Why Example Fading Works: A Qualitative Analysis Using Cascade

Eric S. Fleischman (esfleisc@colby.edu)
Randolph M. Jones*[*] (rjones@colby.edu)
Department of Computer Science, Colby College, 5830 Mayflower Hill Drive
Waterville, ME 04901–8858 USA

Abstract

"Faded" examples are example problems that provide a solution, but first require students to generate a portion of the solution themselves. Empirical studies have shown that such examples can be more effective teaching aids than completely worked examples that require no work from the student. Cascade is a model of problem-solving skill acquisition that was originally developed to explain other empirical regularities associated with human problem solving and learning, most notably the self-explanation effect. Past research demonstrated that Cascade might also explain the mechanisms underlying the effectiveness of example fading. This paper analyzes new protocol data, and finds that it is consistent with predictions derived from Cascade.

Overview

Renkl, Atkinson, and Maier (2000) empirically demonstrated the qualitative result that, when learning problem-solving skills, students studying a series of "faded" examples show improved post-test performance over students studying only completely worked examples. Jones and Fleischman (2001) argue that this result can be explained by Cascade (VanLehn, Jones, & Chi, 1991), a computational model of problem-solving skill acquisition. Cascade was originally developed to understand the mechanisms of the self-explanation effect (Chi, Bassok, Lewis, Reimann, & Glaser, 1989; Pirolli & Anderson, 1985). Jones and Fleischman demonstrated that the mechanisms underlying self-explanation might also explain the effectiveness of studying faded examples. Although they showed that Cascade is consistent with the fading result, the explanation involved assumptions that had not yet been tested empirically. Therefore Jones and Fleischman (2001) finished with a small set of predictions and suggestions for new experiments to confirm or dispute Cascade's account. Since that time, Renkl, Atkinson, and their colleagues have run additional experiments, collecting detailed transcripts of subjects studying two types of faded sequences of problems. Although the experiments are not yet complete, we have been able to perform a qualitative analysis of the protocol data for eight of the subjects. Additionally, we have fine-tuned Cascade's knowledge base (but not its underlying mechanisms) to more faithfully model the current data. This paper reports the result of using Cascade to develop a qualitative analysis of the eight subjects. The primary result is that the findings remain consistent with Cascade's account of example fading, as well as the predictions made by Jones and Fleischman (2001).

Background

Years of research have demonstrated effective techniques for teaching students problem-solving skills in a variety of task domains. In particular, a number of studies show that students benefit from being given a series of completely worked example problems, followed by a series of unworked practice problems (e.g., Chi et al., 1989; Pirolli & Anderson, 1985; Renkl, 1997, VanLehn, 1996). Other studies show that the effectiveness of such a curriculum depends in part on the willingness of the students to explain the worked examples to themselves in detail, rather than simply giving the examples a superficial read (Chi et al., 1989; Fergusson-Hessler & de Jong, 1990; Pirolli & Bielaczyc, 1989). VanLehn and Jones (1993a, 1993b; VanLehn et al., 1991) developed Cascade in order to determine the cognitive mechanisms behind this *self-explanation effect*. In essence, Cascade suggests that thorough study of worked examples help students consciously expose and patch gaps in their task knowledge. In addition, self-explanation provides contextual memories that can guide future problem solving by analogy to familiar examples.

Subsequent experiments by Renkl et al. (2000) suggest that student learning can improve even further by *fading* a curriculum from fully worked examples to partially worked examples. The partially worked examples provide a complete solution to the problem (as with fully worked examples), but first require students to derive one or more steps on their own. This in turn requires the students to understand the rest of the example in at least enough detail to be able to attempt a solution.

Jones and Fleischman (2001) argue that the reason faded examples improve learning is that they retain much of the guidance provided by the context of a solved example, but they force the students to work on particular parts of the problem, in turn possibly forcing them to expose and patch knowledge gaps. This is in contrast to studying completely worked examples, where it is basically up to the students to decide whether they are going to put any effort into understanding the examples (because the students are not required to produce any answers in that case). This argument came directly from the assumption that Cascade is

[*] The second author is also affiliated with Soar Technology, Inc.

an accurate model of human problem solving and learning (at this level of abstraction).

