Proceedings of the IASTED International Conference
Artificial Intelligence and Soft Computing
August 9-12, 1999, Honolulu Hawai'i, USA

Representational and Advisory Guidance for Learning: Alternate Roles for AI

Daniel D. Suthers
Department of Information and Computer Sciences, University of Hawai'i
1680 East West Road, POST 303A, Honolulu, HI 96822
suthers@hawaii.edu

Abstract. Although applications of artificial intelligence to education have traditionally focused on teacher-centered approaches such as Intelligent Tutoring Systems, artificial intelligence can also contribute to student-centered approaches to learning by providing representational and interactive support for collaborative learning. Specifically, AI can inform the design of representational systems that constrain and guide learner's activities, and enable dynamic generation of guidance based on representational artifacts that learners construct in these systems. The paper exemplifies such contributions with an educational software package, known as "Belvedere" that supports students collaboratively solving ill-structured problems in science as they develop critical inquiry skills.

1. Belvedere

Decades of research into cognitive and social aspects of learning [2] has developed a clear picture of the importance of learners' active involvement in the expression, examination, and manipulation of their own knowledge, as well as the equal importance of guidance provided by social processes and mentorship. The development of the "Belvedere" software reflects this trend. Belvedere is a networked software system [8] that provides learners with shared workspaces for coordinating their collaboration in scientific inquiry [9]. Belvedere's core functionality is a diagramming window in which students construct "evidence maps" – graphs in which *nodes* represent statements (primarily empirical observations or hypotheses) within a scientific debate or investigation; and *links* represent the relations between the elements, i.e., consistency or inconsistency. The software also includes artificial intelligence advisors, a "chat" facility for unstructured discussions, and facilities for integrated use with Web browsers. The diagramming window is shown in Figure 1. The default "palette" (the horizontal row of icons) makes salient the most crucial distinctions we want learners to acquire in order to conduct scientific inquiry. Left to right, the icons are "data" for empirical statements, "hypothesis" for theoretical statements, and "unspecified" for others statements about which learners disagree or are uncertain; then links representing "for" and "against" evidential relations. The rightmost icon invokes the automated advisors.

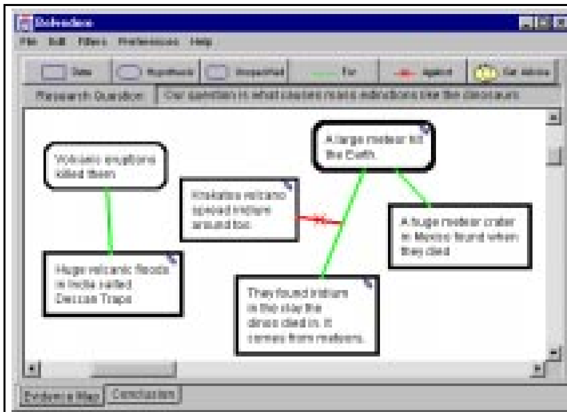


Figure 1. Belvedere Evidence Mapping Software

2. Representations and Discourse

An early version of Belvedere was designed to determine whether students can learn the nuances of scientific argument if provided with a visual "argument-mapping" language capable of capturing all of these nuances, along with an intelligent coach that interacts with and guides the student. A crucial assumption was that students would express their arguments in the language. However, we found that much of students' relevant argumentation was "external," arguing from the representations rather than arguing in the representations. Faced with a decision concerning some manipulation of the representations, students would begin to discuss substantial issues until they reached tentative agreement concerning how to change the representation. In the process, statements and relations we would have liked students to represent went unexpressed. Recognizing this as an opportunity, we subsequently downplayed the originally intended roles of the representations in favor of their role as a stimulus and guide for collaborative learning discourse. This perspective led to consideration of the role of representational bias in shaping learning activities.

2.1 Representational Bias

Representational tools are artifacts (such as software) with which users construct, examine, and manipulate external representations of their knowledge. A representational tool is an implementation of a *representational notation* that provides a set of primitive elements out of which representations can be constructed. Developers choose a representational notation and instantiate it as a representational tool, while the user of the tool constructs particular *representational artifacts* in the tool. We are concerned with interactions between learners and other learners, specifically verbal and gestural interactions termed *collaborative learning discourse*.

