Georgia State University

ScholarWorks @ Georgia State University

2016

Improving Problem Solving with Subgoal Labels in Expository Text and Worked Examples

Lauren Margulieux
Georgia State University, lmargulieux@gsu.edu

Richard Catrambone
Georgia Institute of Technology, richard.catrambone@psych.gatech.edu

Follow this and additional works at: https://scholarworks.gsu.edu/ltd_facpub

Part of the Instructional Media Design Commons

Improving Problem Solving with Subgoal Labels in Expository Text and Worked Examples

Submitted: December 16, 2015

Abstract

In highly procedural problem solving, procedures are typically taught with context-independent expository text that conceptually describes a procedure and context-dependent worked examples that concretely demonstrate a procedure. Subgoal labels have been used in worked examples to improve problem solving performance. The effect of subgoal labels in expository text, however, has not been explored. The present study examined the efficacy of subgoal labeled expository text and worked examples for programming education. The results show that learners who received subgoal labels in both the text and example are able to solve novel problems better than those who did not. In addition, subgoal labels in the text appear to have a different, rather than an additive, effect on learners compared to subgoal labels in the example. Specifically, subgoal labels in the text appear to help the learner articulate the procedure, and subgoal labels in the example appear to help the learner apply the procedure.

Keywords: STEM education; subgoal learning; worked examples; expository text; procedural instructions.

1. Improving Problem Solving with Subgoal Labels in Expository Text and Worked Examples

Knowledge of science, technology, engineering, and mathematics (STEM) subjects is increasingly necessary in our society. As STEM fields advance, individuals need to generally understand more about these fields to make well-informed decisions, such as those made when buying technology, and to understand technical information, such as that in a medical diagnosis (Committee on Highly Successful Schools or Programs in K-12 STEM Education (CHSSP), 2011). In addition, individuals with advanced STEM knowledge are needed to fill increasingly technical jobs and promote innovation (Katehi, Pearson, & Feder, 2009). These demands are particularly prevalent in computer science. Using data from the U.S. Bureau of Labor Statistics, Miller (2014) argues that computer science skills are increasingly necessary for careers in all industries and that, by 2020, we will have a million more computer science positions than computer scientists. Cooper, Grover, Guzdial, & Simon (2014) argue that to address the deficit of general computing literacy and qualified computer scientists, interventions to improve computing instruction are needed.

In computing, like in other STEM disciplines, instruction of problem solving procedures includes context-independent expository text to provide conceptual information about a procedure that is abstract enough to transfer to novel problems and context-dependent worked examples to provide procedural information that is concrete enough to grasp (Trafton & Reiser, 1993). Expository text provides information about reasoning within a domain (Reder & Anderson, 1980). Worked examples, in contrast, demonstrate how to apply procedures to specific cases (LeFevre & Dixon, 1986).

Like students in other STEM disciplines, computing students rely upon worked examples to learn procedures (Anderson, Farrell, & Sauers, 1984; LeFevre & Dixon, 1986). Eiriksdottir and Catrambone (2011) argue that worked examples are preferred over expository text because

they take less effort to understand and apply to closely related problems. The concrete features that make worked examples easier to understand also tie them to a particular context. Therefore, when learners rely too heavily on worked examples, they are commonly unable to glean conceptual information about the procedure from these concrete examples, which can inhibit transfer to novel problems (Bassok & Holyoak, 1989; Eiriksdottir & Catrambone, 2011). For this reason, much research has focused on designing worked examples in a way that promotes conceptual understanding and transfer (e.g., Catrambone, 1998; Mayer, 2009; Renkl, 2002).

One instructional design technique for worked examples that has improved problem solving and transfer in STEM disciplines, including computing, is using subgoal labels in worked examples to promote subgoal learning (see an example in Figure 1; Catrambone, 1998; Margulieux, Catrambone, & Guzdial, 2012). To understand what a subgoal is, consider how someone might solve a complex math problem. To achieve the solution (or the overall goal), the problem solver takes multiple functional steps towards that solution (e.g., finding the value of different parts of the problem). These functional pieces are called subgoals, and they comprise one or more individual steps (e.g., summing numbers together). The same subgoals tend to appear across problems within a topic area; therefore, teaching learners to identify and achieve subgoals increases their success at solving novel problems (Catrambone & Holyoak, 1990). Research on subgoal labeled worked examples suggests that by providing a small amount of extra information and structure, subgoal labels improve outcomes in three ways:

- by highlighting the structure of the worked example for the learner,

- by helping the learner mentally organize information,

- and by inducing the learner to self-explain the examples (see Figure 2).

1.1.1 Highlight Structure. To help students learn problem solving procedures from worked examples, the structure of worked examples needs to be emphasized (Atkinson, Derry, Renkl, & Worthham, 2000; Van Gog, Paas, & Van Merriënboer, 2004, 2006). Subgoal labels highlight the procedural structure of examples by grouping steps by structural features. Catrambone (1996) found that learners who receive subgoal labeled examples were more likely than those who did not to mentally chunk steps that were grouped. This chunking changed learners' mental representation of a problem solving procedure from individual steps to subgoals, reducing the demand on cognitive resources (Atkinson & Derry, 2000; Catrambone, 1996). Moreover, this abstract representation of the procedure helped students transfer knowledge to solve novel problems (Catrambone, 1996).

1.1.2 Organize Information. Novices need help to effectively organize new information, even if they recognize the structure of examples (Committee on Developments in the Science of Learning (CDSL), 2000). For example, even though students might recognize the components of a math procedure, they do not necessarily understand how those components relate to each other or how they could be applied to a new problem. Helping learners create an organizational scheme in a domain is critical because how learners organize and interpret new information affects their proclivity to remember, use, and acquire new knowledge (CDSL, 2000). When subgoals in examples are meaningfully labeled, the student gets information about the function that a group of steps achieves (Renkl & Atkinson, 2002). Catrambone (1995a) found that participants who received meaningful subgoal labels in worked examples tended to explain their solutions using those labels, suggesting that is how they mentally organized information. These learners solved problems better than those who did not receive subgoal labels, perhaps because

better mental organization of information in a domain enables better transfer within the domain (CDSL), 2000).

Leaners who receive explanations about the purpose of steps in a worked example typically organize information about the procedure better and transfer the procedure to novel problems better (Van Gog et al., 2004, 2006). For this reason, subgoal labeled worked examples and process-oriented worked examples, which describe the purpose and rationale for each step of a worked example, help students solve novel problems (e.g., Van Gog et al., 2004). Both subgoal labeled worked examples and process-oriented worked examples achieve the same purpose, describing to the learner the purpose of steps of the example, but they differ in the specificity of the instructional explanation. Subgoal labels are context independent, meaning that the same subgoal label is used across a class of problems (e.g., Catrambone, 1998), while process-oriented explanations tend to be context specific, meaning that the explanations in an example apply to only that example (e.g., Van Gog et al., 2004). A recent meta-analysis by Wittwer and Renkl (2010) found that the effect of including instructional explanations on learning was small. This meta-analysis included worked examples with process-oriented explanations but not subgoal labels. The effect of subgoal labels tends to be larger, $d = .2 - .3$ (e.g., Catrambone, 1998; Margulieux et al., 2012), than the effect reported in Wittwer and Renkl (2010), $d = .16$, though this average effect of subgoal labels is based on fewer experiments than were included in the meta-analysis. Subgoal labeled worked examples might be more effective than process-oriented worked examples because the abstract nature of subgoal labels promotes self-explanation.

1.1.3 Promote Self-Explanation. Worked examples have been designed to promote self-explanation and help learners to conceptually understand procedures rather than simply memorize them (Chi, Leeuw, Chiu, & LaVancher, 1994; Wylie & Chi, 2014). The more learners

understand concepts, the better they can adapt and transfer procedures to novel problems (e.g., Bransford, Brown, & Cocking, 2000). Moreover, Sweller (2010) argues that self-explanation guides cognitive resources to focus on relevant information and reduces the effort spent on extraneous information. The amount of self-explanations a typical learner makes is low (Chi et al., 1994; Wylie & Chi, 2014), but explanations can also be prompted by the design of instructional materials (Catrambone, 1998; Renkl & Atkinson, 2002).

By grouping steps of a worked example under a meaningful label, subgoal labels prompt the learner to self-explain how the steps are related to the label (Catrambone, 1995a). Furthermore, when the same subgoal label appears multiple times, students can explain to themselves how the same label applies in multiple instances. Receiving subgoal labels might inhibit the learner from self-explaining the purpose of steps, but it can prompt self-explanation of how the steps of the worked example are related to the label. Furthermore, students can compare methods for achieving a subgoal across several instances and develop a more abstract understanding of the procedure (Atkinson et al., 2000; Catrambone & Holyoak, 1990). Renkl and Atkinson (2002) found that because the subgoal labels describe the function of steps, learners who received subgoal labels had fewer incorrect self-explanations than students who did not. Perhaps most important, though, Renkl and Atkinson (2002) found that designing worked examples to include subgoal labels externally and consistently prompted students to make these self-explanations, unlike self-explanation training, which decayed in efficacy over time.

1.1.4 Expository Text. Past research suggests that students learn better from *worked examples* when they include subgoal labels than when they do not, but the effect of subgoal labels on learning from *expository text* has not been explored. Learning from worked examples typically includes receiving expository text that describes the procedure used in the examples in

general terms. For example, a worked example shows how to use a probability formula to find the probability of an event, whereas expository text describes the formula itself and its different parts. While worked examples can improve immediate application of a procedure to problems similar to that in the worked example, expository text can improve performance and transfer to problems different from that in the example by helping students understand procedures conceptually. Smith and Goodman (1984) found that participants who received structural or functional information about the system on which they completed tasks read the steps faster, recalled them more accurately, and transferred their knowledge to a novel system better than participants who did not receive that information. Even with well-designed worked examples that help learners build a conceptual understanding of a procedure, expository text could be more efficient at conveying this conceptual information. Learners might have to review multiple examples of concrete problems to extract the same level of abstract information that expository text provides directly.

Subgoal labels might be beneficial in expository text because instructions that emphasize the concepts in a domain can help novices learn how to process and organize new information (for an example of subgoal labeled text, see Figure 3). Atkinson, Catrambone, and Merrill (2003) found that learners could transfer knowledge to solve problems more successfully when they received concept-oriented equations (i.e., written to show the purpose of the equation) compared to calculation-oriented equations (i.e., written to expedite calculation). In addition, Catrambone (1995b) found that when learners received general instructions for a domain, they transferred to novel tasks more successfully than participants who received specific instructions.

Subgoal labels in expository text could also be considered a form of signaling. Signals in text refers to clues from the writer to the reader about what is important (Lemarié, Lorch,

Eyrolle, & Virbel, 2008). Signals support identification of important concepts and efficient

processing of text (Lemarié et al., 2008). Subgoal labels could be used as signals to highlight

concepts in expository text and draw learners' attention to them.