Jones and Fleischman ran Cascade on a mock "faded" curriculum in order to demonstrate the plausibility of their hypothesis. This exercise confirmed that the proposed explanation is a sufficient account of the general fading results, but the explanation rests on a number of assumptions that had not yet been confirmed by empirical data. The first assumption was that the classical physics problem domain (implemented in Cascade and studied by Chi et al., 1989) is sufficiently similar to computing simple probabilities (studied by Renkl et al., 2000). The second assumption was that Cascade's underlying processes accurately match what subjects do when learning from faded examples. Data had simply not yet been collected to argue this point either way. Thus, Jones and Fleischman (2001) presented three specific predictions to be confirmed or denied by subsequent empirical research:

1. "Faded examples cause effective learning by forcing the student to encounter and overcome an impasse."
2. There is likely "…at least some benefit to example fading from the learning of search control knowledge."
3. "The primary benefit of a faded example is that it forces the student to process parts of the example that they might otherwise ignore."

They also suggested that these predictions be tested with new experiments that include the collection of protocol data.

The current work tackles both of these issues. To begin with, Renkl and Atkinson (and their colleagues) have initiated an additional study to collect more detailed subject data, including transcribed talk-aloud protocols generated by the subjects while studying and solving problems. Although their experiment and analysis is not yet complete, they provided us with eight initial transcripts, enabling us to generate a partial coding that tests the predictions listed above.

We have also generated a new task knowledge base for computing probabilities, so Cascade can solve precisely they same problems given to the subjects in the new experiments. This allows us to remove a model assumption, and verify the Cascade results with a more accurate match to the data. The next section describes the methods we used to generate the new knowledge base and encode the protocols. The following sections present the results of those activities in more detail.

Methods

Given the study material presented to the experimental subjects, we first performed a thorough task analysis. This involved identifying the probability equations required for solving the set of study problems. This set serves as the target knowledge base that we would expect a "perfect learner" to have acquired after complete study of the curriculum. The task analysis allowed us to replace Cascade's physics task knowledge with task knowledge about computing probabilities. It is important to note that we only changed Cascade's task knowledge. We did not change any of the underlying problem-solving or learning mechanisms built into Cascade. Once we defined the target knowledge base, we represented each problem as a set of given and sought quantities, using Cascade's representation language within Prolog.

After doing the task analysis, we ran Cascade on each problem in order to perform a content analysis. The content analysis records the required equations for each solution. This allows us to predict interactions between performance on separate problems by a single subject. For example, if a subject fails to use an identified equation in one problem (as suggested by an error combined with protocol evidence), but then correctly solves a subsequent problem that requires the same equation, we can safely hypothesize that some sort of learning took place even if there is no direct evidence of a learning episode in the protocol transcript. This helps us track learning across a series of problems. The content analysis also helps constrain the encoding of the subject protocols. In the face of ambiguous utterances that lead to a correct solution, we can generally infer which equations the subject must have used correctly.

Our final task was to encode the subject protocols for behavior episodes relevant to the predictions reported above. Future work (when all subject protocols are available) will contain thorough quantitative analyses of various protocol encodings. For the current effort, however, we performed a qualitative analysis, looking for general trends in the data. The goal was to investigate whether there were any relationships between example fading and learning, the use of analogies for search control, and the generation of self-explanations. We constrained the protocol encoding by performing a goal decomposition to match each subject protocol. Running Cascade on the same problems generates similar goal decompositions, which we can then use to inform the coding process.

Task and Content Analysis

As mentioned above, the task analysis determined all of the equations, or knowledge chunks, required to solve the set of probability problems from the empirical study. We did not include more basic arithmetic reasoning (such as addition, multiplication, and the ability to isolate variables) in the analysis. This is because, for the domains Cascade has been used to study so far, it simplifies the model to assume that subjects have well rehearsed knowledge of these tasks. For the problems in question, we identified twelve distinct, required equations. Some of the equations compute simple probabilities by dividing the cardinality of various sets of objects. The task knowledge includes cases for choosing objects at random with and without replacement. The target knowledge base also includes various equations for combining probabilities. These include the special multiplication rule

$$P(e1 \text{ and } e2) = P(e1) \times P(e2)$$

the addition rule

So I have a total of 12 bottles and there are 4 that are turned into vinegar. So a total of 4 vinegar and 8 drinkable. Probability of vinegar is 1/3 and drinkable is 2/3. Now if we take one then we're left with…There is a 1 in 3 chance that that will be vinegar…	S: value(p(event1a)) S: solve(p(event) = size(selectionpool) / size(totalpool)) S: value(size(selectionpool)) F: value(size(selectionpool)) = 4 S: value(size(totalpool)) F: value(size(totalpool)) = 12 F: solve(p(event) = size(selectionpool) / size(totalpool)): 1/3 F: value(p(event1a)) = 1/3

Table 1. A protocol excerpt and its corresponding analysis in the form of a Cascade trace.

$$P(e1 \text{ or } e2) = P(e1) + P(e2) - P(e1 \text{ and } e2)$$

and the subtraction rule

$$P(\text{not } e) = 1 - P(e).$$

It is worth comparing some features of this domain with classical mechanics, the domain used in previous studies with Cascade. The physics and probabilities domains share the feature of involving mostly symbolic problem-solving skills, which we feel is the defining characteristic that allows Cascade to model both well. However, there are also some potentially significant differences between the two domains.

