
Learnersourcing Subgoal Labels for How-to Videos

Sarah Weir
MIT CSAIL
sweir@mit.edu

Abstract

Websites like YouTube provide an easy way to watch the billions of how-to videos on the web, but the interfaces are not optimized for learning. Previous research suggests that users learn more from how-to videos when the information from the video is presented in outline form, with individual steps and higher-level groupings of steps (subgoals) shown. We intend to create an alternative video viewer where the steps and subgoals are displayed alongside the video. In order to generate this information we propose a *learnersourcing* approach where we gather useful information from people trying to actively learn from a video. We believe learnersourcing is a sustainable and constructive method for enhancing educational material. To demonstrate this method, we created a workflow that encourages users to contribute and refine subgoals for a given how-to video. Users in our pilot study of three videos were able to generate subgoals comparable to those created by trained subgoal extractors, which suggests that learnersourcing may be a viable approach.

Author Keywords

Video tutorials; how-to videos; subgoals; video annotation

ACM Classification Keywords

H.5.2 User Interfaces: Graphical User Interfaces

Paste the appropriate copyright/license statement here. ACM now supports three different publication options:

- ACM copyright: ACM holds the copyright on the work. This is the historical approach.
- License: The author(s) retain copyright, but ACM receives an exclusive publication license.
- Open Access: The author(s) wish to pay for the work to be open access. The additional fee must be paid to ACM.

This text field is large enough to hold the appropriate release statement assuming it is single-spaced in Verdana 7 point font. Please do not change the size of this text box.

How to put Christmas lights on a house

Which of the following best describes the video section you just watched?

Choose the best answer or add your own.

- Materials you'll need to decorate your house with lights
- Collect necessary tools
- Preparation
-
- None apply

Submit **Cancel**

Question and instructions appear here for each stage.

Wiki Video Outline **Add subhead**

Materials you'll need to decorate your house with lights **Edit** **Delete**

Collect necessary tools **Edit** **Delete**

Preparation **Edit** **Delete**

1. Gather materials: tools, extension cord, lights, ladder
2. Use plenty of extension cords

Preparation Tips **Edit** **Delete**

Safety advice **Edit** **Delete**

3. Use a weather protected outlet strip
4. Use a lighting timer if you'll be away from home

Follow safety tips for the different tools **Edit** **Delete**

5. Make sure ladder is locked in place, secure, and stable
6. Use cord wraps and color-coded containers to store materials

Decide on method to fasten lights to your fascia **Edit** **Delete**

Installation **Edit** **Delete**

7. Optionally, use screws or staples to secure lights

Put lights up on house **Edit** **Delete**

8. Clip gutter hooks onto gutter
9. Snap in the wiring to the gutter hooks

Outline appears here for each stage. When users answer the questions, the outline is updated accordingly. For stage 2 (displayed here), subgoals generated by previous users are shown in tandem and answering the question refines the outline.

Figure 1. The alternative video interface that we used to test the learnersourcing method. The question asked users and the initial outline shown differ based on the stage in the learnersourcing workflow.

Introduction

Previous work [10] suggests that presenting subgoal labels along with lower-level steps for procedural tasks can increase learning. Nguyen's work [8] on crowdsourced annotations suggests that crowdsourcing

on Mechanical Turk can generate viable annotations for the steps in a how-to video. However, paid crowdsourcing on a large scale is expensive and relies on Turkers that do not receive a pedagogical benefit from completing the tasks.

Learnersourcing [7] looks to people actively trying to learn from a how-to video, who may contribute higher-quality information than Turkers. Because learners are already watching the video, the method is scalable. A learnersourcing workflow can also provide a pedagogical benefit to learners who complete the tasks. The goal of learnersourcing is to generate a high-quality list of subgoals that can then be displayed to other learners. This paper introduces a learnersourcing workflow for how-to videos from YouTube.

This paper makes the following contributions:

- An exploration into the idea of learnersourcing, a form of crowdsourcing that asks learners for information in a way that is pedagogically useful
- A learnersourcing workflow design and interface (Figure 1) that can be used to generate subgoals for how-to videos from YouTube
- Results from a pilot study suggesting that the workflow design is viable and can be applied to multiple video domains.

