Title
Cascade Explains and Informs the Utility of Fading Examples to Problems

Journal
Proceedings of the Annual Meeting of the Cognitive Science Society, 23(23)

ISSN
1069-7977

Authors
Jones, Randolph M.
Fleischman, Eric S.

Publication Date
2001

Peer reviewed

Cascade Explains and Informs the Utility of Fading Examples to Problems

Randolph M. Jones (rjones@colby.edu)
Eric S. Fleischman (esfleisc@colby.edu)
Computer Science, Colby College
5847 Mayflower Hill Drive
Waterville, ME 04901-8858 USA

Abstract

Recent research demonstrates that people learn to solve problems more effectively when presented with a series of faded examples and problems, than when presented with completely worked out examples and completely unsolved problems alone. We propose an explanation for this effect, based on the Cascade model. Cascade was originally built to model the self-explanation effect, but it also accounts for other aspects of human learning and problem solving strategies. A relatively straightforward application of Cascade, without alteration, also explains why fading might be beneficial. This explanation provides further support for Cascade as an accurate model of human learning and problem solving, and it also augments the results on fading with specific insight into how, and when, example fading should be effective.

Introduction

There are a variety of research results characterizing factors that facilitate the human ability to learn how to solve problems. Among recent results, Renkl, Atkinson, and Maier (2000) report that people learn more effectively when presented with a set of "faded" examples and study problems than they do when presented with study examples alone (followed by completely unworked problems). Given this result, our interest is in identifying the cognitive mechanisms that explain why the effect exists. Additionally, if we understand the underlying mechanisms, they may provide extra insight into precisely how to create effective sets of study examples and problems. We have identified a potential set of explanatory mechanisms, which are precisely the mechanisms that define Cascade, an existing model of human learning in problem solving (VanLehn, Jones, & Chi, 1991; VanLehn & Jones, 1993a). The Cascade model suggests which study and learning mechanisms contribute to the fading effect, and provides a tool for pinpointing precisely which types of faded examples will be most effective.

Fading Examples

A common curriculum for teaching students to solve problems in a particular domain involves having the students first read some domain material, then study some completely worked out example problems, and finally solve problems that have not been worked out. Various studies explore why it is beneficial for students to study completely worked out examples (e.g., Chi et al., 1989; Pirolli & Anderson, 1985; Renkl, 1997, VanLehn, 1996). The technique of *fading examples* constitutes a particular variation on such a course of study. When fading, the teacher first presents a completely worked example and then follows this with a series of *nearly* completely worked examples. Each subsequent problem removes an explanatory step, forcing the students to solve that portion of the problem themselves. This gives more guidance than a regular problem, because the student still has the guidance of the rest of the worked out example, but it gives less guidance than a completely worked example.

Renkl et al. (2000) demonstrated that different particular methods of presentation can yield different levels of learning. Specifically, they showed that fading sometimes improves learning over other orders of presentation. In their initial study, they presented subjects with a completely worked example, followed by an example that omits the last step in the solution. This was then followed by an example that omits the second to last and last steps and finally one presentation of a completely unworked problem. They compared this situation to one in which students first studied a completely worked example, and then solved a completely unworked problem. The results of this study confirmed that the group of subjects exposed to faded examples learned more effectively than the other group, even though both groups had similar pre-test performances. A more thorough, but similar, experiment produced comparable results.

Renkl et al.'s study provides us with valuable information about potential ways to design study curricula. However, the research also leaves a few unanswered questions. Among these are *why* fading works as well as it does. In addition, it would be useful to know exactly what forms of fading will be effective. Is it always the case that examples should be faded from back to front? Or are there more formal methods we can use to design faded examples? Such questions can, and should, be answered in part by further experimentation. However, we believe that an existing computer model of human learning in problem solving can also shed light on some of the answers. At worst, the model can help guide future experimentation. At best, the model makes specific predictions and recommendations about how to fade examples.

