Abductive Theorem Proving for Analyzing Student Explanations

Pamela W. JORDAN and Maxim MAKATCHEV and Kurt VANLEHN*
Learning Research and Development Center, University of Pittsburgh
pjordan@pitt.edu

Abstract. The Why2-Atlas tutoring system presents students with qualitative physics questions and encourages them to explain their answers via natural language. Although there are inexpensive techniques for analyzing explanations, we claim that better understanding is necessary to provide substantive feedback. In this paper we motivate and describe how the system creates and utilizes a proof-based representation of student essays and provide some preliminary evaluation results.

1 Introduction

The Why2-Atlas system presents students with qualitative physics problems and encourages them to write their answers along with detailed explanations to support their answers [17]. The student explanation shown in Figure 1, which is from our corpus of human-human computer-mediated tutoring sessions, illustrates the type of explanation the system strives to elicit from students. It is a form of self-explanation so it has the potential to lead students to construct knowledge [4], and to expose deep misconceptions [15]. But it is difficult to encourage these explanations without giving the student substantive feedback [1]. We claim that before a system can give substantive feedback it has to be able to understand student explanations to some degree.

Statistical text classification approaches, such as latent semantic analysis [9] and naive bayes [10], have shown promise for assessing student explanations [13] and are appealing because they can be trained directly on a domain specific natural language corpus and on short texts that represent prescriptively good and bad descriptions against which a student text can be compared. For instance, a bad description that should match Figure 1 is the often-observed impetus misconception: "If there is no force on a moving object, it slows down".

These approaches statistically derive a semantic representation for a text relative to the training data but do so by treating language as an unordered bag of words in which the organization of the words imparts no meaning. Because these techniques fail to capture this additional semantics, they are insensitive to a number of language phenomenon that help distinguish between good and bad explanations. Two problems of interest for this paper are weak inferencing and lack of precision. In Figure 1, the student has the extreme belief that the pumpkin has *no* horizontal velocity. This inference probably would not be recognized as a case of "slowing down" by bag of words approaches. Students also make true but vague

*This work was funded by NSF grant 9720359 and ONR grant N00014-00-1-0600.

Question: Suppose you are running in a straight line at constant speed. You throw a pumpkin straight up. Where will it land? Explain.

Explanation: Once the pumpkin leaves my hand, the horizontal force that I am exerting on it no longer exists, only a vertical force (caused by my throwing it). As it reaches it's maximum height, gravity (exerted vertically downward) will cause the pumpkin to fall. Since no horizontal force acted on the pumpkin from the time it left my hand, it will fall at the same place where it left my hands.

Figure 1: The statement of the problem and an example explanation.

statements. For example, a student may make a true statement about the velocity of an object but not report it in terms of its horizontal and vertical components. Since just a few content words are missing the student's statement would look close to its more precise counterpart. To address these problems, we need a deeper understanding of the student's explanation.

The PACT Geometry Tutor does a deeper semantic classification [1] of student utterances by parsing a student explanation into a propositional representation and then classifying it relative to prescriptive categories that are expressed as terminological knowledge. This approach is promising for cases where the explanation is typically a single sentence, as is the case for PACT, but with the the longer, more complex explanations that the Why2-Atlas system strives to elicit, the discourse-level meaning of the text would be largely overlooked.

Why2-Atlas also parses student utterances into propositional representations. It uses a syntactic grammar and lexical semantics to create a representation for a each sentence [14] and then resolves temporal and nominal anaphora [7]. But instead of classifying each of the resulting propositions, it constructs abductive proofs that combine the propositions. A proof-based approach gives more insight into the line of reasoning the student may be following across multiple sentences because proofs of many propositions should share subproofs. Indeed, one proposition's entire proof may be a subproof of the next proposition. Moreover, subtle misconceptions such as impetus are revealed when they must be used to prove a student's explanation. The proof-based approach also opens the possibility of implementing interactive proof generation via a dialogue with the student. This interaction can serve the dual purpose of revealing to the student the conjectured argumentation behind her statement, and disambiguating the student's intended meaning when there are multiple proofs.

First we explain the pedagogical considerations that motivate our selection of a proof-based representation of the student's essay. We then motivate our choice of weighted abduction and explain our implementation of the Tacitus-lite+ abductive prover and illustrate with an example how it builds a proof. Finally, we discuss our preliminary evaluation results.

2 Deriving Student Feedback

The proofs that Why2-Atlas produces represent the student's knowledge and beliefs about physics with respect to the problem to which he is responding. Acquiring and reasoning about student beliefs and knowledge is one of the central issues addressed by work in student modeling. In the case of Why2-Atlas, the system needs this representation to identify communicative strategies and goals that will 1) effectively help the student realize and correct his errors and misconceptions and 2) enable the student to realize what reasoning is necessary when generating a complete explanation.