Subgoal labels were predicted to be most effective when included in both expository text

and worked examples. Instructions are more effective when students can find connections

between different representations of content (McGee & Reis, 2012). Using subgoal labels in

both expository text and worked examples might help learners connect information from these

two types of instruction. If learners connect information sharing the same label, then they could

more easily integrate information presented in each type of instruction. Though expository text is

important, learners can have difficulty applying abstract instructions to problems (Eiriksdottir &

Catrambone, 2011; VanLehn, Jones, & Chi, 1992). Worked examples are an important part of

instruction because they provide specific information about how to apply concepts to problem

solving (Catrambone, 1998; Trafton & Reiser, 1993). If learners receive only subgoal labeled

expository text, they might have trouble applying that information to problem solving without

subgoal labeled worked examples to guide them.

In summary, subgoal labels in both expository text and worked examples might improve

learners' conceptual understanding of procedures and problem solving performance by

- helping learners structurally understand problem solving procedures in a way that
  enables transfer to novel contexts,

- guiding learners' mental organization of knowledge,

- helping learners understand information by encouraging learning strategies like self-
  explanation,

- And helping learners integrate information from multiple sources.

These first three points match the benefits that research on subgoal labeled worked examples has supported. Subgoal labeled expository text is expected to help students organize information, by signaling the main points of the text, and promote learning strategies by prompting the student to explain how the label relates to the information in the text. Receiving labels in both text and an example was expected to help learners integrate general information from the expository text with the specific information from the worked example. The general purpose of expository text is to teach the structure of a procedure; therefore, subgoal labeled expository text was not expected to help learners structurally understand procedures more than unlabeled text.

1.2 Overview of Experiments

The present study explored the effectiveness of subgoal labeled instructional materials compared to unlabeled instructional materials to teach computer programming. Participants learned to create applications (apps) for Android devices using Android App Inventor. This computer programming language was chosen because it is a drag-and-drop language. Drag-and-drop programming languages allow the user to drag components from a menu and place them together like puzzle pieces, instead of writing code, to create programs (view a screenshot of the interface in Figure 4). This type of code creation is more easily understood by novices (Hundhausen, Farley, & Brown, 2009). Instructions from the ICE Distance Education Portal (Ericson, 2012) were used to develop instructional materials. Materials in all conditions were identical except for the subgoal labels. Subgoals were determined using the Task Analysis by Problem Solving (TAPS; Catrambone et al., 2014) technique with subject-matter experts.

In TAPS the subject matter expert (SME) identifies a set of problems that the SME thinks learners should be able to solve if they conceptually understand the procedure. The SME solves some of the problems while justifying each step to the knowledge extraction expert (KEE) who

is a domain novice. The KEE develops detailed notes based on the solution procedures and justifications (the "why" for each step) provided by the SME. The SME is not invited to provide abstract theory outside of justification for steps. Eventually the KEE uses the notes to solve the problems that the SME already solved; the KEE can request help from the SME to resolve impasses. Throughout this process the KEE continuously updates and reorganizes the notes; this reorganization allows the KEE to develop solution procedures that are independent of specific examples. Once the KEE can solve all of the old problems, the KEE then attempts to solve new problems provided by the SME. When the KEE can, using the notes, solve all problems given by the SME, without the help of the SME, then these notes represent a complete task analysis of the procedure.

The procedure was completed in App Inventor, which has two interfaces. The Designer interface allows the user to make components for an app. For example, if an app had an image, the image would have been created in the Designer (Figure 5 shows the Designer interface). The Blocks Editor interface allows the user to program behaviors for the components using pieces of program called blocks. For example, if an image becomes bigger when it is touched, this behavior would be programmed in the Blocks Editor (Figure 4 shows the Blocks Editor interface).

For instruction, participants received expository text detailing how to create apps (excerpt in Figure 3) and a video demonstration detailing how to create a Fortune Teller app (i.e., a worked example). A video demonstration of an expert making the app and explaining the procedure was used as the worked example because videos can quickly and naturally show learners how to use direct-manipulation interfaces (Palmiter, Elkerton, & Baggett, 1991) such as App Inventor. Participants were also asked to make the app themselves using a textual step-by-

step guide (excerpt in Figure 1) because studying an example and applying the content can lead to better learning than studying alone (Trafton & Reiser, 1993).

The sessions were completed in a computer-based learning environment in a laboratory with an experimenter to provide administrative instructions that were unrelated to the programming task. Experiment 1 explored the efficacy of subgoal labels in general expository text in a laboratory study that had higher control but less ecological validity. Experiment 2 attempted to replicate performance results from Experiment 1 in a more ecologically valid learning scenario with lower control. In Experiment 1, participants were not allowed to use instructional materials when solving novel problems to isolate the effect of instructional design on learning as much as possible. This constraint, however, is not typical for learners who have had less than 30 minutes of instruction. Thus, Experiment 2 permitted participants to use instructions during problem solving. This learning situation affords exploration of the effect of subgoal labeled instructional materials when they could be used as a resource during problem solving.

The assessment tasks in the experiments asked participants to solve novel problems to measure participants' skill in problem solving. It was predicted that subgoal labeled worked examples would lead to better performance than unlabeled examples. This effect is well-documented in the literature (e.g., Catrambone, 1998). It was also predicted that subgoal labeled expository text would not independently lead to better performance compared to unlabeled expository text. It was expected that if participants received only subgoal labeled expository text that abstractly describes procedures for problem solving, they would have trouble applying these concepts to solving problems. However, it was predicted that subgoal labeled expository text would improve performance when paired with subgoal labeled worked examples. It was

expected that subgoal labeled expository text would enable learners to identify subgoals earlier and organize procedural information better, which would improve problem solving performance when paired with subgoal labeled examples that demonstrated how to apply this knowledge. Furthermore, it was expected that subgoal labels on both expository text and examples would facilitate the integration of information in the expository text and examples by matching the information grouped under matching subgoal labels. For this reason, it was also predicted that learners who received subgoal labeled expository text and examples would perform better than those who received subgoal labeled examples without subgoal labeled expository text.

Patterns of performance across groups were not expected to differ between experiments. For Experiment 1 in which learners were not allowed to view the instructions during the problem solving assessment tasks, receiving subgoal labels in both types of instruction was expected to improve organization and memory of information by guiding attention to important details of the instructions and making connections between the expository text and worked example explicit. For Experiment 2 in which learners were allowed to view the instructions during problem solving, receiving subgoal labels in both types of instruction was expected to improve learners' use of the instructions as a resource during problem solving by guiding attention and making connections.

2. Experiment 1

2.1 Method

2.1.1 Participants. Participants were 120 students from a mid-sized, urban university who received class credit for participation. Participants must not have had experience with App Inventor or taken more than one course either before or during college in computer science or programming. These restrictions were necessary to ensure participants had about the same level

of prior knowledge about programming because instructions were designed for novices. Previous

experiments using similar instructional materials found no statistical differences in scores

between participants who had taken one course in computer science and those who had taken

none (Margulieux et al., 2012). On average, participants were 19.3 years old, had completed 2.1

years of college, and were comfortable with computers ($M = 6.0$ on a scale of "1 – Not at all

comfortable" to "7 – Very comfortable"). The majority (67%) of participants were men.

      2.1.2 Design. The experiment was a two-by-two, between-subjects, factorial design: the

format of the expository text (subgoal labeled or unlabeled) was crossed with the format of the

worked example (subgoal labeled or unlabeled). In each of the conditions, participants received

the same instructional content, but the presence of subgoal labels differed. The dependent

variables were performance on the assessment tasks (to determine participants' knowledge

organization and effectiveness in solving problems), minutes spent completing the assessment

tasks (to determine participants' efficiency), and minutes spent looking at instructions.

      2.1.3 Materials. Participants learned the procedure for creating apps with three types of

instruction. The first, expository text, gave participants general, conceptual information about the

procedure. These instructions were displayed as a PDF on a computer screen. The second, a

video that showed the Fortune Teller being created with a voice over describing the process,

gave participants a specific worked example of the procedure being completed. The third, a

textual step-by-step guide to create the Fortune Teller app, gave participants step-based

instructions to complete the Fortune Teller app themselves. These instructions were also

displayed as a PDF on a computer screen. Examples of subgoal labeled and unlabeled materials

are in Figures 1 and 3. For participants who received the subgoal labeled worked example, the

video presented subgoal labels in pop-up text boxes while the subgoal was being achieved. The

pop-up text did not obscure the part of the interface that was being used (see Figure 5). Participants who received subgoal labels in the video also received subgoal labels in the textual step-by-step guide.

The assessment tasks measured problem solving performance with a series of problem solving tasks and measured mental organization of information with an explanation task and a generalization task. Participant responses were scored by two raters. Interrater reliability was measured with an intraclass correlation coefficient of absolute agreement (ICC(A)) because the absolute values of continuous scores were compared (Shrout & Fleiss, 1979). Raters were undergraduate and graduate research assistants who had taken a research methodology course and were trained on the protocol and scoring by the investigators. During scoring, raters were blind to the participants' conditions to avoid bias.

The first assessment, the series of problem solving tasks, asked participants to list the steps that they would take to make four features of an app (e.g., "Write the steps you would take to italicize the fortune presented," or "Write the steps you would take to create a list of colors and make a ball change to a random color whenever it collided with something"). This assessment was meant to measure how well participants could solve novel problems.

To score this assessment, participants' solutions were compared to the correct solutions for each problem. Participants earned one point for each correct step that they took towards the problem solution. To complete all four features correctly, 22 steps needed to be taken; therefore, the maximum score was 22. Because the solutions are complex, this scoring scheme afforded more sensitivity than judging an entire solution as correct or incorrect. ICC(A) for this assessment was .94.

To better understand participants' performance, the problem solving tasks were also scored in terms of how much of the solution participants attempted. This score is meant to measure how many functional components, or subgoals, of the solution the participants attempted, regardless of whether their answers were correct. Attempting subgoals of the solution would suggest that a participant recognized the components needed in the solution, even if they could not achieve them. To create an "attempted" score, the correct solutions for the problem solving tasks were deconstructed into the subgoals that were necessary to complete the solution (e.g., "create a component" or "set the output"). Participant solutions earned a point for each subgoal that was attempted. Attempting a subgoal was operationally defined as listing at least one step required to complete it, listing a step that would achieve a similar function (e.g., listing a step to change a property regardless of whether it was the correct property), or describing the function of the subgoal. In the four features created in the problem solving assessment, there were 10 subgoals; therefore, the maximum score was 10. ICC(A) for this assessment was .95.

In the explanation task, participants were given an expert's solutions for the problem solving tasks in the first assessment and asked to group steps of the solutions however they thought apt. Then, participants were asked to provide a label describing the purpose of each group. This assessment was meant to measure how well participants could explain solutions. To do well on this assessment, participants did not need to solve problems, but they did need to recognize the steps of the solution that were structurally-related and explain why they were related. Participants received two scores for this assessment: a grouping score for how well they organized steps and a labeling score for how well they explained groups. To score the grouping portion of this task, participants received points for grouping together structurally-related steps (i.e., steps that achieve a function). For each group that contained only structurally-related steps,

participants received one point, and they could earn up to 10 points. To score the labeling portion of this task, the labels that participants used to describe the groups were analyzed qualitatively using the methods from Taylor-Powell and Renner (2003) to determine if participants correctly identified the purpose of the chunks. ICC(A) was .97.