The Cascade Model

Before presenting Cascade's account of example fading, we will first provide an overview of Cascade's cognitive mechanisms, and how they contribute to explaining other experimental results on human problem solving and learning. Cascade was originally developed to explain the cognitive mechanisms involved in the *self-explanation effect* (Chi et al., 1989; Fergusson-Hessler & de Jong, 1990; Pirolli & Bielaczyc, 1989). Simplifying a bit, the effect shows that people learn more effectively by studying examples when they are careful to explain to themselves as many steps of the example as they can. Students who do not carefully explain worked out example steps do not perform as well on subsequent problems.

Cascade explains this effect as the interaction between two basic learning mechanisms (VanLehn et al., 1991; VanLehn & Jones, 1993b). The basic prediction of Cascade is somewhat intuitive: A student learns effectively when the student is able to identify and patch a specific gap in their knowledge. In Cascade, such knowledge acquisition can occur both while studying worked out examples and while solving problems. When studying an example, Cascade must self-explain each worked step. When the system finds a step it cannot explain, this signifies a knowledge gap. Because the example specifies what the answer is, Cascade can (given appropriate domain-general background knowledge) compensate for the gap by constructing an explanation for the answer. When Cascade successfully creates such an explanation, it learns a new piece of knowledge that it has some faith will work in future problems.

When solving problems, Cascade may also learn by exposing and patching knowledge gaps. However, during problem solving, the process is less constrained because the system does not know the correct answer ahead of time. If the system is missing knowledge, there are potentially a number of ways Cascade could go wrong in attempting to solve a problem. Thus, it is highly likely that the system may expose and attempt to patch a "false" knowledge gap. However, when the system explains examples, it also stores search-control knowledge, essentially remembering the subgoal structure of each example. Experiments with Cascade show that this search-control knowledge can be enough to guide the system into *real* knowledge gaps during problem solving. It can then successfully patch those gaps (VanLehn et al., 1991). Without the search-control knowledge provided by self-explaining examples, Cascade cannot distinguish between real knowledge gaps and otherwise unproductive dead-ends in the attempted problem solution.

Experiments with Cascade have focused mostly on problems involving Newtonian physics, due to the fact that the system's creators had access to physics problem protocol data from Chi et al. The experiments show that Cascade's interacting learning mechanisms account for the basic self-explanation effect, and are able to explain a variety of problem-solving and learning strategies on an aggregate and an individual basis (Jones & VanLehn, 1992; VanLehn et al., 1991; VanLehn & Jones, 1993c).

Because it plays into Cascade's account of example fading, it is worth describing part of the experimental methodology used in the previous Cascade work. The basic method for running experiments with Cascade involves three steps:

1. Configure the system's initial knowledge base, to reflect the hypothetical initial knowledge state of a human subject.
2. Force Cascade to explain exactly those pieces of a series of examples that correspond to observed self-explanations in the subject's protocol (by running it in "explain" mode only on those example pieces).
3. Run Cascade on a series of problems, recording answers, errors, and learning events, to compare with similar events in the encoded subject protocol.

This methodology makes explicit that Cascade does not model all the cognitive process in learning from examples and problem solving. For example, although Cascade can explain the results of self-explanation episodes, it currently has no way to predict which pieces of an example a student will choose to self-explain. However, for the cognitive processes that Cascade models, it does a good job of matching detailed protocol data (Jones & VanLehn, 1992). Thus, we will use the same basic formula to test Cascade's account of example fading.

A Potential Explanation for Fading

Given the background details of how Cascade studies examples and learns to solve problems, we can now sketch the theory behind how Cascade ought to be able to explain the fading effect reported by Renkl et al. It should be clear that, if the Cascade model is accurate, fading of examples can lead to improved learning only if it provides the students with more opportunities to patch their knowledge successfully.

However, it is certainly a valid question whether this is really an accurate characterization of the source of the benefit of example fading. According to Cascade, if a student completely self-explains an example, the student would successfully encounter and patch every potential knowledge gap relevant to that example. Thus, there would be absolutely no benefit to first self-explaining the example and then combining self-explanation with regular problem solving on a similar subsequent example. The only way example fading could be effective (according to the Cascade model) is if it leads the student to patch a knowledge gap that it would not patch through example studying alone.

