Encouraging Self-Explanation through Case-Based Tutoring, A Case Study [1]

Michael Redmond
Computer Science
Rutgers University
Camden, NJ 08102
(609) 225-6122 Fax: (609) 225-6624
E-mail: redmond@pizza.rutgers.edu

Susan Phillips
Chemistry
Holy Family College
Philadelphia, PA 19114
(215) 637-7700
E-mail: phillips@pizza.rutgers.edu
Keywords: case-based education, self-explanation

Abstract. This paper presents a case-based tutor, CECELIA 1.1, that is based on techniques from CELIA, a computer model of case-based apprenticeship learning [Redmond 1992]. The teaching techniques include: interactive, step by step presentation of case solution steps, student predictions of an expert's actions, presentation of the expert's steps, student explanations of the expert's actions, and presentation of the expert's explanation. In addition, CECELIA takes advantage of a technique from VanLehn's [1987] SIERRA – presenting examples in an order so that solutions only differ by one branch, or *disjunct*, from previously presented examples. CECELIA relies on its teaching strategy encouraging greater processing of the examples by the student, rather than on embedding great amounts of intelligence in the tutor. CECELIA is implemented using Hypercard on an Apple Macintosh, and has been pilot tested with real students. The tests suggest that the approach can be helpful, but also suggest that eliciting self-explanations from students who normally do not self-explain may be challenging.

[1] This research follows from work on CELIA which was supported by the Army Research Institute for the Behavioral and Social Sciences under Contract No. MDA-903-86-C-173, and Contract No. MDA-903-90-K-0112 and by DARPA contract F49620-88-C-0058 monitored by AFOSR. Work on CECELIA was partially supported by Rutgers University and Holy Family College. Thanks to Janet Kolodner for her advice and guidance concerning CELIA, and to Justin Peterson and Joel Martin for helpful discussion. Colleen Griffin did some of the development of the initial version of CECELIA.

1 Introduction

Research dealing with models of learning can be examined from the opposite direction – implications for teaching. Redmond [1992,1990] presented a case-based computer model, CELIA, that shows how a student can learn by observing an expert and explaining the expert's actions to himself. CELIA was inspired and influenced by protocol studies [Lancaster and Kolodner 1988]; empirical experiments with the program suggested that the model was an effective learning method [Redmond 1992]. The success of CELIA suggested several educational implications [Redmond 1994]. We have developed a case-based tutor, CECELIA (Chemistry Education CELIA) based on some of the learning techniques that made CELIA successful, along with another tactic. The framework used in CECELIA is an iterative, presentation of the steps in the correct solution of a case. There are four central features of the teaching strategy:

- student predictions of an expert's actions. This helps a student to identify when they need to learn something.
- student (self-)explanations of the expert's actions. Self-explanation [Chi, Bassok, Lewis, Reimann and Glaser 1989; Chi, de Leeuw, Chiu, and LaVancher 1994; Renkl 1996] involves the student trying to explain how the expert's current action fits into the overall problem solving. This helps both in identifying the need to learn and in more fully understanding the expert's problem solving.
- focus on step by step (iterative) problem solving. This helps both by simplifying credit and blame assignment and by illustrating procedural knowledge.
- problem sets that isolate one *disjunct* (or branch) at a time [VanLehn 1987]. This helps the learner identify the important difference from known solution methods.

It should be noted that CECELIA in *not* an *intelligent* tutoring system (ITS) [Wegner 1987; Farr and Psotka 1992]; it is a tutoring system that uses a teaching strategy based on successful case-based learning research. ITSs know how to solve the problems being presented, know about common errors, follow the student's problem-solving, and diagnose the student's failures. In many cases, even the choice of problems presented to the student is determined by the tutor's assessment of what the student knows. CECELIA does not attempt to model what the student knows; it does not try to debug what the student is doing wrong; it does not adapt the problems to the student; and it does not know how to solve the problems.

While a full intelligent tutoring system is a potentially powerful teaching tool, the investment of effort to produce these systems is a significant obstacle, especially since much of the effort involves domain-specific and especially task-specific engineering. In fact, building ITSs subsumes the task of artificial intelligence; being able to solve problems in a cognitively plausible manner is just the beginning. This means that for each new task to be tutored, person-months of (sophisticated) effort is required to develop the tutor.