Each given representational notation manifests a particular representational bias, expressing certain aspects of one's knowledge better than others [11]. Representational bias manifests in two major ways: Constraints: limits on logical expressiveness [7]; and Salience: how the representation facilitates processing of certain knowledge units, possibly at the expense of others [4, 13]. Representational tools mediate collaborative learning discourse by providing learners with the means to express emerging knowledge in a persistent medium, inspectable by all participants, where the knowledge then becomes part of the shared context. Representational bias constrains the knowledge that can be expressed in the shared context, and makes some of that knowledge more salient and hence a likely topic of discussion.

2.2 Ontological Bias as Constraint

Belvedere requires all knowledge units (statements and relations) to be categorized at the time of creation. We observed that learners who were using Belvedere often initiated discussion of the appropriate categorical primitive for a given knowledge unit when they were about to represent that unit. Although this is not surprising, it is a potentially powerful guide to learning. In some cases, the choice forced by the tool led to a peer-coaching interaction on a distinction that was critically important for how they subsequently handled the statement. Yet it is not always useful to confront learners with choices, even if they may become important at some point in the development of expertise. With more complex sets of primitives, we sometimes observed students becoming confused by choices that were not relevant at their stage of learning.

Based on these observations, we simplified Belvedere's representational framework to focus on the most essential distinction needed concerning the epistemological source of statements: empirical ("data") versus hypothetical ("hypothesis"). Further simplifications were motivated by observations concerning the use of relations (links). The original set of argumentation relations included evidential, logical, causal, and rhetorical relations as well as the various classifications of statements exemplified above. In exchanges similar to the previous example, we observed students' confusion about which relation to use. Sometimes more than one applied. We felt that the ontologically mixed set of relations confused students about what they were trying to achieve with the diagrams, and did not help them focus on learning key distinctions. In order to encourage greater clarity, we decided to focus on evidential reasoning, and specifically on the most essential relational distinction for evidence based inquiry: whether two statements are consistent or inconsistent.

At one time there were at least three versions of the "consistency" relation: "predicts" and "explains" (both drawn from hypotheses to data), and "supports" (drawn from data to hypotheses). Early versions of our evidence pattern coach (described later) attempted to reason about and even enforce these semantics. However, we found that users' use of these relations (as expressed in their links) was inconsistent and sometimes differed from the intended semantics. When the users' semantics differed from the coach's semantics, confusion or frustration resulted. The use of "predicts," "explains," and "supports" links was misguided not only because different agents had different semantics for them, but also because the links were "surface" level discourse relations that did not encourage learners to think in terms of the more fundamental consistency relationships. Whether a hypothesis predicts or explains a datum is an artifact of the chronology of the datum with respect to statement of the hypothesis. Whether one uses "supports" or one of the other two links is an artifact of the focus of the discourse process by which the diagram is being constructed (argumentation about hypotheses versus explanation of data). Hence we eliminated these in favor of a single non-directional relation that expresses the more fundamental notion of evidential consistency.

To summarize, a representational notation provides a set of primitive elements out of which representational artifacts are constructed. These primitive elements constitute an "ontology" of categories and structures for organizing the task domain. The present hypothesis claims that learners will see their task in part as one of making acceptable representational artifacts out of these primitives. Thus, they will search for possible new instances of the primitive elements, and hence (according to this hypothesis) will be biased to think about the task domain in terms of the underlying ontology.

2.3 Salience of Represented and Missing Units

In working with Belvedere we found suggestive evidence that salience of information in conjunction with task requirements may guide discourse. For example, Figure 2 outlines a diagram state in which three statements were clustered near each other, with no links drawn between the statements. One student pointed to two statements simultaneously with two fingers of one hand, and drew them together as she gestured towards the third statement, saying "Like, I think that these two things, right here, um, together sort of support that" (videotape of an early laboratory study of Belvedere).