As we mentioned above, Cascade can be configured to emulate the self-explanation behavior of any individual subject (Jones & VanLehn, 1992), or to emulate aggregate self-explanation effects (VanLehn et al., 1991). However, it does not explain how or why a subject chooses to self-explain any particular piece of an example. Although an idealized version of Cascade can self-explain a worked example in its entirety, there was not a single subject in Chi et al.'s study who was so thorough in their self-explanation behavior.

Thus, the evidence and model suggest that completely worked examples are never (or at least seldom) as beneficial to students as they could be, because students never (or at least seldom) completely self-explain the examples. This in turn may cause the students to miss opportunities to learn from the examples. However, when a student is given a faded example, there should be some confidence that the student will focus their attention at least on the part of the example that they are requested to solve. Thus, a faded example is similar to a completely worked-out example with a special highlight that says "make sure you self-explain at least this portion of the example".

Our hypothesis is that there are no additional mechanisms required by Cascade to account for the example-fading effect. Rather, using Cascade's existing mechanisms, we hypothesize that example fading forces the student to pay thorough attention to particular portions of the example. Because this forcing is highly focused, the student has a good chance of successfully exposing and patching a knowledge gap (if the faded piece of the example relies on such missing knowledge).

When Cascade is forced to self-explain an individual "faded" portion of an example, we predict that Cascade will acquire new knowledge if that portion of the example exposes a knowledge gap. In such a case, Cascade will exhibit improved learning over normal example studying *if* Cascade was not originally forced to self-explain the same example portion. Due to the fact that we do not have such protocol data from Renkl et al., there is currently no empirical way to tell if this is an accurate characterization of when Renkl et al.'s subjects learned. However, it does provide a testable prediction. Before that experiment is run, however, we should first test the Cascade model to make sure our predictions about its behavior during example fading are accurate.

As an example, a first-year college physics text will often contain an example with a block suspended from a string and sitting on an inclined plane. The example will include some lines that, implicitly or explicitly, involve the inclination of the normal force exerted by the inclined plane on the block. It is not uncommon for a first-year college student never to have heard of normal force, much less to know its inclination. If the student self-explains the entire example, they will come across these portions, reach an impasse, and be forced to create some new understanding concerning normal force. If the student does not bother to self-explain the important parts of the example, they will happily go off and solve inclined plane problems without involving normal force at all (such episodes appear in Chi et al.'s protocol data). Consider taking the same example, but replacing one part of the work with the problem statement, "What is the inclination of the normal force?" This certainly draws the student's attention to normal force, and it explicitly requires the student to understand the concept, if the student is going to be able to finish the problem. In contrast, a student can very easily read a completely worked example without bothering to learn about normal force at all.

Experimental Design

We ran Cascade on Newtonian physics problems, because we already had a thorough task analysis and knowledge representation for the physics domain.

The basic approach involves first using Cascade as a knowledge analysis tool. We ran the system on all the examples and problems given to Chi et al.'s subjects, determining which examples and problems provided the first opportunity to learn a variety of pieces of knowledge. In addition, we used the model to determine which problems required the application of particular pieces of knowledge. This exercise allowed us to create a dependency chart for a number of different knowledge chunks. The chart basically tells us "if you force Cascade to learn this chunk from this example, then it ought to be able to solve this piece of this problem." Or in some more interesting cases, if you force Cascade to learn a particular chunk from an example, it provides the scaffolding for the student to learn *another* chunk in a subsequent problem.