To begin with, the target knowledge base for computing probabilities is much smaller than the physics knowledge base. In contrast to the current set of 12 equations, Cascade required explicit representation of 62 separate chunks of knowledge for classical physics. Another significant difference is that subjects often relied on common-sense reasoning to explain and learn physics skills. Thus, the Cascade model for physics included a number of general common-sense rules that could be used to guide the learning of correct (and sometimes incorrect) physics knowledge. In contrast, there seems to be much less opportunity to generate common-sense explanations for the rules of probabilities (although there are certainly some). In the protocols we have studied so far, subjects generally made little use of common-sense principles when they got stuck.

To perform the content analysis, we encoded the problems provided by Renkl and Atkinson into Cascade's representation. This essentially involved translating the problems' given and sought quantities into a Prolog-style predicate representation. Once complete, we ran Cascade to ensure it could solve each problem with the complete knowledge base. Each problem run generated an execution trace that provides explicit detail about which equations are necessary to solve each problem. With these tools in hand, we proceeded to analyze each subject protocol to track the usage of individual equations, self-explanation behavior, the use of analogy to guide problem solving, and learning.

Protocol Analysis

The basic approach to the protocol analysis was to assume that Cascade provides an accurate model of each subject's behavior and then to look for inconsistencies. We patterned this approach after Jones and VanLehn's (1992) evaluation of Cascade's ability to model the fine-grained behavior of individual subjects studying and solving physics problems. For each subject-problem pair, we generated the hypothetical solution trace that Cascade would have to generate in order to produce the utterances observed in the subject. We allowed ourselves to tune the trace only by assuming that Cascade has missing or incorrect knowledge about computing probabilities. Any other discrepancies between Cascade and the protocol data are marked against Cascade's ability to explain the subject's performance. Table 1 presents an example protocol excerpt and the corresponding Cascade trace that matches the internal behavior suggested by the subject's utterances.

We constructed Cascade-like traces for a number of problems, and used those results to guide the rest of our protocol analysis. Recall that the predictions presented by Jones and Fleischman (2001) focused on self-explanation, forced impasses, knowledge acquisition from impasses, and knowledge tuning. Thus, our protocol analysis focuses on these three issues.

Self-Explanation

For Cascade's account of fading to be correct, it must mean that subjects tend to generate more self-explanations (or problem-solving activity) for faded examples than they do for completely worked examples. This prediction is borne out in the protocol data we examined. Subjects rarely engaged in self-explaining on fully worked examples. Subjects were much more engaged in the faded examples, presumably because the faded examples demanded them to generate some kind of answer. The protocols show evidence that sometimes even this was not enough to ensure self explanation. Sometimes, subjects would simply "click through" the faded portions of the examples, and skip on to the solutions. For example, after revealing a solution step, one subject simply said "Boy…summmsumm…I don't know this right now," and proceeded to the next step. However, there were certainly many more instances of self-explanation for faded examples than there were for completely worked examples.

Impasses

Subjects encountered impasses during fully worked examples even more rarely than they bothered to self-explain the examples. This is because a subject cannot experience an impasse without first engaging in some self-explanation. However, subjects experienced many impasses when working on faded examples. This is because, if the subject made a mistake, they received relatively immediate feedback by then being shown the correct solution step. If the subject bothered to read the revealed solution, they would have to acknowledge a discrepancy and go back to refigure things. The following excerpt shows an example of a subject first reading a completely worked example (Problem 5) with no impasse, and then working a faded example (Problem 6) that forces an impasse. Both problems require precisely the same set of equations to generate a correct solution.

> Problem 5:
> S: Okay, let's see here. Probability is 1/10 time 1/5, okay, I see how they did it, allright. Probability of stitching and/or color defects is 1/10 plus 1/5 minus the total probability that's 1/50, and that equals (reads aloud) okay, next.
>
> Problem 6:
> S: Okay, allright. Now. This is the difference, that's going to be 1 minus the 2/50 plus the 23/50, that's going to be, 1 minus okay, 2/50 or, 2/50 equals .04, and plus 23/50 equals .46, now, .46 and .04, give me .50, 1 minus .50 equals .50, so, I'm doing this right, it should be .50, no, okay, allright, let's see, okay, I guess I…, okay, so that's 1 minus that, okay, I see what I did.