This event is an example of how external representations facilitate the expression of complex ideas [1]. However, this observation applies to any external representation. Several features of the particular representational system in use may have made the

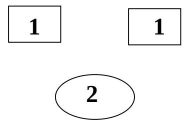


Figure 2. Gesturing to express a relationship

student’s utterance more likely. First, elaboration on these particular statements is more likely because they (instead of others) are expressed as objects of perception in the representation. Second, this event is more likely to occur in a representational environment that provides a primitive for connecting statements with a support relation than in one that does not: the students perceive their task as one of linking things together. Third, it may have been easier to recognize the relationship between the three statements because they happened to be spatially nearby each other [4]. In this example, proximity was determined by the users rather than intrinsic to the representational toolkit. However, a representational tool could constrain proximity based on potential relationships between knowledge units.

We concluded that learners will be more likely to attend to, and hence elaborate on, the knowledge units that are perceptually salient in their shared representational workspace than those that are either not salient or for which a representational proxy has not been created. The visual presence of the knowledge unit in the shared representational context serves as a reminder of its existence and any work that may need to be done with it. Also, it is easier to refer to a knowledge unit that has a visual manifestation, so learners will find it easier to express their subsequent thoughts about this unit than about those that require complex verbal descriptions. To the extent that two representational notations differ in kinds of knowledge units they make salient, the representations will encourage elaboration on different kinds of knowledge units.

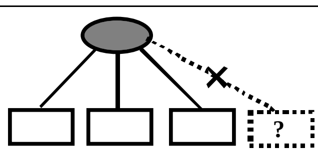
Some representational notations provide structures for organizing knowledge units, in addition to primitives for construction of individual knowledge units. Unfilled “fields” in these organizing structures, if perceptually salient, can make missing knowledge units as salient as those that are present. If the representational notation provides structures with predetermined fields that need to be filled with knowledge units, salience predicts that learners will be more likely to search for and discuss the corresponding information.

3. Design of Computer Advisors

We also redirected our quest for computer intervention in human learning processes. An advisor that fully understands students’ argumentation and provides advice based on a deep understanding of the domain of inquiry would require substantial knowledge engineering, and would mandate a role of representations in discourse that is inconsistent with the observations just reported. Instead we sought to determine how we might provide useful advice while minimizing the amount of knowledge engineering required on the part of both users and developers. In this section we discuss two methods of advice generation that we have implemented.

3.1 Evidence Pattern Strategies

The first approach [6] gives advice in response to situations that can be defined on a purely syntactic basis, using only the structural and categorical features of the students’ argument graphs. Principles of scientific inquiry are instantiated as patterns to be matched to the diagram and textual advice to be given if there is a match. An example advice pattern from our Belvedere 2.0 implementation is given in Figure 3. Implemented versions of the system included about 20 different such patterns. When the solid-lined portions are present and the dashed portions are missing, the corresponding advice can be given. Objects that bind to variables in the patterns (shaded in Figure 3) are highlighted in yellow during presentation of advice to indicate the target(s) of definite references such as “this hypothesis.”




```
(def-advice ‘CONFIRMATION-BIAS
  :query ‘(retrieve (?h)
           (and  (hypothesis ?h)
                 (Exists-Multiple-Consistent-DataP ?h)
                 (Multiply-LinkedP ?h)
                 (fail (Exists-Inconsistent-DataP ?h))))
  :advice (“You’ve done a nice job of finding data that is consistent with this hypothesis.
However, in science we must consider whether there is any evidence *against* our
hypothesis as well as evidence for it. Otherwise we risk fooling ourselves into believing a
false hypothesis. Is there any evidence against this hypothesis?”)
  :advice-types ‘(cognitive-bias))
```

Figure 3. Evidence Pattern Advice

Typically, several advice patterns will match an evidence map, sometimes with multiple matches per pattern. This is more than a student can be expected to absorb and respond to at one time. It is necessary to be selective in a context sensitive manner. Selection is performed by a preference-based quick-sort algorithm that discriminates potential advice base on factors such as prior advice that has been given, how recently the object of advice was constructed and by whom, and various categorical attributes of the applicable advice. After sorting, a redundancy filter is applied that removes all but one of multiple instantiations of a given advice pattern, retaining the highest priority instantiation. This provides the final prioritized list of advice. The advice-on-demand version of the advisor then sends the first advice on the list to the requesting client. If further advice is requested before the diagram changes, subsequent advice instances on the sorted list are used without reanalysis.