Related work

Previous work suggests that presenting learners with subgoals for procedural tasks affords greater learning. Eiriksdottir et al. [4] explored the effects of presenting learners with different forms of instructional material. They discovered that including specific instructions helped learners complete the initial task but those learners did not retain the information. Conversely, learners presented with more holistic instructions had greater learning and transfer. Margulieux, et al. studied the effects of presenting learners with instructions that

(a)

1. Gather materials: tools, extension cord, lights, ladder
2. Use plenty of extension cords
3. Use a weather protected outlet strip
4. Use a lighting timer if you'll be away from home
5. Make sure ladder is locked in place, secure, and stable
6. Use cord wraps and color-coded containers to store materials
7. Optionally, use screws or staples to secure lights
8. Clip gutter hooks onto gutter
9. Snap in the wiring to the gutter hooks

(b)

Materials you'll need to decorate your house with lights Edit Delete

1. Gather materials: tools, extension cord, lights, ladder

Useful accessories Edit Delete

2. Use plenty of extension cords
3. Use a weather protected outlet strip

Notes before starting Edit Delete

4. Use a lighting timer if you'll be away from home
5. Make sure ladder is locked in place, secure, and stable
6. Use cord wraps and color-coded containers to store materials

Decide on method to fasten lights to your fascia Edit Delete

7. Optionally, use screws or staples to secure lights

Gutter hooks as alternative Edit Delete

8. Clip gutter hooks onto gutter
9. Snap in the wiring to the gutter hooks

(c)

Prepare necessary materials Edit Delete

1. Gather materials: tools, extension cord, lights, ladder
2. Use plenty of extension cords

Safety advice Edit Delete

3. Use a weather protected outlet strip
4. Use a lighting timer if you'll be away from home
5. Make sure ladder is locked in place, secure, and stable
6. Use cord wraps and color-coded containers to store materials

Decide on method to attach lights to house Edit Delete

7. Optionally, use screws or staples to secure lights

Put on lights using gutter hooks Edit Delete

8. Clip gutter hooks onto gutter
9. Snap in the wiring to the gutter hooks

Figure 2. A comparison between the blank video outline (a), the video outline with learner-generated subgoals (b), and the outline with subgoals generated by trained extractors (c).

included both specific steps and subgoals [10] and found that learning and transfer for tasks were increased over learners that received the steps alone. Buykx and Petrie [1] included subgoals in recipe instructions to show that including the steps and subgoals information can be effective in domains other than software applications.

Significant work has already been done on tutorial interfaces. Many existing interfaces improve a user's ability to discover specific tutorials [9, 12], while others extract information from user actions or existing material to improve the effectiveness of the tutorials themselves [3, 5, 6, 13]. These systems suggest that exploiting metadata from the videos, such as the steps involved in the task and the order in which they are done, can improve skill learning for procedural tasks. However, existing systems use the underlying structure of the application to generate content [5, 9, 12, 13] and cannot easily be generalized to domains other than software. In his TAPS study [2], Catrambone proposes a method for extracting subgoals with a domain expert and a knowledge extraction expert working together. This work suggests that knowledge extraction experts are a viable source for providing information to help other learners. Unlike domain experts, learners tend to provide information at a level that can be absorbed by other learners.

Proposed method

We propose to investigate the learnersourcing method for extracting useful information from how-to videos. In this study, we focus on generating subgoal labels, assuming that the lower-level steps have already been extracted. Systems exist to generate steps [e.g. 5, 8, 12], but nothing focuses on subgoal creation. We believe learnersourcing is a viable approach because it is not only scalable and cheap, but may be pedagogically useful to learners as the tasks are designed to cause learners to think critically about the video they have just watched. Such summarization

causes learners to recall what they have just seen, which could help them retain the information.

Our proposed interface is a three-stage workflow with two major goals: to generate high-quality subgoals and to be pedagogically useful for the learners at every step. We define high-quality subgoals as having the correct scope, meaning the goal refers to the correct subset of steps, and correct language, meaning active voice, not too long, and with the right specificity. In order to reduce the task's effort to keep learners' focus on the video, we decided to tackle different aspects of subgoals in each stage.

Workflow design

Stage 1: Subgoal Generation

In the initial stage, learners are presented with a video and the list of steps (Figure 2(a)) that coincide with the task being described in the video. The steps, which we generated ahead of time, comprise the beginning of a video outline that learners will contribute to as they watch the video. This workflow is agnostic to the method used to generate the steps. Every thirty seconds, the video stops and learners are asked "What was the overall goal of the video section you just watched?" (Figure 3(a)). The answer to this question becomes the subgoal for the steps that were covered. Learners are given visual feedback as a draggable and editable subgoal populates the video outline at the correct spot. After learners answer the question, they are given instructions to edit or move the subgoal if it is not correct.

We stop the video at specified intervals and ask learners a directed question because test users in our early prototype stages contributed more subgoals when asked a direct question versus when they were asked to make contributions but not forced to. In order to not detract from the video, we made the questions quick to answer.

(a) Stage 1 question asked to learners

What was the overall goal of the video section you just watched?

e.g. Mix the dry ingredients

(b) Stage 2 question asked to learners

Which of the following best describes the video section you just watched?

Choose the best answer or add your own.

Materials you'll need to decorate your house with lights

Collect necessary tools

Preparation

 None apply

(c) Stage 3 question asked to learners

Does this subheading accurately summarize the below steps?