Given this knowledge analysis, it allows us to create experimental runs that simulate effective example fading. The analysis tells us exactly where in each example each knowledge chunk can be learned. Thus, we can focus the system, telling it to explain exactly that portion of the example that will lead to the acquisition of the desired knowledge chunk. This corresponds to creating a focused "faded example", where we allow Cascade not to self-explain most of the example, but force it to self-explain exactly the correct piece. We can then run Cascade on the same example, but with an additional faded portion, so the system can learn a second chunk from the same example. As a control, we can force Cascade to self-explain example pieces that do not cause any learning, and demonstrate that this has no beneficial effect on future problem solving. An additional control has Cascade "study" the examples without self-explaining the lines, in which case no useful learning occurs.

Results

As predicted, the experiments with Cascade were able to demonstrate improved learning with simulated fading of examples over normal processing of examples. The knowledge analysis showed where Cascade required, or did not require, particular pieces of knowledge to solve a problem. In some cases, the system could solve problems even without any example studying. In other cases, the model was unable to solve problems due to a particular lack of knowledge, even if the system had "studied" an appropriate example without actually self-explaining the key portion of the example. With simulated fading, the system was forced to self-explain the appropriate portions of the examples, and then could solve more problems. Even more, the knowledge analysis was able to tell us exactly which pieces of certain examples should be faded in order to allow the system to solve each particular problem.

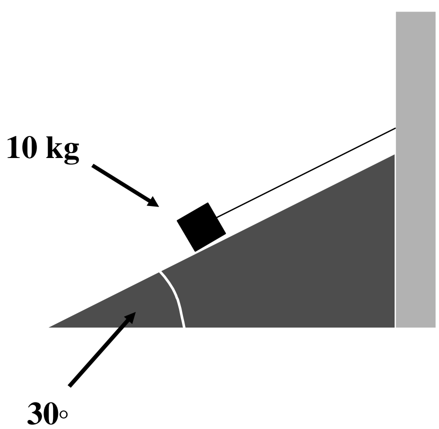

Figure 1: A typical inclined-plane problem. What is the tension in the string?

The experiments we report here concentrated on a thorough analysis of one particular sequence of examples and problems in the Chi et al. study. These examples and problems were all variations of inclined-plane problems (see Figure 1). Given Cascade's initial knowledge base, complete self-explanation of two inclined-plane examples and one "weights and pulleys" example provides the opportunity to expose and patch 12 knowledge gaps.

These opportunities appear in specific portions of the explanation traces for the examples. For example, Cascade can only learn about normal force from self-explaining the normal-force vector in the free-body diagram of the inclined-plane example. If Cascade fails to self-explain this portion of this example, it generates incorrect answers on any future problems that involve normal force. In fact, it usually generates the same incorrect answers as students who also fail to self-explain that portion of that example (Jones & VanLehn, 1992). A portion of the dependency table that covers this chunk appears in Table 1.

Table 1: Dependency chart for the normal-force chunk.

Chunk	`normal-force`
Learning opportunities	Example ixa, line 2 Example ixb, line 2
Used in problems	i1a, i1b, i2a, i2b, i3a, i3b, q5

In one experiment, we forced Cascade to learn about normal force by fading the normal-force portion of the example. Even if Cascade did not self-explain any other part of that example, it learned the normal-force chunk because we forced its attention there.

In each of our experiments, we deleted one of the 12 target rules from Cascade initial knowledge. We then compared Cascade's behavior without any self-explanation to Cascade's behavior when it is forced to self-explain only the portion of the examples that leads it to learn the deleted chunk. The results show that Cascade cannot solve all of the inclined-plane problems without acquiring all 12 of the target pieces of knowledge. It cannot solve *any* of the problems without learning at least some of the 12 chunks (although each of the problems only makes use of a subset of the 12 chunks). More importantly, the results tell us exactly which example portions will lead to success on which problems.

In some cases, Cascade is also able to learn new knowledge chunks during problem solving. In fact, it cannot solve some of the problems without also patching some knowledge gaps during the solution of that problem. The experiments also show that this learning is not always successful unless Cascade has first acquired all the knowledge it needs from the examples. Again, perhaps most informative is that the experiments tell us which pieces of the examples are most beneficial to fade. For example, Cascade cannot solve any problems correctly without first being forced to learn about normal force (as discussed above). But if Cascade is not forced (during example studying) to learn about the inclination of an inclined block's acceleration, it is able to learn that particular piece of knowledge (or at least compensate for it and get the right answer) during problem solving. It can only do this, however, if it has learned about the *existence* of normal force from one of the examples (or from prior experience).