The generalization assessment asked participants to describe the general approach that they would take to create an app that provided an output when a button was pressed. This assessment was meant to measure how well participants could abstractly describe the problem solving procedure that they learned in the session. To score the generalization task, participants received a point for each structural feature that they described that was necessary for creating the app. Participants did not receive points for specific descriptions or unnecessary features. Specific descriptions included information about how to achieve a step using the interface or specified a particular block to be used. The maximum score on this assessment was six. The ICC(A) was .89.

2.1.4 Procedure. Sessions were between 70 and 90 minutes depending on how quickly participants completed the protocol. All instruction about programming was given through a computer (one participant per computer). This computer-based learning environment was chosen to isolate the effect of the instructional materials and reduce possible bias from an instructor. During the sessions, experimenters answered questions about the study (e.g., "When will the credit show up in my account?" or "Can I watch the video again?") but did not answer questions about the instructional materials or App Inventor (e.g., "How do I make a button?" or "Where do I find the text blocks?").

Participants first filled out a demographic questionnaire to provide information about their age, gender, field of study, SAT scores, high school and college GPA, year in school,

number of completed credits, primary language, computer science experience, comfort with computers, and expected difficulty of learning App Inventor. These factors were collected because they are possible predictors of performance in computer science (Rountree, Rountree, Robins, & Hannah, 2004).

The instructional period was next. During this time, participants received expository text about creating apps generally and the video (worked example) demonstrating how to create the Fortune Teller app. To give participants a chance to practice using the interface and the procedure, participants were asked to create the Fortune Teller app by following a textual step-by-step guide. Participants had up to 30 minutes to create the app using the App Inventor website and the instructions in whatever way they wanted.

Next was the assessment period. During the assessment tasks, participants could not access the instructional materials, but they could access the App Inventor website and the app that they had created during the instructional period to serve as a memory cue to aid problem solving. The assessment tasks included the 1) series of problem solving tasks, 2) explanation task, and 3) generalization task. For the series of problem solving tasks, participants were limited to a maximum of 25 minutes. Thus, like an exam, participants were not permitted to work on the problems for an unlimited amount of time. They were not limited on time for the explanation and generalization tasks.

2.2 Results and Discussion

Of the demographic factors collected as possible predictors, two were correlated with performance. There was a negative correlation between SAT Writing scores and time spent on the instructional period, $r = -.28$, $p = .022$. Participants' with higher SAT Writing scores tended to finish the instructional period faster. There was a positive correlation between participants'

subjective ratings of their comfort with computers and number of attempted problem solutions, r = .25, p = .009. These predictors were not expected to confound the analyses of the performance metrics because there were no differences among experimental conditions on these predictors (see Table 1), and, therefore, no group would have an advantage.

2.2.1 Time on instruction. Participants who received the subgoal labeled example (M = 20.9, SD = 3.26) finished the instructional period 12% faster than those who did not (M = 23.7, SD = 4.69), F (1, 116) = 12.62, MSE = 16.83, p < .001, est. ω^2 = .10, f = .32. Estimated ω^2 represents the proportion of variance that is accounted for by the intervention; in this case, 10% of variance is attributed to the intervention. The effect size f represents the size of the difference between groups in units of standard deviation; in this case, there is a difference of .32 standard deviations between groups. This effect could be the result of the subgoal labels in examples helping participants to group or chunk the steps of the step-by-step guide. Chunking steps could have helped participants to remember more steps to complete in the App Inventor interface before referring back to the guide, and the labels could also have helped participants find their spot in the guide faster when they did refer to it. There was no significant difference between text designs, F (1, 116) = .25, MSE = 16.83, p = .62, and there was no interaction of text and example design, F (1, 116) = .69, MSE = 16.83, p = .69.

2.2.2 Problem solving performance. This task measured participants' problem solving performance. Participants could earn a maximum score of 22. Participants who received subgoal labels in the example (M = 13.1, SD = 6.0) successfully completed 58% more steps than those who did not (M = 5.5, SD = 4.8), F (1, 116) = 70.19, MSE = 24.47, p < .001, est. ω^2 = .32, f = .76. In addition, participants who received subgoal labels in the text (M = 11.0, SD = 7.1) completed 31% more steps than those who did not (M = 7.6, SD = 5.7), F (1, 116) = 13.90, MSE

$= 24.47, p < .001$, est. $\omega^2 = .06, f = .34$. Moreover, there was an interaction between text and example design, $F(1, 116) = 12.82$, $MSE = 24.47$, $p = .001$, est. $\omega^2 = .05, f = .57$. The interaction shows that participants who received subgoal labels in the text performed better than those who did not only when they also received subgoal labels in the example (see Figure 6). This pattern suggests that the interaction caused the main effect of text by inflating the average for subgoal labeled text. Closer evaluation showed that there was not a consistent simple effect of text design meaning subgoal labels in the text did not independently improve performance (see Table 2).

Several studies (e.g., Atkinson et al. 2003; Catrambone, 1998; Renkl, 2002), including a study using similar instructional materials (Margulieux et al., 2012), have demonstrated that subgoal labeled worked examples help participants learn procedures in a way that allows them to transfer their knowledge to solve novel problems. The primary explanation for this effect is that subgoal labeled examples might help participants to learn the subgoals necessary to solve a class of problems in a domain, which improves their problem solving performance (Catrambone, 1998).

Despite the benefits of subgoal labels in worked examples, a simple effect of subgoal labels in expository text was not found. Perhaps because learners in procedural domains rely on worked examples to show how to apply domain knowledge to problem solving (e.g., LeFevre & Dixon, 1986), subgoal labeled expository text might not have provided enough information to help students apply subgoals to problem solving. However, the interaction between text design and example design demonstrates that subgoal labeled text can improve problem solving performance when paired with subgoal labeled examples.

These results were expected for two reasons. First, having subgoal labels in both types of instructional material could have helped participants integrate the general information in the text

with the specific information in the example, leading to a better representation of the procedure. Additionally, receiving the subgoal labeled text, similar to receiving principles in text (Bassok & Holyoak, 1989), might have helped participants organize components of the expository text better. Better organization of the expository text could lead to more effective processing of the example when the same labels were used in both.

2.2.2.1 Attempted problem solutions. Participants' performance on the problem solving tasks was also measured by number of components of the solution that they attempted, even if those components were not correct. Participants could earn a maximum score of 10. Participants who received the subgoal labeled example ($M = 6.9$, $SD = 2.7$) attempted 41% more components than those who did not ($M = 4.1$, $SD = 2.8$), $F(1, 116) = 30.43$, $MSE = 7.73$, $p < .001$, est. $\omega^2 = .20$, $f = .50$. There was no significant difference between text designs, $F(1, 116) = .52$, $MSE = 7.73$, $p = .47$, and there was no interaction of text and example design, $F(1, 116) = .00$, $MSE = 7.73$, $p = .96$. These results suggest that participants who received the subgoal labeled example recognized the necessary components of the task solutions better than those who did not, regardless of whether they could solve the problem correctly. These results, in conjunction with problem solving performance, suggest that the subgoal labeled text did not enable participants to attempt more components but, when paired with the subgoal labeled example, helped them correctly achieve more of their attempted components.

2.2.2.2 Time on task. The amount of the time that participants spent working on the problem solving tasks was also measured. The majority of participants (75%) used the entire 25 minutes, but despite this range restriction, there were main effects of text and example design for time on task. Participants who received the subgoal labeled example ($M = 20.5$ minutes, $SD = 3.0$) completed the task 6% faster than those who did not ($M = 21.7$, $SD = 2.8$), $F(1, 116) = 7.88$,

$MSE = 7.84$, $p = .006$, est. $\omega^2 = .06$, $f = .26$. Additionally, participants who received the subgoal labeled text ($M = 20.25$, $SD = 2.2$) completed the task 7% faster than those who did not ($M = 21.8$, $SD = 3.4$), $F(1, 116) = 9.19$, $MSE = 7.84$, $p = .003$, est. $\omega^2 = .07$, $f = .28$ (see Figure 7). There was no interaction of text and example design, $F(1, 116) = .15$, $MSE = 7.84$, $p = .70$. These findings suggest that receiving subgoal labels in instructional materials reduced time on the task. Moreover, when paired with the performance results, these findings show that participants who performed better also completed the problems faster. These results are contrary to the typical tradeoff between speed and accuracy and suggest that participants who received subgoal labels in both the text and example were more successful and quicker at solving problems compared to those who did not.

The findings from the problem solving assessment provide two important pieces of information about subgoal labeled instructional materials. First, they demonstrate that for subgoal labeled text to improve performance, it needs to be paired with subgoal labeled examples. Second, the results show that subgoal labels can lead to better problem solving when the labels appear in both the example and text than when subgoal labels appear only in examples. It is possible that receiving more instantiations of each subgoal label, whether in the text or in additional subgoal labeled examples, would allow learners to gather more information about each subgoal, refine their procedural rules, and solve problems better. Though this possibility is not directly explored in the present study, viewing more examples might be less efficient than viewing expository text, especially for topics in which the examples are long like programming. Furthermore, the results from other tasks suggest that subgoal labels have a different effect on learners' mental representations of the procedure when presented in expository text than when presented in worked examples.

2.2.3 Explanation task. The grouping part of the explanation task measured how well participants organized steps of problem solutions into functional components. Participants could earn a total of 10 points, 1 point for each component in the problem solutions. Compared to other conditions, participants who received subgoal labels in both the text and example made significantly more groups that included only structurally-related steps. There was no significant difference between text designs, F (1, 116) = 3.60, MSE = 4.69, p = .06, and there was no significant difference between example designs, F (1, 116) = 3.93, MSE = 4.69, p = .05. There was, however, an interaction between text and example design, F (1, 116) = 3.99, MSE = 4.69, p = .047, est. ω^2 = .04, f = .18 (see Table 3 for simple main effects). To perform well on this task, participants needed to integrate procedural knowledge (to identify functional groups) and application knowledge (to apply the groups to specific problems), and subgoal labels in both types of instructional material might have aided this integration.

The labeling part of this task measured how well participants could describe the function of the groups. Over 50% of the responses given by participants who received the subgoal labeled text correctly described the function of a group of steps. In contrast, less than 10% of the responses given by participants who received the unlabeled text correctly described the function. There was no meaningful difference for example design. Both subgoal labeled and unlabeled example groups produced 30% correct functional descriptions. The content of incorrect responses included superficial information such as how the blocks of code were placed together or where in the interface the steps were completed.

For time on task, the only statistically significant difference between conditions was within the group that did not receive the subgoal labeled text. People who received subgoal labels only in the worked example (M = 4.7, SD = 2.1) completed the explanation task

significantly faster than people who did not receive subgoal labels in any of the instructional

materials ($M = 7.0$, $SD = 3.1$), $t(58) = -3.06$, $p = .004$ (see Figure 8 and Table 4 for full pattern

of results). These results indicate that people who performed best on this task (i.e., those who

received subgoal labels in both the text and example) did not take longer to group and label the

solutions with more accuracy than it took the other participants to group and label the solutions

with less accuracy. This pattern of results suggests that subgoal labels in the text made

participants more likely to articulate the purpose of groups of steps, perhaps by encouraging self-

explanation (Chi, 2009; Hill & Levenhagen, 1995).