Feel free to add your own or decide that none apply. Subheadings for the wiki outline should be consistent!

The subheading should be actionable, i.e. "Mix the dry ingredients" instead of "Dry ingredients".

Subhead: Materials you'll need to decorate your house with lights

1. Gather materials: tools, extension cord, lights, ladder

This subhead applies

Replace subhead:

There should be no subheading here

We chose the thirty-second interval because after analyzing multiple videos of around five minutes in length, we found that many of the subgoals we generated fell around that mark. Future work will explore methods to determine a better interval.

Stage 2: Subgoal Evaluation

In this stage, we present learners with the video and the video outline that now includes the steps along with the subgoals generated from multiple learners in stage 1 (Figure 1). The outline is intentionally cluttered to indicate to users that it is unfinished. The video stops where each subgoal ends and learners are asked to choose between previously generated subgoals (Figure 3(b)). When they make their choice, users get visual feedback as only the option they chose remains in the outline and the rest disappear.

Stage 3: Subgoal Proofreading and Evaluation

In this stage, we present the users with the video and the outline including the final set of subgoals generated through iterations of stage 2. The video stops after each subgoal and learners are presented with the subgoal and the corresponding steps that have been assigned to it (Figure 3(c)). They are then asked if the subgoal is a valid summarization, and if it follows certain language guidelines (such as being active voice and consistent with other subgoals in the outline) Users can either send back the same subgoal or suggest a new one. Again, learners are given visual feedback in the outline when they make an update.

Experiment

To test the workflow, we simulated the three-stage workflow for three videos: "How to use the Symbol Tool in Illustrator", "How to put Christmas lights on a house", and "How to Sort and Reverse Sort Arrays in Perl" all from YouTube. To study whether this approach is generalizable to multiple domains, we chose videos

from three different domains: software tutorials, do-it-yourself type videos, and programming.

We recruited 9 college undergraduates, graduate students, and working professionals to complete the workflow for each video. The learners we chose were novices in each of the domains, and they were asked to imagine they were interested in learning the given topic. Participants were not asked specifically to answer the questions throughout the video, and were not told about subgoals prior to participating. The users' results were manually funneled through the stages, as the output from each stage informs the next stage's interface.

Results

After engaging each step of the workflow, users were able to generate high-quality subgoals for each of the three videos. In the following analysis, we will define a 'correct' subgoal as one that has similar language and is in the same location as subgoals generated by trained subgoal extractors. Knowledge extraction experts who had been trained to extract subgoals from the domain expert in the video generated these subgoals. We had three users complete each stage for each of the videos, and for these results we consider the average of all three submissions as our number of correct subgoals. We found in general that at the end of each stage there was more progress towards the correct answer (Table 1).

The results are not statistically significant due to our small number of users, but even without multiple iterations we find that most of the subgoals are comparable to the expert-generated subgoals. Additionally, the incorrect subgoals still made sense in the context of the video and we would feel comfortable displaying them to other users.

For the video on Christmas Lights, the number of correct subgoals decreased after the second stage because users converged on a different subgoal than

Figure 3. The questions asked during each stage of the workflow. The question in the first stage (a) lets learners generate subgoals. In the second stage, learners choose between previously generated subgoals (b), and in the third stage they proofread learner-generated subgoals (c).

Video	Number of steps in video	Average number of correct subgoals after stage 1	Average number of correct subgoals after stage 2	Average number of correct subgoals after stage 3	Average number of total expert subgoals
Christmas Lights	9	2.7	3.25	2.7	4
Symbol Tool in Illustrator	27	5	9.3	10	11
Sorting Arrays using Perl	13	3	3.7	4	4

Table 1. Results from our pilot study show that in general, subgoal accuracy increases as users complete each of the three stages.

the expert chose, but both could be considered valid given the context of the video. Specifically, the video talks about accessories to gather and mentions safety tips. The expert subgoal for this section was "Follow safety advice," and the user-generated subgoal was similar to "Gather useful accessories."

Discussion

Each of the stages encouraged different input from users. In the first stage, we found that most users left the subgoals in the same positions that they were generated. This generally led to there being more subgoals than necessary after the first stage. In the second stage, users were asked to choose between all of the user generated subgoals, and with only three users completing that stage it was hard to judge what the most popular subgoal was. The most major changes in subgoal position were made at this stage. In the third stage, users were encouraged to make sure the language was consistent and actionable, so most of the changes were text edits to previous subgoals. As users made different kinds of changes to the subgoals at each

stage, we believe that it is valuable to have multiple stages to generate subgoals, each with a different task.