While CECELIA cannot present personalized instruction, it is built under the hypothesis that the greater initiative CECELIA asks of the student will lead to successful learning outcomes. Other recent research efforts have come to the same conclusion. Nathan and Resnick [1995] have shown that robust learning can occur without the tutor "understanding" students' actions. Their system instead promotes active learning and engages the student in generating and evaluating possible solutions.

CECELIA is implemented using Hypercard on a Macintosh, since Hypercard provides good support for developing a user friendly interface, and this environment is highly available, making future educational use practical. The current version focuses on teaching "balancing equations", a task involving choosing the correct coefficients for a chemical equation. This was chosen since it is the first task requiring multi-step problem-solving in general chemistry. In initial pilot testing, while some students' learning gains were encouraging, many students did not show signs of doing self-explanation and did not benefit from the tutor.

We first discuss the teaching strategy sketched above, then briefly discuss the balancing equations task. Then we show how CECELIA implements the teaching strategy, and discuss the results of initial pilot testing.

2 Features of the teaching strategy

When presented with a case, prediction of an expert's actions helps a student identify their weaknesses and gaps in their knowledge. When a student incorrectly predicts the expert's actions, that shows the student that his own problem-solving would have been incorrect for that step in the case. Prediction is useful even when the student is doing self-explanation. It takes greater understanding to actually come up with the correct action on one's own than to understand why an observed action was done. Prediction gives the student a test of their procedural knowledge.

Self-explanation can help a student detect a lack of knowledge or help the student better understand the expert's actions. Chi *et al* [1989] found that good students identified a lack of knowledge *more frequently* than did poor students, who remained unaware of what they didn't know. In addition, Tarmizi and Sweller [1988] found that studying worked out examples can be an even more successful learning technique than actually solving problems. Prompting the student to do self-explanation increases the possibility that the student will recognize the gaps in their knowledge, and try to learn something. Also, *successful* explanation of the expert's actions allows the student to come to a better understanding of the expert's problem solving, and may also help the student remember this example for future use. From a case-based perspective, the case that is more thoroughly explained to themself by the student is more useful in the future since it is a more fully elaborated memory, and more likely to be retrieved at an appropriate time since it is more likely to be appropriately indexed.

A focus on step by step, iterative, problem-solving provides advantages over approaches that focus on only problems ("given" information) and solutions. One, is that it illustrates more detailed, procedural expertise. Showing the process of finding a solution in a step by step manner, instead of simply showing a solution, provides a student correct steps to a solution, which he can use to produce future solutions. The cases thus contain the process necessary to get the solution instead of just the solution. From a case-based perspective, this gives the student exposure to the snippets [Kolodner 1988; Redmond 1990] that make up the details of the case. Second, perhaps more importantly, when student predictions are made on a step by step basis, credit and blame assignment is simplified. Incorrect predictions of *steps* (rather than whole solutions) isolates the problem to that step. Anderson and his group's intelligent tutors [Anderson, Boyle, Corbett, and Lewis 1990; Koedinger and Anderson 1990] make use of such a step by step focus as part of their model-tracing approach – giving feedback after each step taken by the student. Although not without controversy, they have found that it speeds the learning process. Our approach also provides the step by step focus but initially leaves it to the student to try to understand why the expert would take the step. From a case-based perspective, this step-by-step focus, in conjunction with self-explanation, could help students to retain justifications with the steps in the case [Carbonell 1986; Carbonell and Veloso 1988], and/or develop better indices.

The final aspect of CECELIA's teaching strategy is VanLehn's [1987] "one disjunct at a time" convention. Examples are presented in an order such that the solutions only differ from the procedure that the student knows by one difference. This teaching strategy actually has a long history in artificial intelligence, going back to Winston's induction program [Winston 1970], and it works with people too; VanLehn [1983] found that human teachers frequently implicitly follow this constraint though they aren't aware of the convention. CELIA did not benefit from this teaching strategy because of differences in domain. However, since CECELIA operates in a domain in which a wide variety of examples can be generated on demand, CECELIA takes advantage of this teaching strategy. From a case-based perspective, this strategy could facilitate storing difference links [Bareiss 1989] with case steps.