2.2.4 Generalization task. The generalization task measured how well participants could

create a conceptual description of the procedure. Participants could earn a maximum score of six,

one point for each component of the solution. Participants who received the subgoal labeled text

($M = 4.4$, $SD = 1.1$) correctly described 20% more components than those who did not ($M = 3.5$,

$SD = 1.3$), $F(1, 116) = 15.11$, $MSE = 1.49$, $p < .001$, est. $\omega^2 = .10$, $f = .35$. There was no

significant difference between example designs, $F(1, 116) = 2.70$, $MSE = 1.49$, $p = .10$, and

there was no interaction, $F(1, 116) = .20$, $MSE = 1.49$, $p = .66$. These results are consistent with

the explanation task in that subgoal labels in the text seem to have aided articulation of the

abstract procedure. This result might be due to subgoal labels signaling the components in the

instructions that abstractly described the procedure.

For time on task in this assessment, there was also a significant difference between text

designs. Participants who received the subgoal labeled text ($M = 3.9$, $SD = 2.3$) took 23% longer

to complete the task than those who did not ($M = 3.0$, $SD = 1.20$), $F(1, 116) = 5.95$, $MSE = 3.48$,

$p = .016$, est $\omega^2 = .05$, $f = .22$. There was no significant difference between example designs, F

$(1, 116) = .83$, $MSE = 3.48$, $p = .36$, and there was no interaction of text and example design, F

$(1, 116) = 1.65$, $MSE = 3.48$, $p = .20$. In this case, people who performed better took longer to complete the task.

Because responses were written, it could be the case that those who performed well needed more time to write their responses than those who performed poorly because the former group wrote more than the latter group. To explore this possibility, the number of words in each response was counted to estimate the time participants spent writing. Participants who received subgoals in the text wrote 18% ($M = 40$ words) more words on average than other participants ($M = 34$ words). This finding suggests that part of the difference in time on task between these groups is due to time spent writing.

In summary, Experiment 1 explored the efficacy of subgoal labeled expository text to teach a programming task. The results suggest that subgoal labeled text helped learners to solve novel problems when paired with subgoal labeled worked examples. Subgoal labeled expository text also seemed to help learners explain a procedure. Experiment 2 continued this exploration in a different learning scenario. During Experiment 2, participants could use instruction as a resource during the problem solving tasks.

3. Experiment 2

3.1 Method

3.1.1 Participants. Participants were 120 students from a mid-sized, urban university who received class credit for participation. Participants must not have participated in Experiment 1, interacted with App Inventor, or taken more than one course (either before or during college) in computer science or programming. On average, participants were 20.3 years old, had completed 2.5 years of college, and comfortable with computers ($M = 5.9$ on a scale of "1 – Not at all comfortable" to "7 – Very comfortable"). The majority (61%) of participants were men.

3.1.2 Procedure. The procedure and design for Experiment 2 was the same as for Experiment 1. The only differences were that participants could use instructional materials during the assessment period, and the assessment period included only the problem solving tasks. The other tasks used in Experiment 1 were meant to measure mental organization and conceptual understanding of the procedure; therefore, allowing participants to reference instructions would likely lead to participants repeating the instructions on the assessments rather than recording their mental representations. For this reason, these tasks were excluded.

3.2 Results and Discussion

Of the demographic information collected as possible predictors, two were correlated positively with performance on the problem solving tasks: high school GPA, $r = .30$, $p < .01$, and number of college credits completed, $r = .25$, $p = .01$. These predictors were not expected to confound the analyses of the performance metrics because the variance was evenly distributed among groups (see Table 5), and, therefore, no group had an advantage.

3.2.1 Time on instruction. The amount of time that participants spent using instructional materials in the instructional period was recorded. There was a main effect of text design: participants who received the subgoal labeled text ($M = 19.5$, $SD = 3.01$) finished the instructional period 6% faster than those who did not ($M = 20.8$, $SD = 3.3$), $F (1, 116) = 4.36$, $MSE = 9.79$, $p = .038$, est. $\omega^2 = .09$, $f = .19$. There was no main effect for example design, $F (1, 116) = .10$, $MSE = 9.79$, $p = .76$. There was an interaction of text and example design, $F (1, 116) = 4.62$, $MSE = 9.79$, $p = .034$, est. $\omega^2 = .09$, $f = .20$. The interaction suggests that participants who received subgoal labels in both the text and example finished instructions faster than the others. This pattern suggests that the interaction caused the main effect of text by reducing the average for the subgoal labeled text condition. Closer evaluation showed that there was not a

consistent simple effect of text design, indicating that subgoal labels in the text did not independently decrease instruction time (see Table 6). If subgoal labels in text helped participants more effectively process worked examples, then perhaps when the examples had subgoal labels, too, participants could process information more quickly than the other groups.

3.2.2 Problem solving performance. This assessment and scoring was the same as in Experiment 1. The maximum score was 22. There was an interaction between text and example design, F (1, 116) = 5.87, MSE = 24.26, p = .017, est. ω^2 = .07, f = .22. This interaction shows that participants who received subgoal labels in the text and example outperformed all other groups. There were no other significant differences (see Figure 9 and Table 7).

When participants were allowed to reference instructional materials, receiving subgoal labels in only examples no longer improved problem solving. This finding is not surprising because the worked example was the same except for the subgoal labels. If the effect of subgoal labeled examples is helping students decontextualize examples, and if learners have access to context-independent expository text while solving problems, then subgoal labels in the examples would not be expected to cause a difference.

Receiving subgoal labels in both expository text and worked examples improved problem solving even when participants were allowed to reference the instructions during problem solving. Perhaps when subgoal labels were in both instructional materials, in addition to previously discussed benefits, they helped participants find information to resolve specific problem solving impasses. VanLehn et al. (1992) found that when participants had trouble with a problem and consulted the expository text, many spent a long time searching the text, but only a small proportion found relevant information. Subgoal labels might have helped students who are struggling with a problem to find relevant information more quickly. For example, if a

participant was stuck on a part of a problem, he or she might consult the worked example to find

analogous information. If that did not resolve the issue, then he or she might consult the

expository text for information. If both of these materials had the same subgoal labels, then

finding information could be easier.

   3.2.2.1 Attempted problem solutions. This score was calculated using the same method

as in Experiment 1. The maximum score was 10. There was no main effect of example design, F

$(1, 116) = 2.70$, $MSE = 6.35$, $p = .10$, no main effect of text design, $F (1, 116) = 2.21$, $MSE =$

6.35, $p = .14$, and no interaction, $F (1, 116) = 1.40$, $MSE = 6.35$, $p = .24$. These findings were

expected because participants were allowed to use the instructional materials during problem

solving and the instructions were the same except for subgoal labels. All participants were

equally unlikely to overlook components of the procedure.

   3.2.2.2 Time on task. The amount of the time that participants spent working on the

problem solving tasks was also measured. There was a main effect of text design for time on

task. Participants who received the subgoal labeled text ($M = 17.8$, $SD = 4.57$) completed the

task 8% faster than those who did not ($M = 19.3$, $SD = 3.00$), $F (1, 116) = 4.42$, $MSE = 14.2$, $p =$

$.038$, est. $\omega^2 = .09$, $f = .19$. There was no main effect for example design, $F (1, 116) = .00$, $MSE =$

14.2, $p = .99$. The main effect of text is largely irrelevant, however, due to the interaction of text

and example design, $F (1, 116) = 5.88$, $MSE = 14.2$, $p = .017$, est. $\omega^2 = .09$, $f = .22$ (see Figure

10).

   Though the effect sizes for these findings are relatively small, the pattern of results is

interesting. Participants who received subgoal labels in the example took about the same time to

finish the task regardless of whether they received subgoal labels in the text ($M = 18.4$ for

unlabeled text versus $M = 18.6$ for labeled text). For participants who did not receive subgoal

labels in the example, time on task depended on whether they received subgoal labels in the text (M = 20.2 for unlabeled text versus M = 16.8 for labeled text). The program that displayed instructional materials also logged which instructions participants viewed during the assessment period. Most participants who received subgoal labels in the example (about 75%), regardless of whether they received subgoal labels in the text, viewed only the example while working on the problem solving tasks. Of participants who did not receive subgoal labels in the example, most (91%) viewed both the expository text and example while working on the problem solving tasks.

These findings suggest that participants who received subgoal labels in the example tended to ignore the expository text while solving problems. In addition, participants who received subgoal labels in the text and example solved problems better than those who did not, even though they took the same amount of time, on average, to complete the problem solving tasks. Perhaps participants who received labels in both the text and example were more successful than other participants at connecting information from the example and the expository text. As a result, during problem solving they could refer back just to the example to successfully solve novel problems.

For participants who did not receive subgoal labels in the example, they tended to view both the expository text and worked example while solving problems. Participants who received labels in the text performed about the same as those who did not, but they took significantly less time to do so. Perhaps the subgoal labeled text allowed these participants to identify the necessary components of the solution more quickly, but because they did not receive labels in the example as well, they were not able to make those components as effectively.

4. Conclusion

Guided instruction is important for novices because it helps them organize and use new information more effectively (e.g., Kirschner, Sweller, & Clark, 2006; Van Gog et al., 2006). The present study explored a new method for providing guided instruction: subgoal labeled expository text. The findings provide three important pieces of information about subgoal labeled instructional materials:

- Subgoal labeled expository text improved performance only when paired with subgoal labeled examples.

- Subgoal labeled expository text helped learners explain procedures while subgoal labeled examples helped learners apply procedures.

- Subgoal labels led to better problem solving when the labels appeared in both examples and text than when subgoal labels appeared in examples alone.

Participants who received subgoal labels in both the text and example outperformed those in other conditions. This effect was expected two reasons. First, when learners receive multiple representations of content (e.g., text and example), features that help them translate between those representations leads to better integration and understanding of the information (Ainsworth, 2006). Subgoal labels might have helped learners translate between the two types of instructional materials. Second, receiving the subgoal labeled text, similar to receiving signals in instructions (Lemarié et al., 2008), might help learners organize information about the expository text better. Better organization of the information in the expository text might lead to more effective processing of examples that use the same labels.

The results from the explanation and generalization tasks in Experiment 1 suggest that subgoal labels in the text led to different benefits than subgoal labels in the example. Both the generalization task and labeling portion of the explanation task required participants to articulate

their knowledge of the procedure. In both cases, participants who received subgoal labels in the text outperformed those who did not.

To speculate on how subgoal labeled instructional materials affect learning, the model in Figure 2 that describes how subgoal labeled worked examples improve problem solving was expanded. Figure 11 shows a proposed model for how subgoal labels in text and examples jointly improve problem solving. The results of the present study provide some preliminary evidence to support the benefits of subgoal labeled expository text and the combination of subgoal labels in both text and examples for problem solving. Evidence from other researchers supports some of the connections in this proposed model. The results from Chi (2009) and Hill and Levenhagen (1995) support the claim that self-explanations improve articulation and that improved articulation can improve problem solving. Eiriksdottir and Catrambone (2011) argue that integrating information from general and specific instructions can improve the application of general knowledge and transfer of specific knowledge. Lemarié et al. (2008) argue that meaningful labels of text improves organization of information. More research is needed, however, to systematically test the model in Figure 11.