In this excerpt, the subject essentially just reads Problem 5 and claims to understand it (which is, paradoxically, a hallmark of subjects that are *not* doing enough self explanation). The subject generates an answer to Problem 6 that they think is correct, but when they reveal the correct solution step they discover they are wrong. This forces an impasse. In this particular excerpt, there is no convincing evidence that the subject actually resolved the impasse and learned the correct solution sequence, but the impasse at least gave them the opportunity. The following sections discuss analysis of actual learning episodes in the protocols.

Knowledge Acquisition

We were surprised to find no obvious episodes of knowledge acquisition in the protocols. That is, we found no evidence that subject were missing entire chunks of knowledge that they were then able to discover in response to an impasse. This was particularly surprising because Jones and Fleischman (2001) assumed a key role for knowledge acquisition episodes during their initial Cascade study with physics problems. It appears that this is a place where differences in the task domains are significant. As mentioned previously, knowledge acquisition episodes were an extremely important part of Cascade's account of the self-explanation effect for the physics domain. However, in all of the protocols for subject computing probabilities, it appears that they already *know* all of the equations they need; they just have not yet learned the right times to *use* them. This is admittedly a subtle distinction that cannot always be verified in the protocol data, so we plan to give it a much closer look in future studies. However, since our original proposal gave such a large role to knowledge acquisition, we feel we are being conservative by suggesting that there are no knowledge acquisition episodes at all in the current protocol data. There are clearly other types of learning episodes in the data, which we describe below, and we feel that those remain consistent with Cascade's predictions about fading.

Knowledge Tuning

One of the predictions about Cascade's account of fading was that fading enables students to tune knowledge they have already acquired, by allowing them to use it in a useful problem-solving context. In Cascade, all knowledge tuning occurs via a process of analogical search control. Thus, we expect to see subjects learn after they have successfully drawn an analogy between two problems. We observed many such episodes in the current protocol data. The following excerpt provides one of the clearest examples:

> Problem 2:
> S: …The chance of it being drinkable is 8 to 11 so the probability of her drinking, probability that the first bottle is vinegar but the second is drinkable, 2 red balls and 2 white balls is 4, probability is 1/2 so if we multiply 1/3 times 8/11 that will be 8/33…

Problem 2 involves a collection of bottles containing wine and vinegar. However, in the middle of the excerpt, the subject makes an explicit analogical reference to Problem 1, which deals with selecting a particular configuration from a collection of red and white balls. In a subsequent problem that uses precisely the same solution technique, the subject easily solves the problem correctly, without any evidence of an impasse or overt analogical reference.

It seems clear that this particular episode involves knowledge tuning via analogy. There are other episodes of knowledge tuning that are not overtly analogical. The current Cascade model dictates that all knowledge tuning occurs by analogy, but that happens at a low enough cognitive level that it is difficult to prove or disprove. Certainly there are many cognitive theories that posit some sort of similarity-based memory for skills and facts.

There is one aspect of knowledge tuning that Cascade does not model well. Some subject protocols show basically the same pattern as the excerpt above, but the tuning occurs more gradually across 3 or 4 problems. Cascade's knowledge tuning mechanism is more of an all-or-nothing proposition. As soon as Cascade solves one problem by analogy, it can immediately retrieve the same knowledge in similarly structured future problems. This seems to be a clear weakness in the Cascade model. However, it does not invalidate Cascade basic account of fading. The subject protocols show that the use of analogy occurs more frequently during fading-driven impasses than during the study of completely worked examples.

In prior Cascade studies in the physics domain, there were strong interactions between the knowledge tuning and knowledge acquisition mechanisms (VanLehn & Jones, 1993b). We expect that similar interactions could help explain the effectiveness of fading examples. However, since we have so far seen no evidence of knowledge acquisition in the current study, it has not been important to analyze potential interactions.

Conclusions

We conclude by reiterating the predictions that Jones and Fleischman (2001) proposed to gather evidence for Cascade's account of the benefit of faded examples:

1. "Faded examples cause effective learning by forcing the student to encounter and overcome an impasse."
2. There is likely "…at least some benefit to example fading from the learning of search control knowledge."
3. "The primary benefit of a faded example is that it forces the student to process parts of the example that they might otherwise ignore."