3 Balancing Equations

Chemical equations describe reactions, showing what elements or compounds react, and what products are formed during the reaction. The following equation is not balanced:

```
H O   ----->  H    +  O
 2            2        2
```

Integer *coefficients* are placed in front of compounds to show (relatively) how many atoms of each are involved in the reaction. Coefficients are used to *balance* the equation, so that the same number of each element appear on both sides of the equation. Chemical equations must always be balanced if they are to represent a real reaction. The above equation is not yet balanced since the left side has one oxygen atom, but the right has two.

How can this be corrected? A coefficient of "2" in front of the water molecule makes the oxygen atoms equal, but this causes the hydrogen atoms to be unequal, 4 on the left side and 2 on the right side. If a coefficient of "2" is put in front of the H 2 , the equation will be balanced:

$$2\,H_2O \;\text{----->}\; 2\,H_2 + O_2$$

The initial simple general method taught by CECELIA in initial lessons is:

1. Find an element that only occurs once on each side of the equation, and choose coefficients so that the element will have equal numbers of atoms on both sides.
2. From the remaining compounds without coefficients, find an element that occurs in only one. That compound's coefficient is now fully constrained; calculate the value.
3. Repeat step two until all coefficients have been assigned.

4 CECELIA

CECELIA presents cases to the student in groups, called *lessons* following VanLehn [1987]. The examples within each lesson all are solved using the same steps; the only differences are surface features (such as which particular elements are involved, and what the subscripts are). Each lesson contains two cases in order for the student to be able to isolate and understand the extension to the previously understood process (the new disjunct). Note that because of this strategy, CECELIA does not attempt to choose which cases to present to the student at which time; the order is pre-determined. For teaching the balancing equations task, seven lessons are sufficient.

Each case is presented in a simple iterative manner: for each step in the solution, four screens (*cards* in Hypercard) are presented in order:

1. a prompt for the student to predict the next action
2. presentation of the correct action (with an option to see other acceptable alternatives, if any). The student is then asked to self-grade their answer.
3. a prompt for the student to explain why the correct action is appropriate
4. presentation of an expert explanation of the correct action.

Figure 1 shows these types of cards for the start of a simple case. Figure 1(a) shows the prompt asking the student for a prediction; predictions which involve choosing coefficients can be made by filling the numbers directly in the equation. Figure 1(b) shows CECELIA's presentation of the correct action. If an alternative action is possible, the student can click to see the alternative. So that CECELIA doesn't have to either understand natural language or force the student into some pre-set language for responding, the student is asked to mark the correctness of his prediction by clicking on a button. The student's predictions are captured in a file so the student's marking can be checked later if necessary.

Next, the student is asked for an explanation of the expert's action. This prompt is skipped if the student has made greater than N predictions with greater than X percent correct, and has marked their current prediction as correct. N and X are adjustable parameters, currently set to 10 and 80. Figure 1(c) shows the prompt asking the student for an explanation. The student can choose to give their explanation by 1) typing into the field on the screen, or 2) by clicking on boxed items at the top of the screen (such as "CH4", "the left side",

(a) (b)

(c) (d)

Fig. 1. CECELIA 1.1: Example student interaction.

etc), in conjunction with typing, or 3) by talking into a tape recorder (no students in the pilot test chose to tape record explanations). This explanation is also captured into a file for further analysis. Note, that as problem solving in an example continues, the student can see the previous steps taken in the problem. Thus, short term memory is not an obstacle to prediction or self-explanation. This reduces the cognitive load on the student, making it more likely the student can learn by studying the example [Tarmizi and Sweller 1988].

Finally, the student is given the explanation of the expert's action (Figure 1(d)). This explanation is detailed, showing reasoning behind which coefficients were filled in, and the algebra behind the numbers chosen (pilot testing with CECELIA 1.0 showed that the algebraic manipulations are an important step to the pilot audience).

Figure 2 illustrates some of the disjuncts involved in teaching the balancing equations task by showing the self-explanation card for the final steps of examples

(a) (b)

(c) (d)

Example Hypercard cards showing the different paths, or disjuncts, that problem solving can take.

Fig. 2. CECELIA 1.1: Examples of solutions to problems presented to student.

from each of the early lessons. The first case (Figure 2(a)) solution has no complications. In the second lesson (Figure 2(b)), in the first step, instead of just choosing "1" as the coefficient (done in (a) since the element had the same subscript on each side), different coefficients must be chosen on the two sides since carbon (C) has different subscripts in the two terms.