4.1 Application of Subgoal Labels

The subgoal intervention manipulates the instructional materials that students receive; therefore, distributing the intervention would be relatively easy. Furthermore, because the interventions are not reliant on instructors, they can be used in a range of learning environments including face-to-face and online learning. This study did not explore the efficacy of this manipulation in a learning environment with an instructor, but we hypothesize it could improve learning in that situation. Instructors, as experts, sometimes do not realize how to help learners form useful knowledge representations, partly because much of the instructor's procedural

knowledge has become automated. Using subgoal labeled materials would ensure that students received the fundamental knowledge that they needed to conceptually understand procedures.

Subgoal labeled text could also help learners communicate better, especially when they are not face-to-face, by improving their articulation of the procedure. Students in online learning environments often communicate less effectively than in face-to-face environments because they are often forced to express themselves primarily verbally rather than relying on other types of communication, such as drawing (Gecer, 2013). The present results show that people who receive subgoal labeled text are able to articulate the problem solving process better than those who do not, which could result in more effective verbal communication.

The present research advances knowledge about strategies for improving novice problem solving in programming. Subgoal labeled worked examples have already been shown to significantly increase learners' problem solving performance (Catrambone, 1998). The present study demonstrated that subgoal labeled expository text can increase this effect and improve other types of performance. This study suggests that subgoal labels should be used in both expository text and worked examples designed to teach problem solving procedures.

References

Ainsworth, S. (2006). DeFT: A conceptual framework for considering learning with multiple representations. *Learning and Instruction, 16*, 183-198. doi:10.1016/j.learninstruc. 2006.03.001

Anderson, J. Farrell, R. & Sauers, R. (1984). Learning to program in LISP. *Cognitive Science, 3*(2), 87-129. doi: 10.1207/s15516709cog0802_1

Atkinson, R. K., Catrambone, R., & Merrill, M. M. (2003). Aiding transfer in statistics: Examining the use of conceptually oriented equations and elaborations during subgoal learning. *Journal of Educational Psychology, 95*(4), 762-773. doi: 10.1037/0022-0663.95.4.762

Atkinson, R. K., & Derry, S. (2000). Computer-based examples designed to encourage optimal example processing: A study examining the impact of sequentially presented, subgoal-oriented worked examples. In B. Fishman & S. O'Connor-Divelbiss (Eds.), *Fourth International Conference of the Learning Sciences*, (pp. 132-133), Mahwah, NJ: Earlbaum.

Atkinson, R. K., Derry, S. J., Renkl, A., & Wortham, D. (2000). Learning from examples: Instructional principles from the worked examples research. *Review of the Educational Research, 70*(2), 181-214. doi: 10.2307/1170661

Bassok, M., & Holyoak, K. (1989). Interdomain transfer between isomorphic topics in algebra and physics. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *15*(1), 153-166. doi: 10.1037/0278-7393.15.1.153

Biederman, I., & Shiffrar, M. (1987). Sexing day-old chicks: A case study and expert systems analysis of a difficult perceptual-learning task. *Journal of Experimental*

Psychology: Learning, Memory, and Cognition, *13*(4), 640-645. doi:

> 10.1037/0278-7393.13.4.640

Bransford, J. D., Brown, A. L., & Cocking, R. R. (Eds.) (2000). How People Learn: Brain, Mind,

> Experience, and School. Washington, DC: National Academy Press.

Catrambone, R. (1994). Improving examples to improve transfer to novel problems.

> *Memory and Cognition, 22*, 605-615. doi: 10.3758/BF03198399

Catrambone, R. (1995a). Aiding subgoal learning: Effects on transfer. *Journal of

> Educational Psychology, 87*(1), 5-17. doi: 10.1037/0022-0663.87.1.5

Catrambone, R. (1995b). Following instructions: Effects of principles and examples.

> *Journal of Experimental Psychology: Applied, 1*(3), 227-244. doi: 10.1037/1076-

> 898X.1.3.227

Catrambone, R. (1996). Generalizing solution procedures learned from examples.

> *Journal of Experimental Psychology: Learning, Memory, and Cognition, 22*,

> 1020-1031. doi: 10.1037/0278-7393.22.4.1020

Catrambone, R. (1998). The subgoal learning model: Creating better examples so that

> students can solve novel problems. *Journal of Experimental Psychology: General,

> 127,* 355-376. doi: 10.1037/0096-3445.127.4.355

Catrambone, R., Gane, B. D., Adams, A. E., Bujak, K. R., Kline, K. A., & Eiriksdottir, E.

> (2014). Task Analysis by Problem Solving (TAPS): A method for uncovering

> expert knowledge. *Under review*.

Catrambone, R., & Holyoak, K. (1990). Learning subgoals and methods for solving

> probability problems. *Memory & Cognition*, *18*(6), 593-603. doi: 10.1037/0096-

> 3445.127.4.355

Chi. M. T. H. (2009). Active-constructive-interactive: A conceptual framework for

differentiating learning activities. *Topics in Cognitive Science, 1*(1), 73-105.

Chi, M. T. H., Leeuw, N. D., Chiu, M., LaVancher, C. (1994). Eliciting self-explanations

improves understanding. *Cognitive Science, 18,* 439-477.

Committee on Developments in the Science of Learning, National Research Council.

(2000). *How people learn: Brain, mind, experience, and school: Expanded

edition.* Retrieved from http://www.nap. edu/catalog.php?record_id=9853

Committee on Highly Successful Schools or Programs in K-12 STEM Education, National

Research Council. (2011). *Successful K-12 STEM education: Identifying effective

approaches in science, technology, engineering, and mathematics.* Retrieved from

http://www.nap.edu/catalog.php?record_id=13158

Cooper, S., Grover, S., Guzdial, M., & Simon, B. (2014). A future for computing education

research. *Communications of the ACM, 57*(11), 34-36.

Eiriksdottir, E., & Catrambone, R. (2011). Procedural instructions, principles, and examples:

How to structure instructions for procedural tasks to enhance performance, learning, and

transfer. *Human Factors, 53*(6), 749-770. doi: 10.1177/0018720811419154

Ericson, B. (2012, February 12). ICE Distance Education Portal. Retrieved from

http://ice.cc.gatech.edu/dl/?q=node/641

Gecer, A. (2013). Lecturer-student communication in blended learning environments.

Educational Sciences: Theory and Practice, 13(1), 362-367.

Hill, R. C., & Levenhagen, M. (1995). Metaphors and mental models: Sensemaking and

sensegiving in innovative and entrepreneurial activities. *Journal of Management, 21*(6),

1057-1074.

Hundhausen, C. D., Farley, S. F., & Brown, J. L. (2009). Can direct manipulation lower the

    barriers to computer programming and promote transfer of training?: An experimental

    study. *ACM Transactions in CHI, 16*(3). doi: 10.1145/1592440.1592442

Katehi, L., Pearson, G., & Feder, M. (Eds.), Committee on K-12 Engineering Education,

    National Academy of Engineering and National Research Council. (2009). *Engineering*

    in K-12 education: Understanding the status and improving the prospects. Retrieved

    from http://www. nap.edu/catalog.php?record_id=12635

Kirschner, P., Sweller, J., & Clark, R. (2006). Why minimal guidance during instruction

    does not work: An analysis of the failure of constructivist, discovery, problem-

    based, experiential, and inquiry-based teaching. *Educational Psychologist, 41*(2),

    75-86. doi: 10.1207/s15326985ep4102_1

LeFevre, J. & Dixon, P. (1986). Do written instructions need examples? *Cognition and*

    Instruction, 3, l-30. doi: 10.1207/s1532690xci0301_1

Lemarié, J., Lorch Jr., R. F., Eyrolle, H., & Virbel, J. (2008). SARA: A text-based and reader-

    based theory of signaling. *Educational Psychologist, 43*, 27-48

Margulieux, L. E., Guzdial, M., & Catrambone, R. (2012). Subgoal-labeled instructional material

    improves performance and transfer in learning to develop mobile applications.

    Proceedings of the Ninth Annual International Conference on International Computing

    Education Research (pp. 71-78). New York, NY: Association for Computing Machinery.

    doi: 10.1145/2361276.2361291

McGee, P., & Reis, A. (2012). Blended course design: A synthesis of best practices.

    Journal of Asynchronous Learning Networks, 16(4), 7-22.

Miller, A. D. (2014). Computer science: The future of education. Retrieved from

    http://www.edutopia.org/blog/computer-science-future-of-education-alison-

    derbenwick-miller

Mayer, R. E. (2009). Multimedia learning (2nd). New York: Cambridge University Press.

Palmiter, S., Elkerton, J., & Baggett, P. (1991). Animated demonstrations versus written

    instructions for learning procedural tasks: A preliminary investigation. *International*

    Journal of Man-Machine Studies, 34, 687-701. doi: 10.1016/0020-7373(91)90019-4

Reder, L. M., & Anderson, J. R. (1980). A comparison of texts and their summaries:

    Memorial consequences. *Journal of Verbal Learning and Verbal Behavior, 19*,

    121-134. doi: 10.1016/S0022-5371(80)90122-X

Renkl, A. (2002). Worked-out examples: Instructional explanations support learning by

    self-explanations. *Learning and Instruction*, *12*, 529-556. doi: 10.1016/S0959-

    4752(01)00030-5

Renkl, A., & Atkinson, R. K. (2002). Learning from examples: Fostering self-

    explanations in computer-based learning environments. *Interactive Learning*

    Environments, 10(2), 105-199. doi: 10.1076/ilee.10.2.105.7441

Rountree, N., Rountree, J., Robins, A., & Hannah, R. (2004). Interacting factors that

    predict success and failure in a CSI course. *SIGCSE Bulletin, 33*(4), pp. 101-104.

Shrout, P. E., & Fleiss, J. L. (1979). Intraclass correlations: Uses in assessing rater

    reliability. *Psychological Bulletin, 86*(2), 420-428.

Smith, E. E., & Goodman, L. (1984). Understanding written instructions: The role of an

    explanatory schema. *Cognition and Instruction, 1*, 359-396. doi: 10.1207

    /s1532690xci0104_1

Sweller, J. (2010). Element interactivity and intrinsic, extraneous, and germane cognitive load. *Educational Psychology Review*, *22*(2), 123-138. doi: 10.1007/s10648-010-9128-5

Taylor-Powell, E., & Renner, M. (2003). *Analyzing qualitative data.* University of Wisconsin, Cooperative Extension.

Trafton, J. G., & Reiser, B. J. (1993). The contributions of studying examples and solving problems to skill acquisition. In *Proceedings of the Fifteenth Annual Conference of the Cognitive Science Society* (pp. 1017-1022). Boulder, CO.

Van Gog, T., Paas, F., & Van Merriënboer, J. J. G. (2004). Process-oriented worked examples: improving transfer performance through enhanced understanding. *Instructional Science, 32*(1-2), 83-98.