3.2 Expert Path Advisor

The evidence pattern advisor provides advice about abstracted patterns of relationships among statements, but has nothing to say about the contents of these statements. The expert-path advisor was designed to offer specific information that the student may not discover on her own. It makes the assumption that a correspondence can be found between statements in a student's evidence map and those in a pre-stored expert's evidence map. The path advisor searches the latter "expert graph" to find paths between units that students have linked in their evidence maps, and selects other units found along those paths that are brought to the students' attention. Our claim is that this enables us to point out information that is relevant at a given point in the inquiry process without needing to pay the cost of a more complete semantic model of that information, such as would be necessary in traditional knowledge-based tutoring systems. The only costs incurred are in the construction of the "expert diagram" consisting of semantic units that are also available to the student.

The expert advisor was implemented with an A* best-first heuristic search in Belvedere 2.0 [10]. The search finds an optimal path from the start node to the goal node in the expert diagram according to the following cost heuristics. (1) Shorter paths are given lower costs, as more direct relationships are less likely to lead to obscurely related information. This heuristic takes precedence over the following two. (2) Paths that contradict the student's link are preferred, to address the confirmation bias. (3) Paths with more than one against link are given higher costs than other paths. Experience showed that the meaning of such paths is unclear to users. Once a lowest-cost path is found between the start and the goal statements, advice is generated as follows. When the expert diagram has a direct link between the start and the goal, simple feedback is generated based on a comparison to the student's link, either reinforcing or asking the student to reconsider the link. When a nontrivial path is found between the start and the goal, the advisor can confront the student with information that may contradict or corroborate the student's link. This information is selected from those nodes in the path found in the expert graph that do not also exist in the student's graph. It remains for the preference mechanism discussed previously to decide when the generated advice is actually worth giving. One preference was added to promote expert path advice over others, because this advice is more specific to the situation at hand than the evidence-pattern advice. This arbitration scheme can easily be extended to manage additional sources of advice.

3.2.1 Comparison of Advisors

The evidence-pattern advisor can make suggestions to stimulate students' thinking with no knowledge engineering required on the part of the teacher or domain expert. However, the advice is very general. It could better address the confirmation bias by confronting students with discrepant information they may be ignoring. The expert-path advisor can provide students with assistance in identifying relevant information that they may not have considered. The pattern-based advisor cannot provide this assistance, because it requires a model of evidential relationships between the units of information being manipulated by students. With the expert-path advisor, we have shown this assistance can be provided without deep modeling of or reasoning about the domain.

An attractive option is to combine the two advisors reported herein. Patterns could be matched to both student and expert diagrams to identify principled ways in which students might engage in additional constructive inquiry along with information that is relevant to that inquiry. For example, if the pattern matches the expert's graph but one pattern component is missing in the student's graph, the advisor could then present this information as indicated by the missing component's role in the pattern.

4. Conclusions

The phrases "Artificial Intelligence and Education" or "Intelligent Tutoring Systems" most immediately bring to mind the endeavor to build smart machines that teach. Ideally, such machines would "know" a great deal about a particular subject matter, being able to both articulate concepts and principles and engage in expert level problem solving. They would also know about pedagogy, being able to track the progress of individual students and choose the best feedback strategies and trajectory through a curriculum for a particular student [12]. This vision of AI&ED might be termed "strong AI&ED." Although work on "traditional" intelligent tutoring systems continues with a recent emphasis on agent-based systems, other work that does not fall within mainstream AI approaches is increasingly appearing in the AI&ED and ITS conferences.