A more significant increase in complexity is encountered in the third lesson (Figure 2(c)). Here the disjunct arises in the 2nd step – the initial coefficient choice for the terms involving carbon (C) leads to later calculation of a coefficient of 1/2. Since this is not valid, all coefficients are doubled to get all whole number coefficients. The fourth lesson (example in Figure 2(d)) teaches that the two preceding disjuncts can occur together in the same problem. In step 1, different coefficients must be chosen on the two sides; in step 3, all coefficients must be doubled. The fifth lesson involves doubling coefficients twice. Further lessons introduce greater complexity, eventually leading to the most difficult problems where backtracking is required.

5 Pilot Testing

We have carried out pilot testing of CECELIA to obtain initial feedback on the effectiveness of the tutor, and to get qualitative feedback from students. The hypothesis was that students could gain significantly with a tutor using the teaching strategies presented above, without putting intelligence into the tutor. Due to the small number of subjects, and some ceiling effect, we cannot draw statistical conclusions about CECELIA's effectiveness, but several students demonstrated performance gains, as well as increased confidence in their abilities.

5.1 Method

Subjects were students of Introductory Chemistry at Holy Family College. The subjects were recruited after the material (balancing equations) had been completely presented in class; they participated in exchange for extra-credit. The students were given a pre-pre-test; this was used to assign students of relatively equal distribution of prior abilities to the control and test groups. During the next week they had a second session in which they took a second pre-test, used the tutor, and then took a post-test. Though this immediate post-test does not give the student a chance to reflect on and integrate their experience with the tutor, it was done due to previous reviewer concerns about a Hawthorne effect. The subjects were also given a post experiment questionnaire to obtain subjective feedback on the tutor. A total of 13 students participated (a previous pilot study with CECELIA 1.0 involved 11 students).

Pre and post-tests were given on paper. Each test included twelve questions, which were counter-balanced – several versions of the test were constructed so that questions appeared equally on each of the three tests for each of the control and test groups. The students were given 30 minutes to do the pre-pre, pre and post-tests. All of the students completed these tests. Partial credit (1/3 correct) was given to students for problems in which the coefficients balanced the equation but were not the lowest possible values. The tutor's problems were a separate pool, not appearing on any of the paper tests. The control students used a standard practice or drill tutor on the same problems as used in CECELIA. The students were scheduled for 1 hour to use the tutor; this was held constant to ensure equal time-on-task for test and control groups. If students finished before an hour, they were re-presented the same problems again (this happened with several students). If an hour was up before the students were done, they were cut-off (this happened with most of the self-explanation tutor test students).

5.2 Results

The performance improvements of subjects with CECELIA was only 4 percent (due partially to one subject who had a decline, and also partially due to 4

Student	Pct Correct Pre	Post	Diff
SA	55.0	59.2	4.2
SB	83.3	91.7	8.4
SC	94.2	94.2	0.0
SD	75.0	66.7	-8.3
SE	77.5	83.3	5.8
SF	31.7	46.7	15.0
Means	69.44	73.61	4.167

$s_D = \mathrm{sqrt}(s_D) = 7.88811$

$t_{.025} = 2.571$ (5 degrees of freedom)

Difference $= 4.167 \pm (2.571 * 7.888) / 2.4495$

$= 4.167 \pm 8.279407$

Table 1. CECELIA 1.1: Self Explanation Tutor Results from Pilot Study.

other subjects who did well on the pre-test (75 percent or above)). This was not statistically significant (statistically discernable) at the 0.05 level, and in fact was similar to the improvement for those using the control tutor. This was actually a smaller improvement than found in the pilot test with CELIA 1.0 (a 20 percent gain, under different experiemental conditions, *e.g.* students completed the tutor problems regardless of the time it took).