Van Gog, T., Paas, F., & Van Merriënboer, J. J. G. (2006). Effects of process-oriented worked examples on troubleshooting transfer performance. *Learning and Instruction, 16*(2), 154-164.

VanLehn, K., Jones, R., & Chi, M. T. H. (1992). A model of the self-explanation effect. *The Journal of the Learning Sciences, 2,* 1-59. doi: 10.1207/s15327809jls0201_1

Wylie, R., & Chi, M. T. H. (2014). The self-explanation principle in multimedia learning. In R. Mayer (Ed.) *The Cambridge Handbook of Multimedia Learning, 2nd Edition (pp.413-432).* Cambridge University Press.

Subgoal Labeled Worked Example

Create Component

1. From the basic palette drag out a *Button*.

Set Properties

2. Set the image source to "gypsy.jpg".
3. Clear the default text.
4. Set the width to fill the parent's width and the height to 300 pixels.

Create Component

5. From the basic palette drag out a label.
6. Place the label underneath the image.

Set Properties

7. Set the text to *Click button to see your fortune*.
8. Rename it to *fortuneLabel.*

Unlabeled Worked Example

1. From the basic palette drag out a *Button*.
2. Set the image source to "gypsy.jpg".
3. Clear the default text.
4. Set the width to fill the parent's width and the height to 300 pixels.
5. From the basic palette drag out a label.
6. Place the label underneath the image.
7. Set the text to *Click button to see your fortune*.
8. Rename it to *fortuneLabel.*

Figure 1. Worked examples with and without subgoal labels. The subgoal label "create component" corresponds to the function of creating a component for the app, such as a button. The subgoal label "set properties" corresponds to the function of setting the properties of components, such as determining the size of the button.
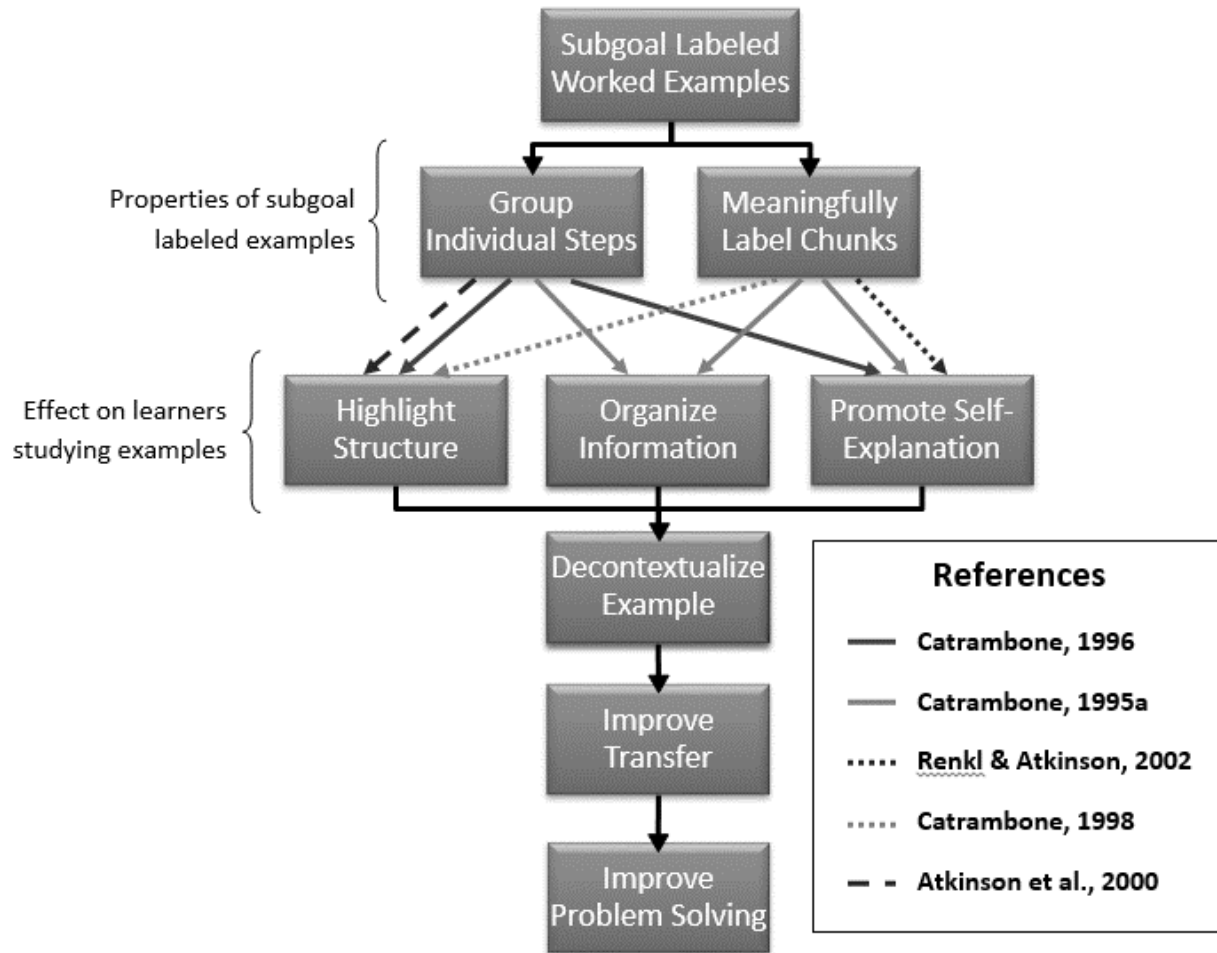
Figure 2. Diagram of how subgoal labeled worked examples can help learners improve problem solving performance. The "properties of subgoal labeled examples" level describes the physical characteristics of subgoal labels. The "effect on learners studying examples" level describes how these characteristics help the learners use effective learning strategies.

Subgoal Labeled Expository Text

Create Component

Components are the pieces that provide your app functionality, such as a *button* that users can press or a *label* to display…

Set Properties

You'll be able to change the properties of each component in the App Inventor Designer as well. For example, you can change…

Unlabeled Expository Text

Components are the pieces that provide your app functionality, such as a *button* that users can press or a *label* to display…

You'll be able to change the properties of each component in the App Inventor Designer as well. For example, you can change…

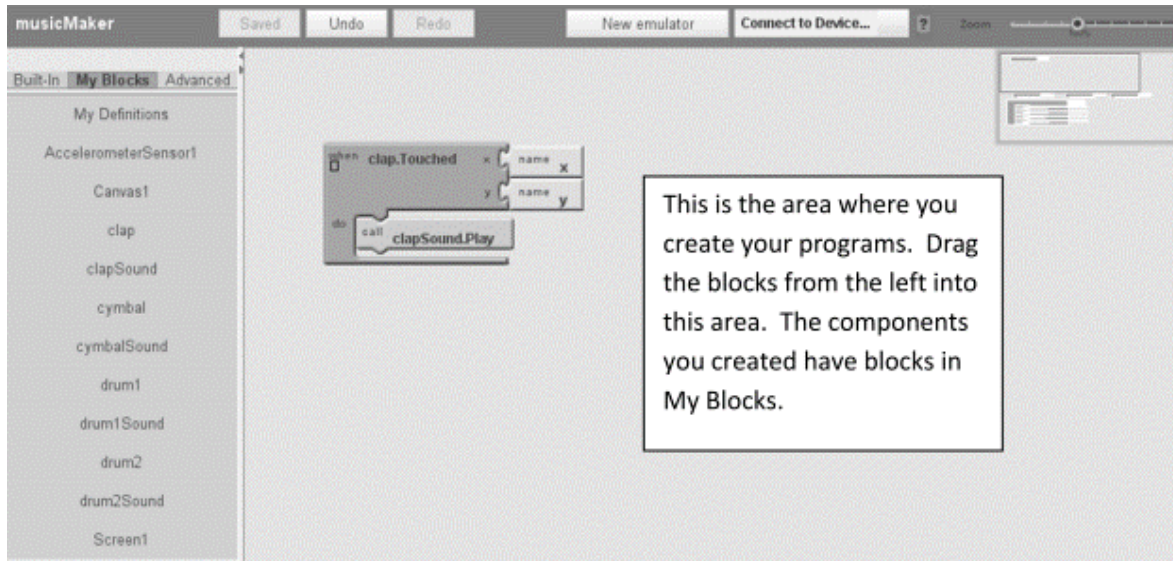*Figure 3.* Expository text with and without subgoal labels.

