AUTOMATICALLY UNDERSTANDING HANDWRITTEN SELF-EXPLANATIONS

James Herold
Computer Science Department

Thomas Stahovich
Mechanical Engineering Department

University of California, Riverside

Abstract

Research has demonstrated that self-explanation hones students' metacognitive skills and increases their performance. We have found however, that not all self-explanation is substantive. Our goal is to develop computational techniques capable of determining whether a student's explanation is relevant or not. This will then enable us, for example, to create an interactive tutoring system capable of prompting students to continue their explanations when necessary. This is a tractable task as self-explanations typically contain a small number of possible concepts. The language used to express these concepts can vary greatly, but our task is only to identify the existence of the concepts, not to perform general machine interpretation. In this paper, we present early work on the automatic understanding of students' handwritten self-explanation of their solutions to homework problems in an engineering statics course. We employ an open information extraction technique popularly used to identify relations present in broadcast news transcripts. In our study, this technique achieved up to 97% accuracy at identifying when the content of a student's self-explanation did not match the concepts used by experts in explaining their own work on the same problem.

Introduction

Research has demonstrated that self-explanation hones students' metacognitive skills and increases student performance. We have found, however, that not all self-explanation is substantive. Our goal is to develop computational techniques capable of determining if a student's explanation is relevant or not. This will then enable us, for example, to create an interactive tutoring system capable of prompting students to continue their explanations when necessary. This is a tractable task as self-explanations typically contain only a small number of possible concepts. The language used to express these concepts can vary greatly, but our task is only to identify the existence of the concepts, not to perform general machine interpretation. In this paper, we present early work on the automatic understanding of students' handwritten self-explanation of their solutions to homework problems.

In the winter of 2011, we conducted a study in which 39 students in an undergraduate statics course were asked to generate handwritten self-explanations of their homework solutions. Students were provided a set of questions with each homework assignment, eliciting an explanation of the reasoning behind key steps of their solution process. For example, students were asked why they chose the free body diagram they used, or why they chose a particular point to compute moments about. To provide a benchmark for the self-explanations, we asked three experts to solve some of the same problems and generate their own self-explanations. We manually analyzed these and identified the concepts used. We found that the experts used only a small set of concepts in their explanation of any particular problem-solving step. We would expect that a student with an expert-stance would utilize the same set of concepts in their explanations.

In the current work, we employ an information extraction technique to automatically identify whether a student's self-explanation responses employ the same concepts used by the experts. For example, this technique can determine if a student assumed that bodies in a friction problem were on the verge of slip, a concept that experts often included in their self-explanations.

In our experiments, this technique has proven to be quite reliable, achieving an accuracy of up to 97%. This level of accuracy can be attributed to the consistent nature of the students' self-explanation; there was typically a small set of concepts expressed in the responses to any given self-explanation prompt. Furthermore this high-level of accuracy suggests that it may be feasible to develop automated systems to elicit meaningful self-explanations from students.

Related Work

Chi et al. [1] argue that "the metacognitive component of training is important in that it allows students to understand and take control of their learning process." Metacognition is the ability to be aware of one's own learning process and it serves as a major foundation for research performed on self-explanation.

Mayer [2] examines differences between retention and transfer. The former is the application of knowledge from one problem to an identical problem, while the latter is the application of that knowledge to a different problem. Mayer argues that metaskill, the ability to control and monitor one's cognitive processes, is an essential part of transfer. Metaskill strategies may be taught just as any other skill, such as arithmetic, via strategy instruction. For example, students who are taught basic reading skills as well as strategies for summarizing their own reading, perform better on transfer questions [3]. These results demonstrate the inadequacy of teaching only basic skills and the need to complement them with metacognitive skills. In this context, we use self-explanation as a means to develop metaskills.

Chi et al. [1] made comparisons between two groups of students: "poor" and "good" performing students. These students were asked to generate self-explanation after studying worked-out example problems. The results of this study demonstrated that students who perform poorly are typically unable to generate sufficient self-explanation of the worked-out example problems.

Steif et al. [4] present and evaluate a strategy for teaching statics concepts which focuses on students' conceptual knowledge. During instruction, students are given examples of free body diagrams and asked whether they are correct. Students are then shown a video explaining what errors are present in the diagram. Additionally, students in an experimental group are asked questions eliciting an explanation of the relationships between the diagram and the forces which act upon it. This work showed a significantly lower error rate among students who generated self-explanation. Additionally though, this work provides motivation for the techniques presented in this paper. Our method enables automatic analysis of the content of self-explanation, which may enable the creation of intelligent tutoring systems that probe a student's understanding if the student's self-explanation is lacking.