5.3 Discussion

There are several things that can be learned from the limited success of this prototype. One, the self-explanation literature identified a strategy that was used by the students who learned more (the "good" students) [Chi *et al* 1989]. In this experiment, we were working with students who may not normally use these self-explanation techniques. In [Chi *et al* 1994], students, whether they normally self-explain or not, when prompted to self-explain, were able to get benefits from that self-explanation. In the current tutoring effort, the program asked the students for self-explanations, however, it did not have a way of checking that explanation was really done. In fact, studying the logs captured by the program, frequently the students had little or no explanation (for example, a single space character). Thus, this tutor was not necessarily eliciting self-explanations from subjects who were not prone to do so otherwise. It is possible that the typing modality was too slow for the subjects. The opportunity to respond verbally for tape recording was offered and not taken. This may have been due to self-consciousness – the room where subjects worked had 3-5 subjects working at the same time. The point and click method of building explanations may have not been clear; the subjects only had one screen of introduction to the technique. The self-explanation literature was developed with verbal protocol studies, where a human experimenter prompted a subject to speak, and perhaps indirectly to think. It may be that a computer tutor may not be as compelling a prompt for promoting self-explanation. Some solution may be necessary for this problem.

Chi *et al* [1989] also found that one of the benefits that good students obtained by self-explaining was a better recognition of when they needed to learn something (they had more incidents when they identified that they didn't understand something). In the current experiment there were many situations where students who self-graded their predictions marked their answer incorrectly (usually marking "correct" when they were actually incorrect). These subjects were not getting an accurate idea of their skills.

It is possible that the subjects were pursuing goals of completing the extra-credit task, rather than learning goals. They, in fact, were not told that they would spend an hour with the tutor no matter how far they got. Some students appeared to try to race through the tutor as fast as possible.

Another problem, at least as far as getting noticeable improvement, was the level of difficulty of the task. This task is learned early in the first semester of college chemistry; almost all of the students had had some high school chemistry, where they most likely had been previously exposed to the task. Between that, and that the tutor was used after the material was presented in class, some students already could solve many of the problems. This, in part, explains the ceiling effect in which many students were already fairly successful at pre-test time. In fact, one student had a clear algorithmic method from high school that differed from the method demonstrated by the expert cases (and scored 94 percent correct on the pre-test). Future harder tasks being considered include Stoichiometry from chemistry (also covered in high school, but still challenging at the college level), and teaching the use of PERT charts to MBA students.

It is worth discussing the changes from CECELIA 1.0, and their impact. These changes were largely made in response to qualitative survey responses ro CECELIA 1.0. The extra choices for self-explanation modality were added; the presentation of expert explanations was added, the cases were divided into steps in a courser granularity - frequently two to three steps in CECELIA 1.0 became 1 step in CECELIA 1.1. This coincided better with the way the students perceived a step. However, it tended to gloss over the details of how to perform the steps – these substeps only appeared in the presentation of the experts explanation of a step. Thus, the students may not have learned the heart of the process implicit in the expert's case solutions. This could have been a factor in students achieving less of a gain with CECELIA 1.1 than with CECELIA 1.0. An improvement in strategy in the tutor, however, could be to initially show very detailed steps, and later *fade* the detail into fewer steps.

An additional comment should be made. The textbook introduces the balancing equations task as a heuristic task (not in those terms, of course). However, the tutor, in presenting examples one disjunct at a time, is actually teaching an algorithmic approach. In fact, only one in 50 problems in the chapter required backtracking when following the tutor's method for solving the problems. Some students found this systematicity very helpful, since they did not have to rely on intuition. CECELIA seemed to be most helpful to those students with weak skills but strong motivation. For instance, some older students, and students with math anxiety, were able to not only improve their ability to balance equations, but were also able to increase their self-confidence, something that would carry over to later material.

6 Conclusion

We have presented the ideas behind the initial version of CECELIA, a case-based tutoring system that relies on a cognitive teaching strategy based on the results of learning research. CECELIA relies on encouraging greater processing of cases by the student, rather than on embedding great amounts of intelligence in the tutor. Thus, CECELIA trades off student initiative for a significant amount of domain and task-specific development effort.

The teaching techniques presented here can be applied in tutorials for any task in which

- solutions are multi-step procedures for which the steps can be identified by an expert
- cases can be obtained that differ from each other by a single new branch.

The development involved in teaching a new task consists only producing case "data" for CECELIA's program. This data includes: steps in getting the answer, explanations, and alternative steps. CECELIA 1.1 can almost be considered a "shell' for case-based self-explanation tutors.

CECELIA has been pilot tested with limited success. Some students showed very important gains, but many students did not. Future work will focus on 1) more difficult tasks; 2) greater motivation of students to self-explain; and 3) more effective monitoring of student progress (without trying to produce an intelligent tutor).