In general, the experimental results confirm our contention that graded fading of examples can map to improvement in Cascade, if we correspond fading with the forced self-explanation of portions of each example. These results demonstrate the basic dependencies we predicted. Thus, without any alteration to the underlying Cascade model, it is able to provide an explanation for the benefit of faded examples. The key assumption is that example fading serves primarily as a focus of attention, forcing the subject to study closely productive portions of each example. In the closing section, we discuss some implications of this assumption, as well as the possibility of further experiments that would identify more complex interactions between fading and learning.

Conclusions

We feel this work provides two basic contributions. First, it offers additional evidence that Cascade is an accurate model of (at least some of) the cognitive mechanisms involved in studying examples and learning to solve problems. In some ways, this may only be of interest to the designers of Cascade. However, more generally, it allows us to have more faith in using Cascade as a tool both to study human behavior, and perhaps as an aid for curriculum development. In this work, we were able to use Cascade not only as a cognitive model, but also as a knowledge analysis tool. It allowed us to perform a focused dependency analysis that would certainly be beneficial to an instructor creating examples and problems.

However, we should note that much of the hard work that made this analysis possible was performed years ago. We have the benefit of the detailed cognitive analysis of college physics that went into the original design of Cascade. To apply Cascade to any new domains would require a similar intensive effort. However, Cascade at least provides a framework and set of assumptions for creating such knowledge representations. Furthermore, it provides clear principles for where, and how, examples and problems will cause a student to learn target knowledge chunks, as well as which target knowledge chunks contribute to the solution of target test problems.

The second contribution is that we have enhanced our knowledge of how, and why, example fading proves beneficial in some circumstances. The Cascade model suggests that arbitrary fading of examples is not likely to be fruitful, in general. Rather, fading should be focused towards the pieces of examples that will enable learning of a teacher's target knowledge chunks.

Again, this conclusion depends on a key assumption that is integral to Cascade's candidacy as an accurate cognitive model. If Cascade is accurate, it must be the case that faded examples cause effective learning by forcing the student to encounter and overcome an impasse. This is a prediction that can easily be tested. Future experiments on fading should include detailed protocol analysis, and should look for evidence of impasses and learning events in the faded portions of the examples. Such data will confirm or disprove Cascade's account of fading.

It would be prudent to note that the account of fading presented here does not need to be the exclusive source of improved learning. As we mentioned in describing Cascade, the system relies on an interaction between a domain knowledge acquisition learning mechanism and a search-control knowledge learning mechanism. The experiment and explanation reported here relies almost exclusively on the knowledge acquisition mechanism. However, our knowledge of Cascade suggests that it would likely also predict at least some benefit to example fading from the learning of search control knowledge.

Even if a faded portion of an example does not force an impasse and learning event, if it forces some amount of problem solving, Cascade predicts that a student would acquire search control knowledge for the goals and subgoals addressed in the faded portion of the example. As with Cascade's model of the self-explanation effect, such search-control knowledge could benefit later learning, even if no knowledge chunks are learned during the solution of the faded example. However, we have not yet run an experiment along those lines. For now, we will leave that form of learning from a faded example to speculation, to be confirmed or rejected later. It would be most interesting to conduct such computational experiments in concert with similar detailed protocol studies of human subjects. The main point is that Cascade predicts the primary benefit of a faded example is that it forces the student to process parts of the example that they might otherwise ignore. Once a portion of the example is processed, all the cognitive mechanisms that Cascade posits can be brought to bear on learning.

Acknowledgements

This work would not have been possible without the original development of the Cascade system. That, in turn, would not have been possible without the efforts of Kurt VanLehn and Michelene T. H. Chi.

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