Some of this work (e.g., [5] and the automated advisor described herein) can be characterized as "minimalist AI&ED." Instead of attempting to simulate a teacher and/or model the minds of students, these efforts provide machines with minimal abilities to respond (in a manner believed to be educationally relevant) to the semantics of student activities and constructions. This research tests the educational value of these minimal abilities, and adds functionality as needed to address deficiencies in the utility of the system. As a research strategy, this incremental approach ensures that we understand the capabilities and limitations of each representational and inferential device unencumbered by the simultaneous complexities of an attempted complete pedagogical agent.

A newly emerging third category of AI&ED work does not attempt to build reasoning machines, even of the minimalist sort, yet which constitutes a contribution of AI to education, and potentially even a source and test-bed of AI ideas. This kind of application can be seen most clearly in the design of representational systems. An artificial intelligence sensitivity to the properties of formal representations for automated reasoning can be applied to the analysis and design of external representations for human reasoning as well as machine reasoning. One revisits the notions of epistemological and heuristic adequacy, but now the interpreter is human and "representational bias" includes a perceptual component [4, 13]. The AI "in" software systems built under this approach is residual, influencing the design but being a run-time factor only for human rather than artificial agents. Examples of work in this category include [3, 7] and the present work.

5. References

[1] Clark, H.H. & Brennan, S.E. (1991). Grounding in Communication. In L.B. Resnick, J.M. Levine and S.D. Teasley (eds.), *Perspectives on Socially Shared Cognition,* American Psychological Association, 1991, pp. 127-149.

[2] Greeno, J. G., Collins, A. M., & Resnick, L. B. (1996). Cognition and Learning. In D. C. Berliner & R. C. Calfee (Eds.), *Handbook of Educational Psychology* (pp. 15-46). New York: Simon & Schuster Macmillan.

[3] Koedinger, K. (1991). On the design of novel notations and actions to facilitate thinking and learning. *Proc. Int. Conference on the Learning Sciences,* pp. 266-273. Charlottesville, VA: Association for the Advancement of Computing in Education. 1991.

[4] Larkin, J. H. & Simon, H. A. (1987). Why a diagram is (sometimes) worth ten thousand words. Cognitive Science 11(1): 65-99. 1987.

[5] Nathan, M. J. (1998). Knowledge and situational feedback in a learning environment for algebra story problem solving. Interactive Learning Environments, 5, 135-159.

[6] Paolucci, M., Suthers, D., & Weiner, A. (1996). Automated advice-giving strategies for scientific inquiry. *Intelligent Tutoring Systems, 3rd International Conference,* Montreal, June 12-14, 1996.

[7] Stenning, K. & Oberlander, J. (1995). A cognitive theory of grahical and linguistic reasoning: Logic and implementation. *Cognitive Science 19(1):* 97-140. 1995.

[8] Suthers, D. & Jones, D. (1997). An architecture for intelligent collaborative educational systems. *AI-Ed 97, the 8th World Conference on Artificial Intelligence in Education,* Kobe, August 20-22, 1997.

[9] Suthers, D., Toth, E., and Weiner, A. (1997). An Integrated Approach to Implementing Collaborative Inquiry in the Classroom. *Proc. 2nd Int. Conf. on Computer Supported Collaborative Learning (CSCL'97),* Toronto, December 10-14, 1997. pp. 272-279.

[10] Toth, J., Suthers, D., and Weiner, A. (1997). Providing expert advice in the domain of collaborative scientific inquiry. *8th World Conference on Artificial Intelligence in Education (AIED'97),* Kobe, August, 1997.

[11] Utgoff, P. (1986). Shift of bias for inductive concept learning. In *R. Michalski, J. Carbonell, T. Mitchell (Eds.) Machine Learning: An Artitificial Intelligence Approach, Volume II,* Los Altos: Morgan Kaufmann 1986, pp. 107-148.

[12] Wenger, E. (1987). Artificial Intelligence and Tutoring Systems: Computational and Cognitive Approaches to the Communication of Knowledge. Los Altos: Morgan Kaufmann.

[13] Zhang, J. (1997). The nature of external representations in problem solving. *Cognitive Science, 21(2)*: 179-217, 1997.