Figure 4. App Inventor interface with interlocking blocks of code used to program features. The image shows blocks that would play the "clapSound" when the "clap" component is touched.
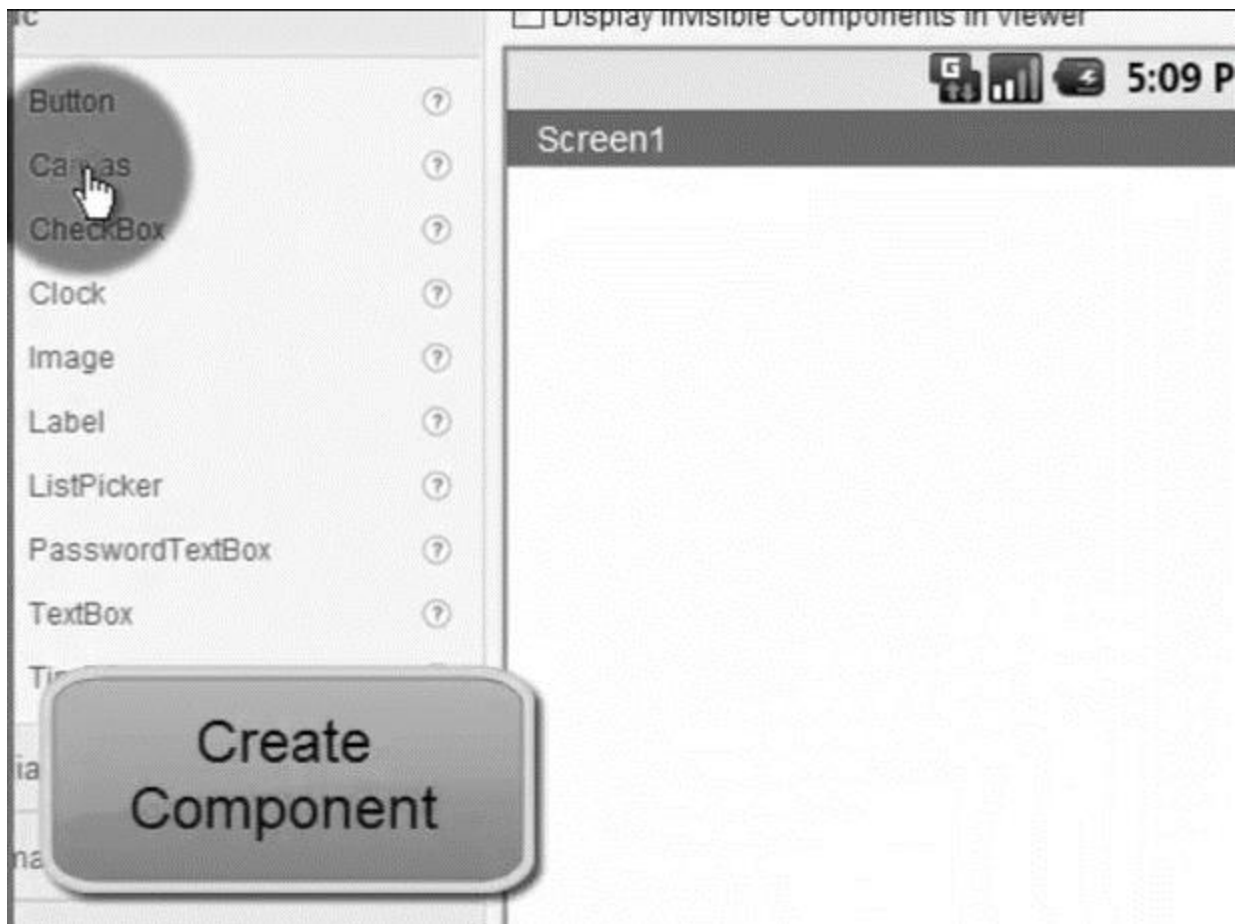
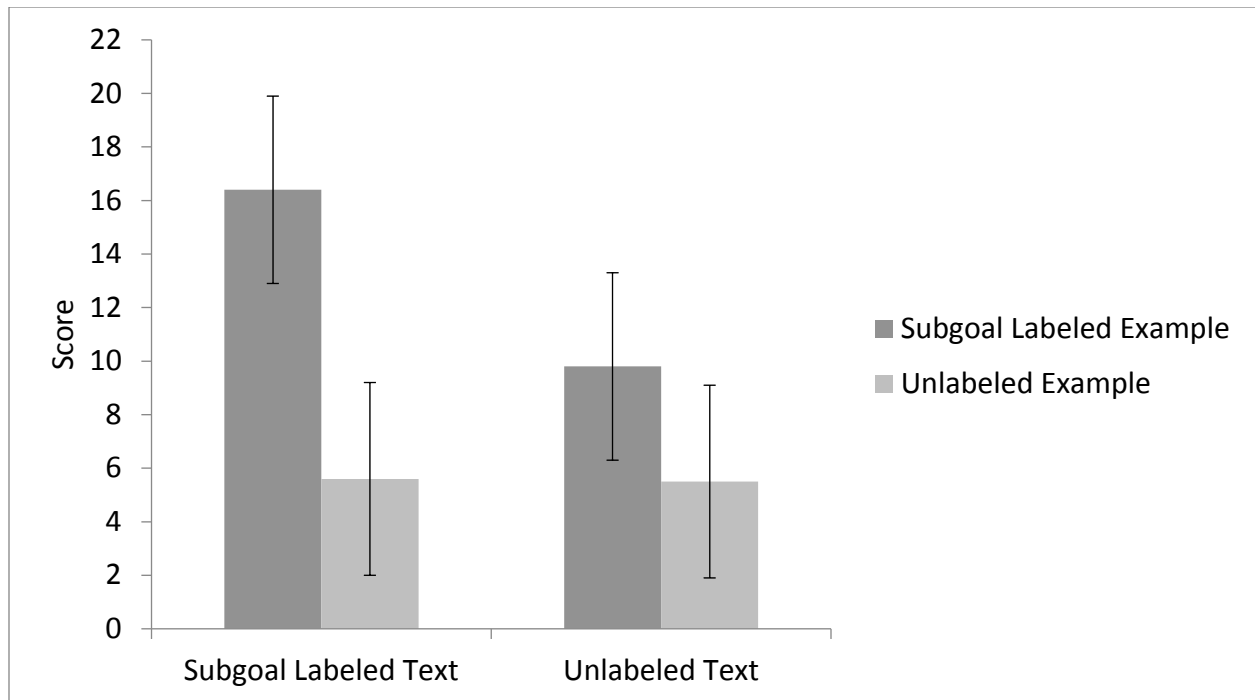*Figure 5.* Screenshot of the "Create Component" subgoal callout in video demonstration.