One difficulty any system must address is uncertainty about the beliefs and knowledge it should attribute to a student. This uncertainty arises because some of the knowledge and

beliefs about the student are inferred based on observed student actions or utterances. So as with decision theoretic approaches [11], the system needs to reason about the utility of separately attributing mutually exclusive representations of varying plausibility to the student. Why2-Atlas tries to estimate this by associating costs with the proofs it creates. However there can still be multiple proofs that are considered equally good representations.

Another consideration is that self discovery of errors may be more effective than always being immediately told of the error and its correction. Currently in Why2-Atlas, if the proof reveals a misconception or error then the system will engage the student in a dialogue that works through an analogous, but simplified problem and summarizes at the end with a generalization of the reasoning that the student is expected to transfer to the current problem. If incompleteness is revealed by the proof then the system will engage the student in a dialogue that leads the student to express the missing detail. It does so by reminding the student of an appropriate rule of physics, a fact that is relevant to the premise or conclusion of the rule and then asking the results of applying the rule.

Working through an analogous problem is the only technique for leading a student to recognize his error or misconception currently implemented in the system. Another possibility is to step through the student's reasoning as represented by the proof and ask the student to supply inferred details. Having some of these details wrong may have led the student to draw a wrong conclusion and making them explicit may enable her to more easily see the source of her error. Other techniques for dialogue strategies to correct misconceptions, errors and incompleteness relative to proofs may be derivable from argumentation strategies used in argument generation as described in [18] (e.g. reductio ad absurdum and premise to goal).

3 Abductive explanation

An *abductive logic programming framework* [8] in the context of our system is defined as a triple $\langle P, A, IC \rangle$, where P is the set of *givens* and *rules*, A is the set of abducible atoms (potential hypotheses) and IC is a set of integrity constraints. Then an *abductive explanation* of a given set of sentences G (observations), is $\Delta \subseteq A$ such that $P \cup \Delta \models G$ and $P \cup \Delta$ satisfies IC and a respective *proof* of G. An abductive explanation is generally not unique.

In building a model of the student's reasoning, our goal is to simultaneously increase a function of measures of utility and plausibility. The utility measure is an estimate of the utility of the choice of a particular proof for the tutoring application given a plausibility distribution on a set of alternative proofs. The plausibility measure indicates which explanation is the most likely. It gives preference to shallow proofs and reflects an assumption we are making about cognitive economy: if a short proof and a long proof both explain the student's utterance, and all rules and assumptions in both proofs are equally likely, then the short proof is the more likely interpretation. Of course, comparison of the depths of proofs is complicated by the fact that the rules in a theorem prover are not all of equal importance in the context of a solution. Thus some steps of the formal proof can be safely omitted in an actual solution provided by an expert. In the context of using the proof as a student model, this preference makes the model optimistic about the student's skills. In the context of using the proof for guiding tutorial feedback a shallow proof has greater utility since according to our assumption it is the type of the proof the tutor would prefer to discuss. Another factor that contributes to the utility is the preference for explanations that use good Physics vs. "buggy" Physics.

Since an explicit estimation of utility requires the generation of multiple proofs and is

therefore computationally expensive, we deploy a number of proof search heuristics to approximate the combination of the two measures. Although the parameters of these heuristics currently are fixed for the duration of a tutoring session, our implementation allows for varying the parameters on-the-fly. We expect this to be useful for dynamic adjustment of the student model in cases where the model should be more pessimistic about the student's skills.

While the depth preference is neutral to the content of the explanation and the correctness preference gives only binary output for each rule, the approaches taken in cost-based [3] and weighted abduction take into account the relative plausibility of individual hypotheses[1]. The proofs can then be ordered by the total cost of their abductive hypotheses.

We have chosen weighted abduction over the cost-based approach since the cost of a hypothesis in the former approach is sensitive to (a) the relative plausibility of the goals (observations) to be explained, (b) the explanatory chain that generated the hypothesis, and (c) the relative plausibility of the antecedents of rules. The drawback of weighted abduction in comparison to cost-based abduction, however, is the lack of a precisely defined semantics of weights. We do not attempt to provide a formal definition of its semantics in this paper, instead we use ad hoc heuristics that are applicable to our particular application.

Following the weighted abductive inference algorithm described in [16], our abductive prover, Tacitus-lite+, is a collection of rules where each rule is expressed as a Horn clause $p_1^{w_1} \wedge \cdots \wedge p_n^{w_n} \rightarrow r$, where each conjunct p_i has a weight w_i associated with it. The weight is used to calculate the cost of assuming p_i instead of proving it where $cost(p_i) = cost(r) * w_i$. The costs of observations are supplied as input to the prover.