We feel that our initial analysis of protocol data from Renkl, Atkinson, and colleagues confirms each of these predictions to some extent. The basic effect is strong: students often do not expend much effort on understanding completely worked examples, but fading the examples gives the students a strong impetus to do so. This encouragement to work out portions of the examples leads to more opportunities to identify incorrect (or incorrectly applied) knowledge, which in turn provides opportunities to correct or tune that knowledge. We were surprised to find that knowledge acquisition did not appear to play a significant role in the probabilities task domain. However, future analysis will more closely search for such episodes. In addition, although it makes the model less interesting in some ways, the preponderance of analogical knowledge tuning is entirely consistent with the Cascade model. Since Jones and Fleischman's (2001) original study of fading with Cascade did not focus strongly on knowledge tuning (because it played less of a role in the physics domain), an important future task is to run a thorough set of experiments with Cascade to confirm that knowledge tuning can account for all of the observed improvements in problem-solving skill.

It is also certainly possible that future analysis will uncover data that is inconsistent with Cascade's predictions. With this possibility in mind, our next course of action is to gather even more data and perform a more thorough quantitative analysis. We also expect that we will find some ways in which Cascade should be improved. For example, we already know that the knowledge tuning mechanism should be adjusted to account for more gradual forms of knowledge tuning observed in the subjects. Any further mismatches of the model to the data should also serve to improve our understanding of how humans learn problem-solving skills and, as a consequence, inform how we ought to teach them.

Acknowledgments

This project could not have proceeded without the gracious collaboration of Alexander Renkl and Bob Atkinson, together with their research groups. This work was also made possible by the support of Colby College, particularly the Colby College Senior Scholars program, and Soar Technology, Inc.

References

Chi, M. T. H., Bassok, M., Lewis, M. W., Reimann, P., & Glaser, R. (1989). Self-explanations: How students study and use examples in learning to solve problems. *Cognitive Science, 13*, 145–182.

Ferguson-Hessler, M. G. M., & de Jong, T. (1990). Studying physics texts: Differences in study processes between good and poor solvers. *Cognition and Instruction, 7*, 41–54.

Jones, R. M., & Fleischman, E. S. (2001). Cascade explains and informs the utility of fading examples to problems. *Proceedings of the Twenty-Third Annual Conference of the Cognitive Science Society* (pp.459-464). Mahwah, NJ: Lawrence Erlbaum.

Jones, R. M., & VanLehn, K. (1992). A fine-grained model of skill acquisition: Fitting Cascade to individual subjects. *Proceedings of the Fourteenth Annual Conference of the Cognitive Science Society* (pp. 873–878). Hillsdale, NJ: Lawrence Erlbaum.

Pirolli, P., & Anderson, J. R. (1985). The role of learning from examples in the acquisition of recursive programming skills. *Canadian Journal of Psychology, 39*, 240–272.

Pirolli, P., & Bielaczyc, K. (1989). Empirical analyses of self-explanation and transfer in learning to program. In G. M. Olson & E. E. Smith (Eds.), *Proceedings of the Eleventh Annual Conference of the Cognitive Science Society* (pp. 450–457). Hillsdale, NJ: Lawrence Erlbaum.

Renkl, A. (1997). Learning from worked-out examples: A study on individual differences. *Cognitive Science, 21*, 1-29.

Renkl, A., Atkinson, R. K., & Maier, U. H. (2000). From studying examples to solving problems: Fading worked-out solution steps helps learning. In L. R. Gleitman & A. K. Joshi (Eds.), *Proceedings of the Twenty-Second Annual Conference of the Cognitive Science Society* (pp. 393–398). Mahwah, NJ: Lawrence Erlbaum.

VanLehn, K. (1996). Cognitive skill acquisition. *Annual Review of Psychology, 47*, 513–539.

VanLehn, K., & Jones, R. M. (1993a). Learning by explaining examples to oneself: A computational model. In S. Chipman & A. L. Meyrowitz (Eds.), *Foundations of knowledge acquisition: Cognitive models of complex learning*. Boston: Kluwer Academic.

VanLehn, K., & Jones, R. M. (1993b). Integration of explanation-based learning of correctness and analogical search control. In S. Minton (Ed.), *Machine learning methods for planning*. Los Altos, CA: Morgan Kaufmann.

VanLehn, K., & Jones, R. M. (1993c). Better learners use analogical problem solving sparingly. *Machine Learning:*

Proceedings of the Tenth International Conference (pp. 338–345). San Mateo, CA: Morgan Kaufmann.

VanLehn, K., & Jones, R. M. (1993d). What mediates the self-explanation effect? Knowledge gaps, schemas or analogies? *Proceedings of the Fifteenth Annual Conference of the Cognitive Science Society* (pp. 1034-1039). Hillsdale, NJ: Lawrence Erlbaum.

VanLehn, K., Jones, R. M., & Chi, M. T. H. (1991). A model of the self-explanation effect. *Journal of the Learning Sciences, 2*, 1–59.