Information extraction (IE) is the process by which target relations are extracted from machine readable documents, such as text transcripts. This is distinguishable from attempting to understand the entire content of such documents. There is a long history of research in IE techniques. [5] Older techniques have typically relied a great deal on domain dependent attributes and were usually rule-based [6] or applied machine learning techniques. [7] While these systems achieved high accuracy, their domain-dependent nature required a great deal of manual effort to adapt them to new domains. More recently, researchers have focused on automatic IE techniques intended for use with the World Wide Web. These techniques are more general and extensible than prior methods and are thus called open IE techniques. The technique we use in this paper is an open IE technique which allows us to easily train a system to extract relations in our domain.

Self-explanation Transcripts and Labeling

In the winter of 2011, we conducted a study in which 39 students enrolled in an undergraduate statics course were asked to provide handwritten self-explanations of their work. In this course, five of the nine homework assignments were accompanied by a series of prompts which elicited from students an explanation of the reasoning behind each of their problem-solving steps.

To ground our analysis, we asked three experts to complete homework assignments three and eight as well as generate the same kinds of self-explanation as the students. These experts comprised one graduate and two undergraduate mechanical engineering students; the latter two had solved these exact homework problems two years prior.

We manually transcribed the student and expert handwritten self-explanations. Spelling errors were corrected, but grammatical errors were left as is. The expert's self-explanation transcripts exemplify the types of responses we expect from students who possess an expert-stance on statics concepts. We performed an in-depth analysis of the experts' responses to three different explanation prompts. Namely, we analyzed the responses to prompts one and four from homework three, and prompt three from homework eight.

Prompt one of homework three asks, "Why did you select the system that you used for your free-body diagram?" We have found that experts generate one of the following four different responses: *required-forces*, *least-forces*, *only-one*, and *alternative-difficult*. A *required-forces* response is one in which the expert explains that the free body diagram contained all forces necessary to solve for the unknowns without revealing extraneous forces. Similarly, a *least-forces* response is one in which the expert explains that the free body diagram contains the minimum number of forces that is needed to solve the problem. While this response is similar to the prior *required-forces* response, the language used in each is different enough to necessitate two different response types. An *only-one* response is one in which the expert explains that the free body diagram is the only one that can be used to solve the problem. Lastly, an *alternative-difficult* response is one in which the expert indicates that an alternate free body diagram can be used but will lead to a solution that is more difficult.

Prompt four of homework three asks, "When computing moments for the moment equilibrium equation, why did you choose the particular point that you used to take moments about?" We have found that experts generate one of three different responses: *directly-solves*, *only-unknown*, or *eliminate-forces*. A *directly-solves* response is one in which the expert indicates that taking the moment about the point he chose directly solves for the unknown indicated in the problem description. An *only-unknown* response is one in which the expert explains that taking the moment about the point he chose results in a moment equilibrium equation with only one unknown which can be directly solved. An *eliminate-forces* response is one in which the expert explains that taking moments about the point he chose eliminates the most unknown variables.

Prompt three of homework eight asks, "To begin your solution you must make several assumptions. Which surfaces if any did you assume were on the verge of slip?" We have found that experts generate one of three different responses: *slip*, *verge*, or *no-slip*. Experts simply responded by indicating whether they assumed bodies in the diagram were slipping (*slip*), on the verge of slip (*verge*), or did not slip at all (*no-slip*).

Using the experts' self-explanations as a guide, we examined the students' responses. If a student's response to a prompt matched one of the types of responses used by the experts, we labeled the response as such. Otherwise, we labeled the response as *none*, indicating that it did not match any of the experts' responses.

Open Information Extraction Algorithm

For our analysis, we implemented the open IE algorithm developed by Soderland et al. [8] This technique learns a set of rules which maps self-explanations to content types. These rules comprise constraints on the existence of words in the self-explanations and the locations of those words. If the correct set of words exists in the correct locations, the rule assumes that a particular concept has been expressed.

This technique begins by using the TextRunner software package[9] to extract all noun phrases present in each self-explanation sentence. Noun phrases take the form of a tuple, (arg_1, *pred*, arg_2), where arg_1 is the subject, *pred* is the predicate, and arg_2 is the object.

A variety of words can be used to express the same concepts. For example, the phrases “on the verge of slip” and “impending slip” have the same meaning. To accommodate these sorts of variations, Soderland’s algorithm relies on lists of synonymous words. More precisely, it requires the identification of word classes, and the enumeration of the words within those classes. Table 1 lists the word classes we use in our analysis. For example, the “variable” class contains the various words that are frequently used to describe the unknown forces to be computed in a statics problem. These words include “variable,” “force,” “unknown,” and “component.” Note that identifying the existence of a concept in a self-explanation is more complex than simply identifying the existence of specific words. The relationship between the words is essential.

