From Arguments to Hints – A Didactic Perspective on Deductive Reasoning

Helmut Horacek
Universität des Saarlandes, FR 6.2 Informatik
Postfach 151150, D-66041 Saarbrücken, Germany
email: horacek@cs.uni-sb.de

Abstract

In intelligent tutoring systems (ITS), the effectiveness of hints plays a central role for the learning task. However, the automated generation of hints is still weak in a methodological sense, since this process is typically domain- and task-dependent to a large degree. Through relating hints to an elaborate explanation model, we intend to put the generation of these communicative acts on better motivated grounds. We present preliminary elaborations for a subarea of mathematics, elementary set theory, based on results of extensive Wizard-of-Oz experiments. These investigations contribute towards reducing the domain- and task-dependency of ITS, which is a main prerequisite for building future ITS more efficiently.

1. Introduction

Supported by the strong public interest in educational affairs, interest in intelligent tutoring by machines became increasingly prominent recently. Among others, intelligent tutoring systems have been built for technical domains, such as physics and mathematics. In such a tutorial setting, the effectiveness of hints is known to play a central role for the learning task, which has been demonstrated by studies on human tutoring [Moore 2000]. However, the automated generation of hints is still weak in methodological terms, since this process is typically domain- and task-dependent to a large degree. We intend to put the generation of these communicative acts on better motivated grounds, which we envision through relating hints to an elaborate explanation model. This relation is established by focusing on selected aspects of explanations and abstracting them to elaborate generalities for hinting purposes. We present elaborations for a subarea of mathematics, elementary set theory.

This paper is organized as follows. We first provide some background information about our investigations in dialogs for tutoring theorem proving. Then we briefly review our model of explanation. We follow with the method of building hint specifications out of these explanations, illustrated by some examples. Finally, we sketch future developments.

2. Background

Several computer tutor programs have developed a variety of hinting strategies. Gernter, Conati and VanLehn [1998] have developed a tutoring system named Andes. Their system has aninterface, Andes' Procedural Helper, which uses templates to generate hints from nodes in the solution graph. The LISP tutor of Anderson, Corbett, Koedinger, and Pelletier [1995] generates hints from production rules and templates. The CIRCSIM- Tutor developed by Zhou et al. [1999] uses heuristics to choose a hinting strategy that is sensitive to the type of student's answer, a tutorial plan, and the tutoring history. In all of these systems, hints are presented with graduated specificity, starting with minimal clues and clues and ending with direct assertions. Hints present facts, ask leading questions or reframe the problem. AutoTutor [Graesser et al. 1999] uses the hints to redirect the student back to the topic that was addressed in the question by stimulating the recall of specific information or by facilitating the student's inferential processing. The two explicit functions of a hint that have been integrated in the tutoring program are (a) to redirect students to the relevant topic, and (b) to direct students to a specific aspect of the answer to a question.

Our investigations are part of the dialog project [Pinkal et al. 2001], which is a joint effort by the computer science and computational linguistics departments to model natural language tutorial dialogs about conducting proofs of mathematical theorems. In comparison to the specific tasks addressed by others, such as formalizing natural language descriptions of numerical relations by algebraic expressions and describing geometrical relations in regular triangles properly (e.g., [Heffernan and Koedinger 2000]), the issue of teaching proving techniques is considerably more complex. Because of that, our primary emphasis was in specifying tutorial strategies in terms elaborate enough for serving as prescriptions to be carried out by a human tutor, and testing the suitability of these strategies by controled Wizard-of-Oz experiments. Through these investigations, which were supported by a tool [Fiedler and Gabsdil 2002], we were able to collect a corpus for our tutorial task, and we have obtained some evidence about the effectiveness of the strategies we have used to teach the required skills.

24 subjects with varying educational background and prior mathematical knowledge ranging from little to fair participated in the experiment. The experiment consisted of several phases: (1) *preparation and pre-test* on paper, (2) *tutorial session* proper, and (3) *post-test and evaluation questionnaire*, on paper again. The subjects had to prove theorems, by appling the De Morgan laws ($K((A \cup B) \cap K(C \cup D)) = (K(A) \cap K(B)) \cup (K(C) \cap K(D))$), and by making use of the concepts power set ($A \cap B \in P((A \cup C) \cap (B \cup C))$) and complement (When $K(B) \supseteq A$, then $K(A) \supseteq B$, where K signifies *K*omplement in German). The subjects were instructed to enter steps of a proof rather than the complete proof as a whole, in order to enable a dialog with the system. The tutor-wizard's task was to respond to the student's utterances following a given algorithm. The wizard first classified the completeness, accuracy, and relevance of the subject's utterance with respect to a valid proof of the theorem at hand. Then the wizard decided what dialog moves to make next and verbalized them.