Given a goal or observation to be proven, Tacitus-lite+ takes one of three actions; 1) assumes the observation at the cost associated with it 2) factors it with an atom (i.e. unifies variables and merges into one atom) that is either a fact or has already been proven or assumed (in the latter case the cost of the resultant atom is counted once in the total cost of the proof, as the minimum of the two costs) 3) attempts to prove it with a rule.

The applications builder can set cost thresholds and bounds on the depth of rules applied in proving an observation and on the global number of proofs generated during search. Tacitus-lite+ maintains a queue of proofs where the initial proof reflects assuming all the observations and each of the three above actions adds a new proof to the queue. The proof generation can be stopped at any point and the proofs with the lowest cost can be selected as the most plausible proofs for the observations.

Tacitus-lite+ uses a best-first search guided by heuristics that select which proof to expand, which observation or goal in that proof to act upon, which action to apply and which rule to use when that is the selected action. The cost threshold allows us to avoid iterative deepening and implement heuristics to help find a low-cost proof before we exhaust depth or number of proofs thresholds. Thus our current search strives to satisfy a criterion of no cheaper proof of the same depth or shorter, and one of the thresholds (depth, number of proofs, proof cost) being met. As we mentioned previously, most of the heuristics in Why2-Atlas are specific to the domain and application[2].

Tacitus-lite+ also includes a fixed set of integrity constraints. A knowledge base $P \cup \Delta$ satisfies an integrity constraint $\phi \in IC$ iff $P \cup \Delta \cup \phi$ is consistent[3]. The first constraint is $\neg[p \wedge p^*]$, where p^* means the opposite of p, following the approach described in [8, 6].

[1] Belief revision in Bayesian networks can be accurately modeled by cost-based abduction [12].

[2] Polynomial algorithms exist for some useful classes of abductive problems [5]. Since weighted abduction is not yet one of them, we are still exploring the best heuristics to use for our domain and application.

[3] This is known as the *consistency view* of integrity constraints (see for example [8]).

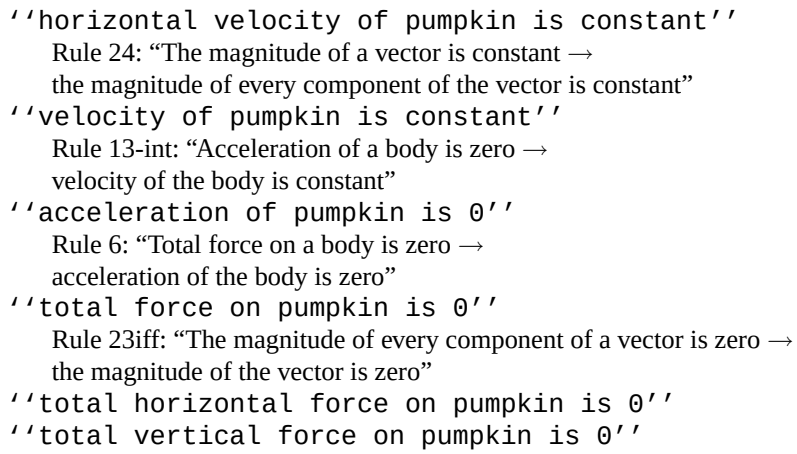

Figure 2: Example of an inconsistent proof. One of the newly generated goals "total horizontal force on pumpkin is 0" is inconsistent with the previously proven fact "total vertical force on pumpkin is a nonzero constant".

For example, if atom p stands for "velocity of pumpkin is constant", then p^* is "velocity of pumpkin is non-constant". The abductive explanation Δ satisfies this constraint iff for every atom p, $P \cup \Delta \nvDash p \wedge p^*$ (**C1**).

For the sake of computational efficiency we did not implement the completeness part of the semantics of negation as failure which requires that one of the following must hold: $P \cup \Delta \vDash p$ or $P \cup \Delta \vDash p^*$. Nor did we fully implement constraint **C1**. In this case each step of a proof must be checked by testing whether the negation of the atom is provable with no new steps or with steps that cost less than the proof of the original atom. As suggested in [2], in the case of weighted abduction one should settle for incomplete consistency checking and focus on detecting the inconsistencies that are most likely to arise in the application domain.

Instead of implementing constraint **C1** above, we prevent the application of abductive rules that would immediately give rise to a new goal p when the proof generated so far has atom p^* (corresponding to the same physical quantity, same bodies and same times) such that p does not unify with p^*. Our unification algorithm takes proper account of a sort hierarchy defined as part of our Qualitative Physics ontology, so that the atoms corresponding to the pair of statements, "velocity of pumpkin is increasing" and "velocity of pumpkin is non-constant", are unifiable while the atoms in "velocity of pumpkin is increasing" and "velocity of pumpkin is constant" are not.