The users in these studies all contributed new information and made the subgoals closer to the expertly generated subgoals. However, we recognize that in our pilot study, learners were not necessarily motivated by a desire to learn the material, and were more likely completing the study because we asked them to. Therefore, we will deploy the system live to real learners to see if our current design provides enough motivation to contribute high-quality subgoals.

We found that the difficulty of the video in relation to the users' experience contributed to the quality of the subgoals that they generated. Users with more experience in programming, for example, tended to use more programming jargon for the Perl video than users who had little or no programming experience. Additionally, for more cognitively intensive or unfamiliar videos, such as the Perl video, users spent a lot of time analyzing the provided steps before submitting their subgoal in the first stage. This suggests that it may be useful to show users which steps were covered by a given video section.

Design Dimensions for Learnersourcing Tasks

Although we present a system for generating subgoals, this workflow can be generalized to retrieving many different kinds of information from learners watching videos. This system was designed without a specific video in mind, and after the pilot study we think it will be domain agnostic. From our work in designing and testing the subgoal workflow, we have identified the following design dimensions for learnersourcing tasks:

- Information available to first time learners. E.g., The list of steps and the video.
- Amount of instruction/context given. E.g., A basic indication of what the task would be, but little background on subgoals.

- Learner’s task. E.g., To generate, evaluate, or proofread subgoals.
- Type of prompting. E.g., A text input or multiple choice question.

We chose these factors based on our subgoal labeling experience, but we believe these dimensions should be considered by others designing learnersourcing tasks.

Future work

We intend to create a more robust and polished interface that can be deployed to the world for a set of specified videos in order to better assess its effectiveness. Specifically, we want to study how well learners answer the questions we pose. We also intend to make the interface more interactive by having each of the steps and subgoals link to a specific point in the video and showing which steps have been covered when learners are asked to contribute a subgoal for that video section. Nevertheless, through our pilot study of the three-stage learnersourcing approach, we were able to extract quality subgoals for three videos of different domains, which suggests that the approach can be generalized to a variety of topics. With our continued research, we will be able to better evaluate learnersourcing and recommend how it can be applied to a variety of applications.

Acknowledgments

Thank you to Juho Kim, Rob Miller, and the UID Group at MIT CSAIL for their constant help and feedback. The MIT EECS SuperUROP Program and Cisco Systems, Inc. have supported this work.

References

- [1] Buykx, L., Petrie, H. Recipe Sub-goals and Graphs: An Evaluation by Cooks. *CEA 2012*.
- [2] Catrambone, R. Task Analysis by Problem Solving (TAPS): Uncovering Expert Knowledge to Develop High-Quality Instructional Materials and Training.
- [3] Chi, P.-Y., Ahn, S., Ren, A., Dontcheva, M., Li, W.,

- and Hartmann, B. Mixt: Automatic generation of step-by-step mixed media tutorial. *UIST 2012*.
- [4] Eiriksdottir, E., and Catrambone, R. Procedural instructions, principles, and examples: how to structure instructions for procedural tasks to enhance performance, learning, and transfer. *Hum Factors 53, 6 (2011), 749–70*.
 - [5] Grossman, T., Matejka, J., and Fitzmaurice, G. Chronicle: capture, exploration, and playback of document workflow histories. *UIST 2010, 143–152*.
 - [6] Kim, J. ToolScape: Enhancing Experience of How-to Videos. *Ext. Abstracts CHI 2013*
 - [7] Kim, J., Miller, R.C., Gajos, K.Z. Learnersourcing Subgoal Labeling to Support Learning from How-to Videos. *Ext. Abstracts CHI 2013*.
 - [8] Kim, J., Nguyen, P., Weir, S., Guo, P., Miller, R.C., Gajos, K.Z. Crowdsourcing Step-by-Step Information Extraction to Enhance Existing How-to Videos. *To appear in CHI 2014*.
 - [9] Kong, N., Grossman, T., Hartmann, B., Fitzmaurice, G., Agrawala, M. Delta: A Tool for Representing and Comparing Workflows. *CHI 2012*.
 - [10] Margulieux, L., Guzdial, M., Catrambone, R. Subgoal-Labeled Instructional Material Improves Performance and Transfer in Learning to Develop Mobile Applications. *ICER 2012*.
 - [11] Nguyen, P., Kim, J., Miller, R.C. Generating Annotations for How-to Videos Using Crowdsourcing. *Ext. Abstracts CHI 2013*.
 - [12] Pavel, A., Berthouzoz, F., Hartmann, B., Agrawala, M. Browsing and Analyzing the Command-Level Structure of Large Collections of Image Manipulation Tutorials. *Technical Report No. UCB/EECS-2013-167*.
 - [13] Pongnumkul, S., Dontcheva, M., Li, W., Wang, J., Bourdev, L., Avidan, S., and Cohen, M. Pause-and-play: Automatically linking screencast video tutorials with applications. *UIST 2011*.