In our experiments, the human tutor had to rely on specifications for two alternative strategies: (1) the *socratic* one, which is based on eliciting information from the student through providing hints about the problem solving situation, and (2) the *didactic* one, which consists in exposing the course of problem-solving through piece-wise explanations. Comparative studies in several domains have demonstrated that the socratic strategy is superior to the didactic one, since it has a better long-term effect in preserving the acquired skills, and it also leads to deeper understanding on behalf of the student (for example, [Ashley et al. 2002]).

In our experiments, it turned out that these benefits of the socratic method could not be substantiated. Through analyzing the dialog protocols and the recorded video tapes in relevant situations, we observed that the hints produced by the tutor were not perceived as particularly helpful by the subjects. In fact, the tutor was given specifications to merely refer to a domain concept or to a relation relevant for the next problem-solving step (e.g., "use the power set"). This was mostly known to the subjects, but they were missing information about *how* to apply these terms. This led us to the conclusion, that the proper specification of hints requires a more in-depth analysis.

3. Explanation Model

In order to come up with better suitable hint specifications, we intend to investigate relations to explanations of the tasks to be accomplished, which were used for the didactic strategy. A crucial assumption behind this approach is that whenever a subject is able to fully understand an explanation, he is in principle able to construct this explanation for himself, provided he is aware of the relevant key information. This awareness, in turn, is exactly what should be triggered by appropriate hints. In order to achieve that in an operational approach, explanations need to be built, with key information extracted. Moreover, this material needs to be reformulated for purposes of hinting, which is not precisely the explanation content, but some more abstract and related descriptions that aim at triggering awareness about the ingredients of the underlying reasoning rather than properly exposing them.

The basis for such an investigation is an elaborate and operational explanation model, which we have built for our domain of consideration [Horacek 1999]. Its generality has also been demonstrated for everyday situations [Horacek 1998], and introduced to the argumentation community [Fiedler and Horacek 2001, 2002]. It takes into account specificities of human argumentation in order to bridge the gap between machine-based and human-oriented deductive reasoning. The most important concepts in the associated construction process are

- The determination of an adequate *level of granularity*, which has been termed *assertion level* in [Huang 1994] – in mathematics, this is generally the level of arguing about applications of axioms. It is made more fine-grained to the *partial assertion level* [Horacek, 1999], with more detailed descriptions for cognitively complex inferences, such as modus tollens, according to psychological insights by [Johnson-Laird and Byrne, 1990]. For example, adequately describing the use of transitivity in reverse (modus tollens) direction requires exposing more inferencing detail – indicating the intermediate result after the proper modus tollens inference step, followed by a disjunction elimination.
- The selection of an adequate *degree of explicitness*, which tends to omit known and easily inferable information from descriptions of inferences (see [Horacek, 1998]). This comprises known facts, evident regularities, and trivial intermediate inference steps, again based on empirical findings [Thüring and Wender 1985]. In the mathematical domain, typical omissions are implicit uses of commutativity and associativity, which our experiments have demonstrated, too.

Explanations produced in this manner are then used for the purpose of generating hints.

4. Building Hints

The notion of a hint is generally conceived as some piece of information that does not directly belong to the solution in question, but is likely to put the student in a better position to uncover the solution than this would be the case without the hint. Hence, concerning the information content provided, hints differ in two basic respects from explanations (described from the perspective of deductive tasks):

- *Partiality*
  Whereas explanations of deductive inferences comprise premises, the rule applied, and the conclusion, a hint typically addresses only one of these ingredients.
- *Imprecision*
  Whereas explanations address all necessary ingredients in detail according to the chosen level of granularity (those that need to be mentioned explicitly), hints only provide a generalized or related reference to a specific solution aspect.

Building hints on the basis of these explanations is done by an interplay between a repertoire of hint descriptions and eventually abstracted portions of the explantions. A choice

conceptual hints	
a condition of a conceptual property	
a consequence of a conceptual property	
functional hints – procedural method	
simplify an expression (reduce the numbers of operators)	
reduce the number of parentheses	
eliminate a variable (in the conclusion of an inference)	
functional hints – pattern-based method	
bring two variables closer together	
isolate a variable	
build an expression with a specific subpattern	

Figure 1. Examples for hint descriptions

among potential candidates is then made according to the hinting strategy [Fiedler and Tsovaltzi, 2003], which takes into account the discourse and tutorial context. For building hints, a distinction is made between a *conceptual* view, which emphasizes a relation to a mathematical concept, and a *functional* view, which emphasizes the effect imposed on the conclusion of an axiom under consideration. We have defined a few hint descriptions for both views, see Figure 1.