A rule learning process is used to learn these relationships. The rules attempt to infer the commonalities between different expressions of the same concept. Initially, the technique creates an overly-specific rule for each tuple. The rule, in effect, assumes that for another tuple to have the same meaning, it must have the same words in the same order. More precisely, the rule contains a constraint for every word class and preposition found in both that tuple and the sentence that contains it. The constraints govern both the existence and locations of those words. This technique recognizes five possible locations for word classes and prepositions: arg_1, *pred*, arg_2, the portion of the sentence preceding the tuple, and the portion proceeding. Each overly-specific rule will likely match only a few other tuples in the training data, if any. To find a more accurate rule, the technique

HWK-Prompt	Class Name	Words in Class
3-1, 3-4	Eliminate	cancel, eliminate, rid, ignore, took out, avoid
3-1	Contains	contains, touches
3-1	Need	need, require, necessary
3-1, 3-4	Variable	variable, force, unknown, component
3-4	Only	only
3-4	Only Unknown	only unknown, only one
3-4	Direct	direct, one step
3-4	Solve	solve, give, gave
8-3	Assumption	assume, occur, think, assumption
8-3	Slip	slip
8-3	FBD Component	block, point, arm, crate, brake, surface, member, box
8-3	Negative	wasn’t, isn’t, didn’t, not
8-3	Verge	verge, impend, about

Table 1: The word classes and the words they contain for the self-explanation prompts for the problems in homework assignments three and eight.

repeatedly relaxes constraints so that the rule has higher precision in identifying the concept.

Here, precision is defined as:

$$precision = \frac{true\ positives}{true\ positives + false\ positives}$$

where "true positives" are tuples that were correctly identified, and "false positives" are tuples that were incorrectly classified as this concept.

A beam search is used to find the version of a rule with highest precision. This search begins by dropping constraints from the overly-specific rule, one at a time, and computing the precision of each resulting, relaxed rule over the training set. The *k* most precise rules are kept, where *k* is called the beam width. The process repeats for each of the *k* relaxed rules. In our implementation, we use a beam width of 10. The process ultimately terminates when an empty rule is reached. The rule with highest precision evaluated during the search is kept as the final rule.

Results and Discussion

We performed leave-one-out cross-validation to train and test this technique. In each fold of cross-validation, the data from one subject (either an expert or a student) is selected for testing, and the data from the other subjects is used for training. In this way, the data used to train and test the system are never the same as each other.

Table 2 shows the accuracy results for identifying concepts for prompt one of homework three. Here, accuracy is defined as the fraction of self-explanations corresponding to a particular concept that were correctly identified as such. For example, the technique correctly identified 30 self-explanations that expressed the *needed-forces* concept, and incorrectly identified seven other self-explanations as expressing this concept. Thus, the technique achieved 81.1% accuracy at identifying this concept. Overall, the technique achieved 75.9% accuracy at identifying concepts used in the experts' self-explanations. Similarly, the technique achieved 70.2% accuracy at identifying self-explanations that contained none of the concepts used in the expert's self-explanations.

Concept	Correct	Incorrect	Accuracy
Needed Forces	30	7	81.1%
Only One	9	4	69.2%
Least Forces	2	1	66.7%
Alternative Difficult	0	1	0.0%
All Concepts	41	13	75.9%
None	52	22	70.2%

Table 2: Accuracy of concept recognition in self-explanations for prompt one of homework three. The "All Concepts" row of the table contains the overall accuracy for identifying self-explanations that contain a concept used in the expert's self-explanations. The "None" row is the accuracy for identifying self-explanations that contained none of the concepts used in the experts' self-explanations.

Concept	Correct	Incorrect	Accuracy
Directly Solves	4	2	66.7%
Only Unknown	6	2	75.0%
Eliminate Forces	40	1	97.6%
All Concepts	50	7	87.7%
None	70	32	68.6%

Table 3: Accuracy of concept recognition in self-explanations from prompt four of homework three. The "All Concepts" row of the table contains the overall accuracy for identifying self-explanations that contain a concept used in the experts' self-explanations. The "None" row is the accuracy for identifying self-explanations that contained none of the concepts used in the experts' self-explanations.

Table 3 contains the accuracy results for identifying concepts for prompt four of homework three. Overall, the technique achieved 87.7% accuracy at identifying concepts used in the experts' self-explanations. Similarly, the technique achieved 68.6%

accuracy at identifying self-explanations that contained none of the concepts used in the experts' self-explanations.

Finally, Table 4 contains the accuracy results for identifying concepts for prompt three of homework eight. Overall, the technique achieved 84.2% accuracy at identifying concepts used in the experts' self-explanations. Similarly, the technique achieved 97.3% accuracy at identifying self-explanations that contained none of the concepts used in the experts' self-explanations.