References

[Anderson *et al.*, 1990] Anderson, J. R., Boyle, C. F., Corbett, A., & Lewis, M. W. (1990). Cognitive modeling and intelligent tutoring. *Artificial Intelligence*, 42:7–49.

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

[Carbonell, 1986] Carbonell, J. (1986). Derivational analogy: A theory of reconstructive problem solving and expertise acquisition. In Michalski, R., Carbonell, J., & Mitchell, T., editors, *Machine Learning: An Artificial Intelligence Approach, Volume II*. Morgan Kaufmann, Los Altos, CA.

[Carbonell & Veloso, 1988] Carbonell, J. & Veloso, M. (1988). Integrating derivational analogy into a general problem solving architecture. In *Proceedings of a Workshop on Case-Based Reasoning*, Clearwater, FL. Morgan Kaufmann.

[Chi *et al.*, 1989] Chi, M., Bassok, M., Lewis, M., Reimann, P., & Glaser, R. (1989). Self-explanations: How students study and use examples to solve problems. *Cognitive Science*, 13:145–182.

[Chi *et al.*, 1994] Chi, M., de Leeuw, N., Chiu, M., & LaVancher, C. (1994). Eliciting self-explanations improves understanding. *Cognitive Science*, 18:439–477.

[Corbett & Anderson, 1990] Corbett, A. T. & Anderson, J. R. (1990). The effect of feedback control on learning to program with the lisp tutor. In *Proceedings of the Twelfth Annual Conference of the Cognitive Science Society*, pages 796–806, Cambridge, MA. Lawrence Erlbaum Associates.

[Farr & Psotka, 1992] Farr, M. J. & Psotka, J. (1992). Intelligent instruction by computer: Theory and practice. Taylor and Francis, New York, NY.

[Koedinger & Anderson, 1990] Koedinger, K. R. & Anderson, J. R. (1990). Abstract planning and perceptual chunks: elements of expertise in geometry. *Cognitive Science*, 14:511–550.

[Kolodner, 1988] Kolodner, J. (1988). Retrieving events from a case memory: a parallel implementation. In *Proceedings of a Workshop on Case-Based Reasoning*, Clearwater, FL. Morgan Kaufmann.

[Lancaster & Kolodner, 1988] Lancaster, J. & Kolodner, J. (1988). Varieties of learning from problem solving experience. In *Proceedings of the Tenth Annual Conference of the Cognitive Science Society*, Montreal, Canada. Lawrence Erlbaum Associates.

[Nathan & Resnick, 1995] Nathan, M. J. & Resnick, L. B. (1995). Less can be more: Unintelligent tutoring based on psychological theories and experimentation. In Vosniadou, S., Corte, E. D., Glaser, R., & Mandl, H., editors, *Psychological and Educational Foundations of Technology-Based Learning Environments.*

[Redmond, 1990] 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, 1992] Redmond, M. (1992). *Learning by Observing and Explaining Expert Problem Solving.* PhD thesis, Georgia Institute of Technology, Atlanta, GA.

[Redmond, 1994] Redmond, M. (1994). Educational implications of celia: Learning by observing and explaining. In *Proceedings of the Sixteenth Annual Conference of the Cognitive Science Society*, Atlanta, GA. Lawrence Erlbaum Associates.

[Renkl, 1996] Renkl, A. (1996). Learning from worked out examples: A study of individual differences. *Cognitive Science*, 20.

[Tarmizi & Sweller, 1988] Tarmizi, R. A. & Sweller, J. (1988). Guidance during mathematical problem solving. *Journal of Educational Psychology*, 80(4):424–436.

[VanLehn, 1983] VanLehn, K. (1983). Felicity conditions for human skill acquisition: validating an ai-based theory. Technical Report CIS-21, Xerox Parc, Palo Alto, CA.

[VanLehn, 1987] VanLehn, K. (1987). Learning one subprocedure per lesson. *Artificial Intelligence*, 31:1–40.

[Wegner, 1987] Wegner, E. (1987). Artificial intelligence and tutoring systems. Morgan Kaufmann, Los Altos, CA.

[Winston, 1970] Winston, P. (1970). *Learning Structural Descriptions from Examples.* PhD thesis, Massachussetts Institute of Technology, Cambridge, MA.