Our understanding of a conceptual view encompasses axioms which relate a central property of a mathematical concept to some assertion other than a purely taxonomic relation to a more general or more specific concept. Depending on the direction of the implication, that axiom expresses a condition for the property of the mathematical concept under consideration, or a consequence. In formal terms, let $f(P(x))$ be a central property f of a mathematical concept P, and $Q(x)$ the assertion related to it. Then $f(P(x)) \Leftarrow Q(x)$ can be used to build a hint description for a condition of a conceptual property, and $f(P(x)) \Rightarrow Q(x)$ for a hint about the consequence of a conceptual property.

For the functional view, the applicability of some descriptions is tested by comparing structural properties of the premise and conclusion of an inference rule: the number of operators, the number of parentheses, and the appearance of a variable in the hint descriptions *simplify*, *reduce the number of parentheses*, and *eliminate a variable*, respectively. The suitability of the other descriptions is defined by patterns that appear in the conclusion but not in the premise of the axiom. These patterns make use of wild card characters which represent abstractions of subformulas. For example, the pattern for *bringing two variables closer together* is '$X * Y$', in both premise and conclusion, where the abstraction underlying '$*$' in the conclusion must be weaker than that in the premise. Moreover, the pattern for *isolate a variable* is '$X *$', to which the conclusion but not the premise must be reducible by abstraction. Finally, some subexpression must appear in the conclusion but not in the premise of a rule for *build an expression with a specific subpattern* to be applicable. This subexpression, however, can also be a partial abstraction of the concrete formula, provided its identification in that formula is still possible.

In order to build specifications for hints on the basis of explanations, an abstraction process on parts of the explanation is performed. The operations for this process depend on the view taken on the inference in question. For the conceptual view, the conceptual elements appearing in that rule are generalized, according to the underlying domain taxonomy. This abstraction can be used in a pure form, as the basis for a rather general hint. Moreover, some of the subformulas present in the concrete instance can be embedded in the abstracted form, to be used as more specific follow-up hints. For the functional view of an inference, the structure of the formula is subject to the abstraction (in our domain). More concretely, atomic elements in a formula (variables, constants, and operators) are successively abstracted into some sort of wild character place holder. Moreover, two immediately adjacent wild characters can be combined into a single one. This way, building formulas reduced to their structural embeddings is possible, with selected variables still present. Intuitively, some intermediate form seems to be most appropriate for generating a hint on that basis.

Let us illustrate the potential uses of this method by an example, which is taken from our Wizard-of-Oz experiments. The theorem to be proved is the following (P stands for power set):

$$(A \cap B) \in P((A \cup C) \cap (B \cup C))$$

In addressing such a problem, hints aim at eliciting individual inference steps. Two essential steps in one of the possible proofs to this theorem are

(1) application of distributivity to the expression in the scope of the power set operator, and

(2) deriving the relation between the set on the left side of the $\in$ operator and the set in the scope of the power set operator (this relation is the subset relation).

When focusing on the conclusion obtained after applying distributivity in (1), which is $(A \cap B) \cup C$, some of the possible and most useful abstractions for the present operation are: $((A \cap B) \cup *)$, $((A \cap B) *)$, $((A * B) *)$, $((*) \cup *)$. The last two fit the hint descriptions "Try to bring A and B closer together" (the last-but-one), and "Reduce the number of parentheses" or "Simplify" (the last one). These hints are much more in the spirit of the socratic strategy than simply specifying the goal, which gives away the underlying inference step (as did the didactic strategy).

Focusing on the relation between the two sets according to (2), the underlying rule can be formulated as $x \in P(y) \Leftrightarrow y \supseteq x$, which is essentially how the notion of power set is defined. The most accurate conceptualization for this relation is "the condition for belonging to a power set". Through generalization, this can be weakened into "the condition for belonging to a set", and simply "the condition". Conversely, these expressions can be augmented by the expressions to which x and y are instantiated in this example: "the condition for $A \cap B$ belonging to $P((A \cup C) \cap (B \cup C))$", and simply "the condition for $A \cap B$ being related to $(A \cup C) \cap (B \cup C)$".

We are convinced that these utterances are more focused and more likely to be successful as hints than, e.g., the straightforward ”Think of using the power set axiom”, which appeared in our experiments.

