Mixture
19. Test - High Float Level (Neg.)
20. Interpret - Rule Out High Float Level
21. Hyp - Malfunction Control System

...

Figure 1: Some steps in a diagnosis.

the relevant information, take repair actions, iteratively refine problems, etc. The easiest way to measure how well CELIA does is in comparison to the actions chosen by the expert on the same problems. Since in our model CELIA is predicting each of the instructor's actions before the instructor makes them, and since it predicts the expert will do what it would do in the same situation, the correctness of CELIA's predictions is a good measure of CELIA's ability.

In the general experimental approach, CELIA is presented a sequence of examples of expert problem solving. The performance measure is the accuracy of the system's predictions of the expert's actions.

Many of the experiments were performed with two different problem sets, both from the domain of automobile diagnosis. One set contained 24 examples, and the other, 48 examples. In the first problem set, there are eight distinct faults, mostly in the fuel system, but also in the electrical system. In the second problem set, there are three distinct faults, all in the fuel system, but 12 different distinct paths to the solution.

In the experiments, ten random orders of the examples from a problem set were presented. The graphs showing performance by position in this paper display the average correct percentage of predictions by amount of experience. Thus, for example, a point on a graph corresponding to 12 on the x-axis shows the average accuracy for predicting the steps in the 12th problem in the sequence of examples (over 10 different orderings).

Taking the measurement during learning (instead of with a separate test set after learning) does not pose a problem since the same problems do not reoccur. This also enables getting a complete learning curve without running an impractical number of runs. Obtaining 10 data points for each of 24 positions in a sequence with measurement during learning requires 10 runs instead of the 240 required with separate test sets.

Overall Evaluation of CELIA and Acquiring Cases

The first evaluation of CELIA involved looking at the program as a whole. We carried out an initial experiment with CELIA in the general manner described above. We expected that through the process of observing an expert, the system would significantly improve its ability to predict the expert's actions. [1] To evaluate that prediction, CELIA was presented ten random sequences of 24 examples from the first problem set. CELIA showed dramatic improvement in its prediction of the expert's actions [Redmond 1992].

(a)

(b)

Accuracy for CELIA predicting expert problem solving actions. All learning except acquiring new cases has been removed. (a) shows data for the first problem set; (b) shows data for the second problem set.

Figure 2: CELIA: Improvement through acquiring new cases.

While empirical evaluation can suggest things to look into, it raises as many questions as it answers. In order to progress, we need to explain why the performance characteristics are the way that they are. We undertook an ablation study to determine the relative effects of the learning processes. In one test, all learning except acquiring cases was turned off and the same experiment as discussed above was carried out and the

[1] The improvement comes through acquisition of new cases, and other methods described in Redmond [1992], including learning indices and censors, and through adjusting feature salience in the retrieval function. Other methods of learning requiring spontaneous interaction with the expert were turned off for the entire set of experiments.

(a)

(b)

Figure 3: CELIA: All data points.

relative performances were compared. The conclusion is that over this experience range, the greatest amount of improvement comes through acquiring cases to use [Redmond 1992].

CELIA's improvement over the course of exposure to examples (with only case acquisition as a learning method) is presented in Figure 2. Figure 2(a) shows the results with the first problem set and Figure 2(b) shows the results of the same experiment with the second problem set. The graphs show the average correct percentage of predictions by amount of experience. These are predictions of the steps to be taken, not just classifications of problems. As can be seen, CELIA dramatically improves its performance after only a few examples have been seen.

It should be noted that one problem set is distinctly easier than the other. Since the faults in the second problem set are all in the same subsystem, many of the early actions are taken by the expert in all problems, making them possible to predict even if a problem with the given fault had not been previously encountered.

However, it was not the case in either experiment that the reasoner steadily improved without setbacks. Figure 3 shows all data points in the two experiments to illustrate the range of variation of performance during learning. Once again, we need to explain these performance characteristics.

The labels actually represent a class of problems, not a single problem. For example, p23 includes problems 23, 33, and 43.

Figure 4: CELIA: Average Performance by Problem Type.

Variance in Performance

CELIA's performance is not monotonically increasing. There is also a great deal of variation in performance for any given experience level, as can be seen in Figure 3. Two explanations suggest themselves. First some problems being observed are more difficult than others. Of the 24 problems in the first problem set, there are three problems involving a fault in the electrical system. These share little in common with the other problems, which involve faults in the fuel system and the carburetor. Thus there is little transfer from the other problems. Figure 4 shows CELIA's average predictive accuracy for each of the types of problems in the first problem set. While there is little variation in difficulty of the problems in the second set, in the first set, the class of problems designated a-p#7 are decidedly more difficult. If one of the hard problems falls disproportionately in particular positions in the example sequences they will bring down the average performance for that experience level. For example, one of these three electrical system problems falls in positions six and seven more often than in positions five and eight. This helps lead to the dip in average performance in positions six and seven despite greater experience.

The second factor is differences in the number of similar problems seen at each point. This ties to two of the factors mentioned in the introduction. As noted, the examples in the first set include three different problems for each of the eight different faults. In the second problem set, the examples include four different problems for each of four different solution paths for each of three distinct faults. We would expect that it would be easier to predict the instructor's actions when CELIA has observed more similar problems. If no previous similar problems have been observed then there may not be an appropriate case to retrieve. Secondly, the more previous similar problems seen, we expect it to

(a)

(b)

Figure 5: CELIA: Average Performance by Number of Previous Similar Problems Seen.