Figure 6. Scores on problem solving task by condition in Experiment 1. Error bars represent a standard deviation.
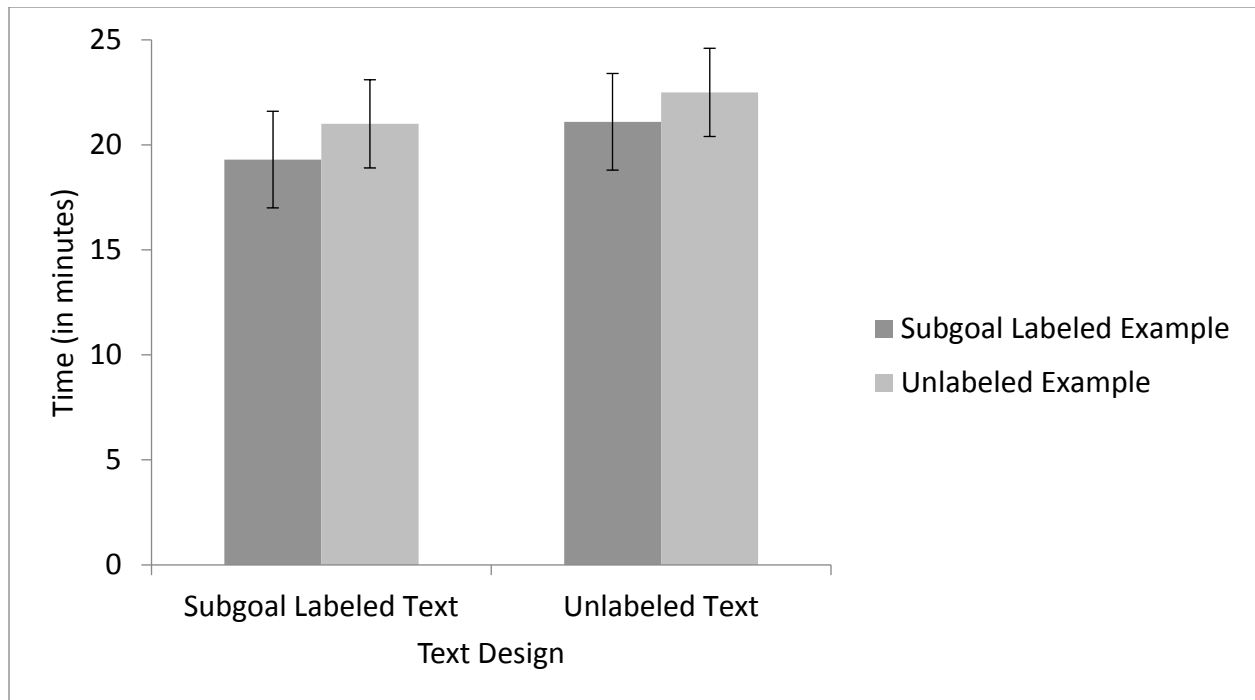
Figure 7. Time spent on problem solving task by condition in Experiment 1. Error bars represent a standard deviation.
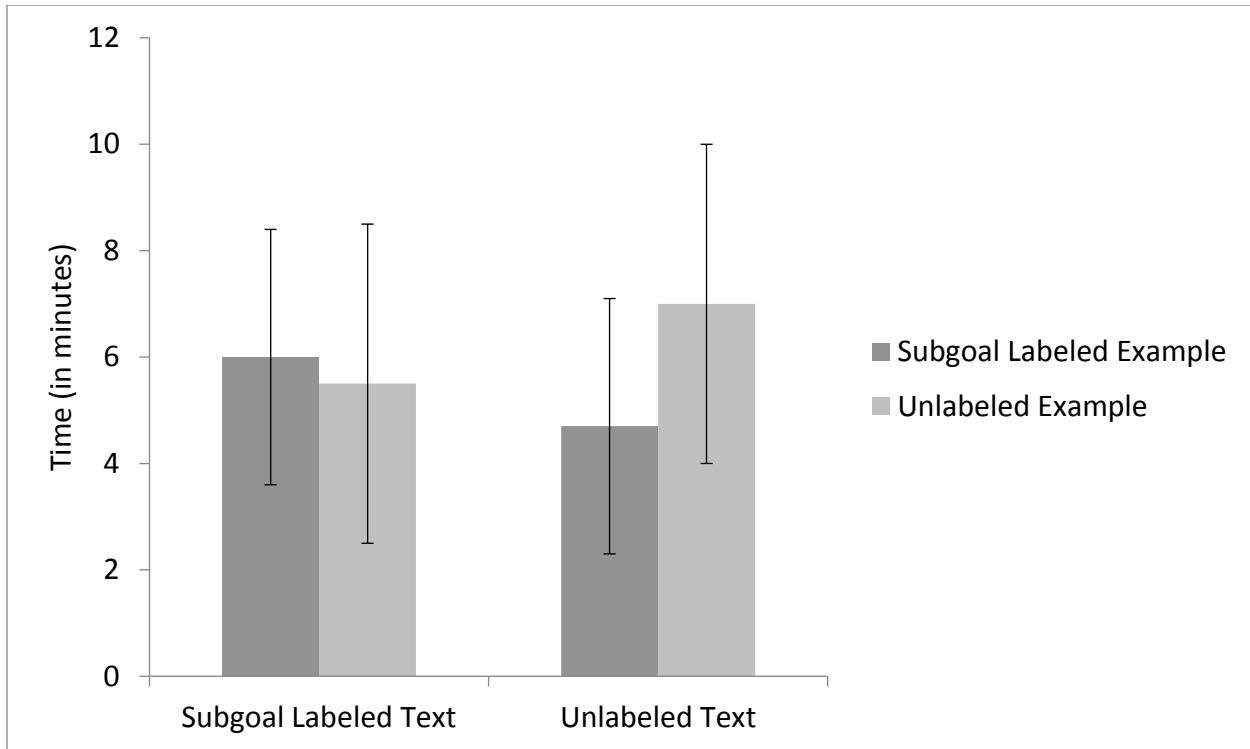
Figure 8. Time spent on explanation task by condition in Experiment 1. Error bars represent a standard deviation.
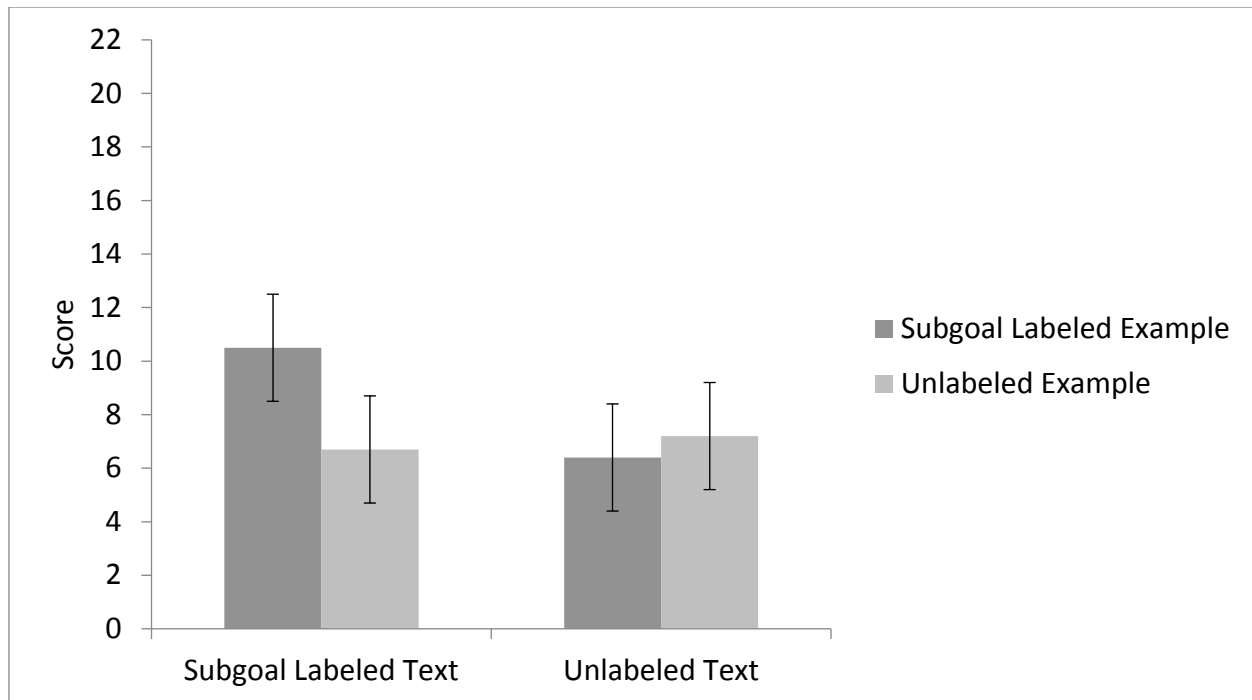
Figure 9. Scores on problem solving task by condition in Experiment 2. Error bars represent a standard deviation.
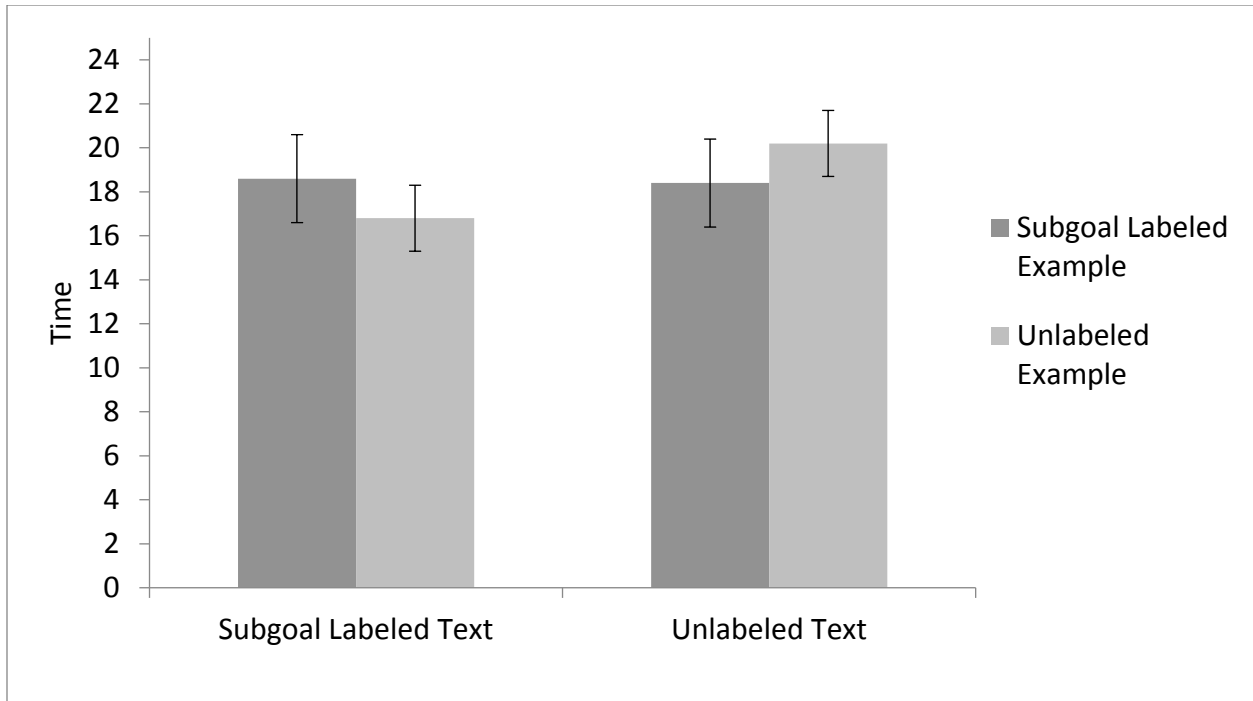
Figure 10. Time on task for problem solving task by condition in Experiment 2. Error bars represent a standard deviation.
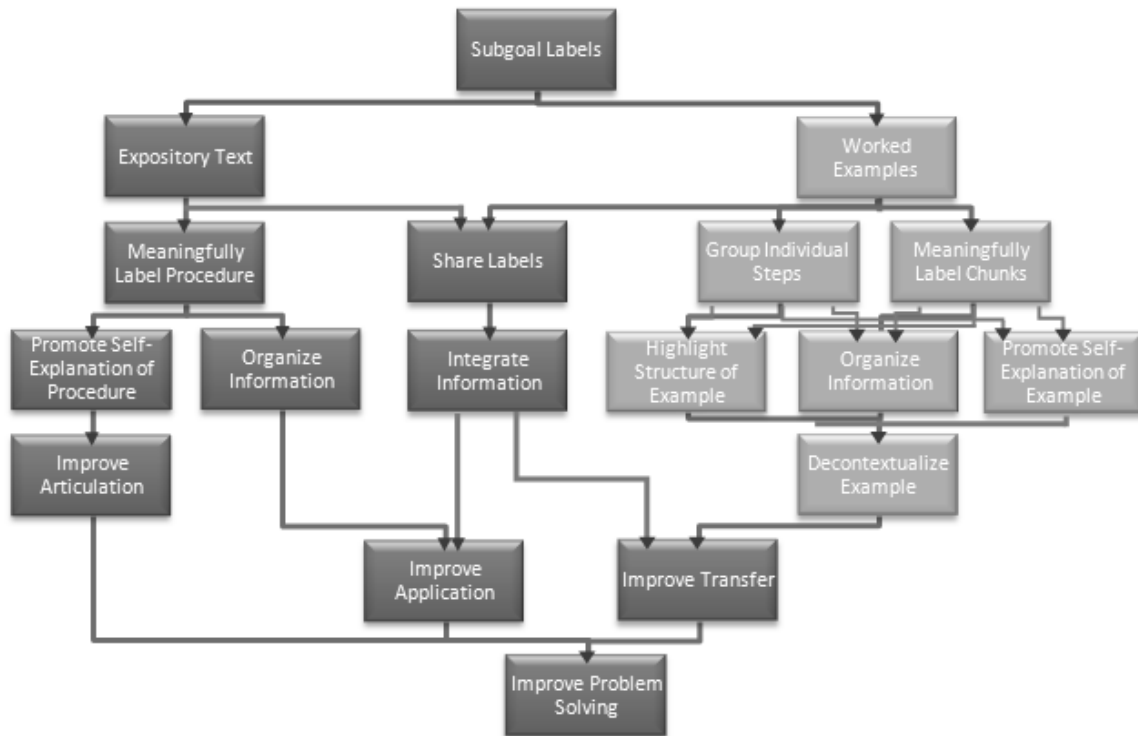
Figure 11. Model of how subgoal labeled expository text and worked examples can help learners improve problem solving performance. Starting from the top, the first two levels describe the type of instructional material. The third level describes the physical characteristics of subgoal labels. The fourth level describes how the physical characteristics help the learners use effective learning strategies. The fifth and sixth levels describe how these strategies help learners understand the instructional material, and the last level describes the outcome.

Table 1

Distribution of selected demographics among conditions in Experiment 1.

Condition	SAT Writing				Comfort with Computers			
	M	*SD*	*r*	*p*	*M*	*SD*	*r*	*p*
Subgoal-text, Subgoal-example	631	87	-.07	.73	5.88	1.16	-.11	.68
Unlabeled-text, Subgoal-example	659	54	.03	.84	6.41	.93	.08	.78
Subgoal-text, Unlabeled-example	625	72	-.31	.08	5.57	1.19	.02	.95
Unlabeled-text, Unlabeled-example	665	41	-.20	.26	6.08	1.05	.41	.15

Note: Comfort with computers on 7-pt. scale (1-Not Comfortable At All and 7-Very

Comfortable).

Table 2

Simple main effects of problem solving score in Experiment 1.

	Simple main effect	F	p	est. ω^2
Text Format	Unlabeled example	.01	.92	$< .01$
	Subgoal labeled example	26.7	$< .001$	.19
Example Format	Unlabeled text	11.5	.001	.09
	Subgoal labeled text	71.5	$< .001$	.38

Table 3

Simple main effects comparing groups for number of groups containing structurally-related steps in explanation task in Experiment 1.

Simple main effect		F	p	est. ω^2
Text Format	Unlabeled example	.01	.91	< .01
	Subgoal labeled example	6.58	.01	.05
Example Format	Unlabeled text	.03	.86	< .01
	Subgoal labeled text	6.88	.01	.06

Table 4

T-tests comparing groups for time on task for the explanation task in Experiment 1.

Condition	*n*	*M*	*SD*	*t*	Std. error	*p*
Unlabeled-text, Unlabeled-example	30	7.0	3.1			
				1.171	.820	.247
Subgoal-text, Subgoal-example	30	6.0	2.6			
				.707	.724	.482
Subgoal-text, Unlabeled-example	30	5.5	2.7			
				1.194	.643	.237
Unlabeled-text, Subgoal-example	30	4.7	2.1			

Table 5

Distribution of selected demographics among conditions in Experiment 2.

	High School GPA (as percentage)				Credit Hours Earned			
Condition	*M*	*SD*	*r*	*p*	*M*	*SD*	*r*	*p*
Subgoal-text, Subgoal-example	87%	.33	.05	.81	63.0	37.8	.27	.16
Unlabeled-text, Subgoal-example	84%	.32	.08	.69	52.2	41.1	.07	.75
Subgoal-text, Unlabeled-example	71%	.42	.25	.12	59.3	41.4	.18	.39
Unlabeled-text, Unlabeled-example	77%	.39	.30	.13	48.2	39.4	.18	.39

Table 6

Simple main effects of instructional time in Experiment 2.

	Simple main effect	F	p	est. ω^2
Text Format	Unlabeled example	.01	.97	<.01
	Subgoal labeled example	9.08	.003	.08
Example Format	Unlabeled text	1.74	.19	.02
	Subgoal labeled text	2.94	.09	.03

Table 7

Simple main effects of problem solving score in Experiment 2.

	Simple main effect	F	p	est. ω^2
Text Format	Unlabeled example	.12	.74	.01
	Subgoal labeled example	9.33	.003	.08
Example Format	Unlabeled text	.46	.50	.01
	Subgoal labeled text	7.40	.008	.07