5. Future Work

So far, our investigations are still in a preliminary stage. In order to ultimately build a suitable subsystem for the tutorial environment, three activities must be addressed:

- The model of building abstractions of the kind described in the previous section needs to be made precise enough to make it fully operational.
- Moreover, metrics need to be designed which enables one to choose among the various abstractions built according to their likely effectivity in a tutoring task.
- Complementing the results obtained, the formulation of hints on the basis of these specifications needs to be done, which is some sort of a generation process that can be tailored to the specificities of the task at hand (the requirement that a hint and no other speect act is the task to do, as well as incorporating specificities of the domain).

Finally, with a reasonably developed subsystem, we are in a position to carry out a new series of Wizard-of-Oz experiments, with a more elaborate hint generation component and, possibly, a number of other new features, which are either operationalized or conceptually improved in relation to the first series of experiments. This will enable us to test the effectivity of hints and other improvements within the environment of the overall tutorial task.

References

Anderson, J., Corbett, A., Koedinger, K., and Pelletier, R. 1995. Cognitive Tutors: Lessons Learned. In *Journal of the Learning Sciences* 4 (2), pp. 167-207.

Ashley, K., Desai, R., and Levine, J. 2002. Teaching Case-Based Argumentation Concepts Using Dialectic Arguments vs. Didactic Explanations. In Proc. of the *6th International Conference on Intelligent Tutoring Systems,* pp. 585-595.

Fiedler, A., and Gabsdil, M. 2002. Supporting Progressive Refinement of Wizard-of-Oz Experiments. In Proc. of the *6th International Conference on Intelligent Tutoring Systems – Workshop W6: Empirical Methods for Tutorial Dialogue Systems*.

Fiedler, A., and Horacek, H. 2001. Argumentation in Explanations to Logical Problems. In *International Conference on Computational Science*, San Francisco.

Fiedler, A., and Horacek, H. 2002. Argumentation within Deductive Reasoning. *Workshop on Computational Models of Natural Argument at ECAI-2002*, Lyon.

Fiedler, A., and Tsovaltzi, D. 2003. Automating Hinting in Mathematical Tutorial Dialogue. In Proc. of the *EACL-03 Workshop on Dialogue Systems: Interaction, Adaptation and Styles of Management*, Budapest.

Gertner, A. S., Conati, C., and VanLehn, K. 1998. Procedural Help in Andes: Generating Hints using a Bayesian Network Student Model. In *AI and Education*.

Graesser, A. C., Wiemer-Hastings, K., Wiemer-Hastings, P., Kreuz, R.J., and the Tutoring Research Group. 1999. AutoTutor: A Simulation of a Human Tutor. Journal of *Cognitive Systems Research*, 1, pp. 35-51.

Heffernan, N. and Koedinger, K. 2000. Intelligent Tutoring Systems are Missing the Tutor: Building a More Strategic Dialog-Based Tutor. In Papers from the *2000 AAAI Fall Symposium on Building Dialogue Systems for Tutorial Applications*.

Horacek, H. 1998. Generating Inference-Rich Discourse Through Revisions of RST-Trees. In *Proc. of AAAI-98,* pp. 814-820.

Horacek, H. 1999. Presenting Proofs in a Human-Oriented Way. In *Proc. of CADE-99*, pp. 142-156, Trento, Italy.

Huang, X. 1994. Reconstructing Proofs at the Assertional Level. In *Proc. of CADE-94*, pp. 738-752.

Johnson-Laird, P., and Byrne, R. Deduction. Ablex Publishing, 1990.

Moore, J. 2000. What Makes Human Explanations Effective? In Proc. of the *Fifteenth Annual Conference of the Cognitive Science Society*, Hillsdale NJ. Earlbaum.

Pinkal, M., Siekmann, S., and Benzmüller, C. 2001. Projektantrag Teilprojekt MI3 – DIALOG: Tutorieller Dialog mit einem mathematischen Assistenzsystem. In Fortsetzungsantrag SFB 378 – Ressourcenadaptive kognitive Prozesse, Saarbrücken, Germany.

Thüring, M., and Wender, K. 1985. Über kausale Inferenzen beim Lesen. In *Sprache und Kognition* 2, pp. 76-86.

Zhou, Y., Freedman, R., Glass, M., Michael, J. A., Rovick, A. A., and Evens, M. W. 1999. What should the Tutor Do When the Student Cannot Answer a Question? In Proc. of the *12th Florida Artificial Intelligence Symposium (FLAIRS-99)*, Orlando, FL.