Concept	Correct	Incorrect	Accuracy
Slip	11	7	61.1%
No-slip	6	6	50.0%
Verge	63	2	96.9%
All Concepts	80	15	84.2%
None	36	1	97.3%

Table 4: Accuracy of concept recognition in self-explanations for prompt three of homework eight. The "All Concepts" row of the table contains the overall accuracy for identifying self-explanations that contain a concept used in the experts' self-explanations. The "None" row is the accuracy for identifying self-explanations that contained none of the concepts used in the experts' self-explanations.

This technique does perform better with more training data. For example, in Table 2, there were numerous examples of the *needed-forces* concept, and only one for the *alternative-difficult* concept. The technique performed accurately on the former and poorly on the latter.

This technique currently works with manual transcriptions. In order for this technique to work in a completely automated system the manual transcriptions will need to be replaced with automatically recognized handwriting made possible by techniques such as the image-based recognizer [10] or the dollar recognizer. [11] Future work will need to account for the errors that may be introduced by such processes.

Conclusion

We have presented a technique that is able to accurately identify whether a student's self-explanation contains the same concepts used in self-explanations generated by experts. The technique correctly identified the existence of such concepts with an accuracy that ranged from 75.9% to 87.7%. Similarly, the technique correctly identified the lack of such concepts with an accuracy that ranged from 68.6% to 97.3%.

The ultimate goal of our work is to build a tutoring system that engages students to create meaningful self-explanations of their work, thus honing their metacognitive skills and increasing their mastery of the subject. The techniques presented here are a first step toward achieving this goal. In the future, these techniques may be implemented within an interactive tutoring system, enabling it to determine if a student has provided meaningful self-explanation. Such a system may then prompt the student to continue his or her explanations when necessary.

Acknowledgement

This material is based upon work supported by the National Science Foundation under Grant No. 0935239.

References

1. T. H. Michelene; Miriam Bassok; Matthew W. Lewis; Peter Reimann; Robert Glaser Chi. Self-explanations: How students study and use examples in learning to solve problems. In *Cognitive Science*, pages 145–182, 1989.

2. Richard E. Mayer. Cognitive, meta-cognitive, and motivational aspects of problem solving. *Instructional Science*, 26(1-2):49–63, 1998.

3. A.L. Brown and J.D. Day. Macrorules for summarizing texts: The development

of expertise. *Journal of Verbal Learning and Verbal Behavior*, 22:1–14, 1983.

4. P.S. Steif, J. Lobue, A. L. Fay, and L. B. Kara. Improving problem solving performance by inducing talk about salient problem features. *Journal of Engineering Education*, pages 135–142, 2010.

5. Jerry R. Hobbs, Mabry Tyson, John Bear, and David Israel. Sri international: Description of the fastus system used for muc-4. In *Proceedings of the Fourth Message Understanding Conference (MUC-4)*, pages 268–275. Morgan Kaufmann, 1992.

6. Scott Miller, Michael Crystal, Heidi Fox, Lance Ramshaw, Richard Schwartz, Rebecca Stone, Ralph Weischedel, and The Annotation Group BBN Technologies. BBN: Description of the Sift system as used for MUC-7. *In Proceedings of the Seventh Message Understanding Conference (MUC-7)*, 1998.

7. Aron Culotta. Integrating probabilistic extraction models and data mining to discover relations and patterns in text. In *Proceedings of the HLT-NAACL-2006*, pages 296–303, 2006.

8. Stephen Soderland, Brendan Roof, Bo Qin, Mausam Shi Xu, and Oren Etzioni. Adapting open information extraction to domain-specific relations. *AI Magazine*, 31:93–102, 2010.

9. Oren Etzioni, Michele Banko, Stephen Soderland, and Daniel S. Weld. Open information extraction from the web. *Communications of the ACM*, 51:68–74, December 2008.

10. Levent Burak Kara. An image-based trainable symbol recognizer for sketch-based interfaces. In *AAAI Fall Symposium Series 2004: Making Pen-Based Interaction Intelligent and Natural*, pages 99–105. AAAI Press, 2004.

11. Jacob O. Wobbrock, Andrew D. Wilson, and Yang Li. Gestures without libraries, toolkits or training: a $1 recognizer for user interface prototypes. *In Proceedings of the 20th annual ACM symposium on User interface software and technology*, UIST '07, pages 159–168, New York, NY, USA, 2007. ACM.

Biographical Information

James Herold earned his B.S. in Computer Science at California Polytechnic State University, Pomona in 2004. He is currently a Ph.D. student in Computer Science at the University of California, Riverside.

Thomas Stahovich received a B.S. in Mechanical Engineering from the University of California, Berkeley in 1988. He received a S.M. and Ph.D. in Mechanical Engineering from the Massachusetts Institute of Technology in 1990 and 1995, respectively. He is currently Chair and Professor in the Mechanical Engineering Department at the University of California, Riverside.