(a)

(b)

Figure 6: CELIA: Performance when zero, one, two, and three previous experiences are available.

be (other things being equal) more likely that the "correct" case will be retrieved. In fact this is true. Figure 5 shows the average performance for times when zero, one, two, and three (for the second problem set) previous similar problems have been seen. The biggest advantage comes when CELIA acquires one case of a type. In fact, with the second problem set, performance is so good with one previous similar example that it is hard to make further gains.

Figure 6, shows the effect of this difference on CELIA's performance curve. It shows the performance when CELIA has seen zero, one, two, and three previous similar problems. The difference is noticeable with the first problem set but is not with the second problem set. Closer analysis of the performance data for the first problem set indicate that *for a given fault and position in the learning sequence*, CELIA almost always does a better job predicting when it has seen more similar problems (35 better, 4 equal, 4 worse).[2] This helps explain the fluctuations in performance. If the problems involving one fault all fall towards the end of the learning sequence, CELIA will not do a good job of predicting the expert's actions on the first one even though it is relatively late in the sequence. Alternatively, if all of the problems involving one fault fall towards the beginning of the sequence, CELIA will exhibit good performance on those problems even though it is early in the sequence. This will increase the average performance in those positions. This situation occurs in our random sequences. For example, in three of the random orders for the first problem set the first problem of one kind showed up in the 14th position. This corresponds to a dip in the performance curve.

Figure 7 shows the average number of previous similar problems seen by position for the first problem set. Note that the dips in average performance in the 6th and 14th positions (best seen in Figure 2) correspond to dips in average number of previous similar problems seen. The jump in average performance in the 5th position corresponds to a jump in the average number of previous similar problems seen. This is certainly not an invariant relationship; some positions do not experience a dip that would be predicted from this one factor. There also does not appear to be such an effect with the second problem set; performance gets very good quickly, without a lot of previous similar examples.

Since the number of previous problems seen has been controlled for, it might be expected that each of the curves in Figure 6 would be monotonically increasing.

[2] Even with the second problem set, controlling for fault and position in learning sequence, CELIA frequently does a better job predicting when it has seen more similar problems (41 better, 28 equal, 19 worse).

Figure 7: CELIA: Average Number of Previous Similar Problems Seen by number of previous experiences.

However, that was not the case. Besides the differences in problem difficulty, there are other reasons as well. First, consider the situation for when CELIA has not seen any previous similar problems. Problems with different root faults can provide some transfer, and thus allow the performance curve to initially rise. However, they also provide interference, and they cannot provide all the right actions. Thus, after a while performance starts to taper off. Next, consider the situation when CELIA has seen previous similar problems. When faults show up several times early in learning, CELIA doesn't face much interference. As was noted, this occurs in our random sequences. The cases that can provide useful guidance are easily accessed. As more examples are seen, there is some drop off as there is more potential for incorrect actions to be suggested. Thus, the curves for one and two previous similar problems start off very high, and then show some decrease. We expect that performance should rise again later due to more appropriate access, if indices and censors are learned.

In sum, we have seen that the variance in performance is a result of differences in the problems and in the presentation order of the problems.

Related Work

By retaining examples as cases, a student can carry out early learning in a domain despite lack of a strong domain model. Bareiss's [1989] PROTOS learns knowledge for classification through apprenticeship. PROTOS attempts to classify a problem and as a result of feedback, retains some examples as cases and also some domain knowledge. However, the assumption that classification is the task limits PROTOS. For example, PROTOS would require extension before it could determine that it requires more information to solve a problem. Our model shows how an apprentice can learn a more complicated and flexible problem solving procedure, making it possible to solve problems that require more than classifying the problem.

Golding's [1991] Rational Reconstruction (RR) is another related method. Given a problem and an answer, RR attempts to explain the answer. Through this process, it acquires a new case. The major difference from our process is in level of explanation. RR explains a classification; in our model the learner explains each goal in a sequence. In RR, the relationships between actions is not explained, so the learner does not gain as complete an understanding of the case.

Conclusion

We have evaluated our model in a number of ways. The program, CELIA, demonstrates that the general approach to learning leads to dramatic increases in performance after only limited exposure to examples. This improvement is mainly due to the acquisition of cases that reflect the problem solving done in the examples. In this paper, we have attempted to explain some of the variations in performance detected in the experiments. Performance during learning is affected by the difficulty of the problem, and the number of previous similar problems already seen.

Acknowledgement

Portions of this research were supported by the Army Research Institute for the Behavioral and Social Sciences under Contracts MDA-903-86-C-173, and MDA-903-90-K-0112 and by DARPA contract F49620-88-C-0058 monitored by AFOSR. Thanks to Joel Martin, Steve Robinson and Janet Kolodner for advice.

References

Bareiss, R. 1989. Exemplar-based knowledge acquisition: a unified approach to concept representation, classification, and learning. New York, NY: Academic Press.

Golding, A. R. 1991. *Pronouncing Names by a Combination of Case-Based and Rule-Based Reasoning.* Ph.D. Dissertation, Stanford University, Palo Alto, CA.

Kolodner., J. 1993. Case-based reasoning. Los Altos, CA: Morgan Kaufmann.

Redmond, M. 1990. Distributed cases for case-based reasoning; facilitating use of multiple cases. In *Proceedings of the National Conference on Artificial Intelligence (AAAI-90).* Boston, MA: Morgan Kaufmann.

Redmond, M. A. 1992. *Learning by Observing and Explaining Expert Problem Solving.* Ph.D. Dissertation, Georgia Institute of Technology, Atlanta, GA.