As an example of the above, consider a fragment of a proof tree starting from the subgoal "horizontal velocity of pumpkin is constant" as shown in Figure 2. First, assume that the fact "total vertical force on pumpkin is a nonzero constant", which refers to the time the pumpkin is in free-fall, has been proven in another branch of the proof tree. In this case, the application of rule "23iff" would not be allowed in this proof since it results in the need to prove the contradictory statement "total vertical force on pumpkin is 0".

Another kind of inconsistency is related to meta-knowledge reasoning, in which rules have buggy counterparts. A distinctive feature of the task of modeling the student's reasoning is that it is necessary to account for erroneous facts and rules. Some false facts correspond to a wrong idealization and the rest are typically conclusions that students make via the application of false domain rules. Both are modeled by buggy domain rules and buggy meta-knowledge rules.

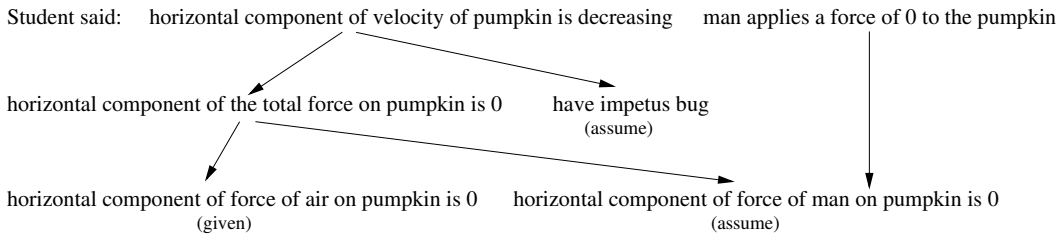

Figure 3: Example of Simplified Abductive Proof for "The pumpkin's horizontal motion slows down because the man is not exerting a force on it."

For example if a correct rule (in the sense of a rule schema, e. g. p, q and q^* have unbound variables), $p \rightarrow q$ has a buggy counterpart, $p \rightarrow q^*$, then both of them cannot be part of a logic program P that includes fact $\exists X p(X)$, provided we want to keep P consistent. This constitutes constraint **C2** which we implemented at the meta-level by preventing the simultaneous appearance of both members of any paired rules in the same proof.

A canonical problem idealization is formalized as *givens*, or facts, of the abductive prover. Facts that may be misunderstood by the student because of a wrong idealization are represented as *pairs of correct and buggy givens*. In the context of a student's reasoning about the problem, buggy givens are wrong assumptions the student made during idealization. For example with the pumpkin problem, the two facts, "$\rightarrow$ the force of air resistance on the pumpkin is zero", and, "$\rightarrow$ the force of air resistance on the pumpkin is nonzero", are a pair of correct and buggy givens respectively.

Note that while the consistency constraints we describe are natural in theorem proving, from the point of view of student modeling they represent a risky assumption; that the student does not simultaneously hold inconsistent beliefs.

4 Building and Utilizing Abductive Proofs

The system currently has 105 qualitative physics rules available to use in building proofs. These rules cover 5 problems as well as parts of many other problems.

Figure 3 is an example of a simplified abductive proof for "The pumpkin's horizontal motion slows down because the man is not exerting a force on it." Each level of downward arrows from the gloss of a proposition in Figure 3 represents a domain rule that can be used to prove that proposition. One way to prove that the horizontal component of the pumpkin's velocity is decreasing is to apply a buggy physics rule that is one manifestation of the impetus misconception; the student thinks that a force is necessary to maintain a constant velocity. In this example, Tacitus-lite+ assumes the student has this misconception but alternatively the system could try to gather more evidence that this is true by asking the student diagnostic questions.

Next Tacitus-lite+ proves that the total force on the pumpkin is 0 by proving that the possible addend forces are 0. In the context of this problem, it is a given that air resistance is negligible and so it factors with a fact for 0 cost. Next it assumes that the student believes the man is applying a horizontal force of 0 to the pumpkin.

Finally, it still needs to prove another proposition that was explicitly asserted by the student; that the force of the man on the pumpkin is 0. As with the velocity, it will try to prove this by proving that the horizontal component of that force is 0. Since it has already assumed that this is true, the abductive proof is complete.

5 Preliminary Evaluation Results

For a set of 5 correct, ideal essays, the average processing time per sentence was 21.22 seconds with a search bound of 50 proofs, a depth bound of 3 and a cost threshold of .05. However with 47 essay submissions for 2 students working on 5 problems, the students waited an average of just 8.4 seconds for a response from the system[4]. While we expect that a response delay of 8-22 seconds during a dialogue could be cognitively detrimental, it is not an unreasonable delay for understanding an essay and devising an initial response.

We also assessed the recognition of incorrect sentences relative to the processing bounds. Our test suite so far consists of 11 sentences that were extracted from actual student essays for 3 problems; 6 are incorrect, 4 are correct and 1 is ambiguous[5]. We then used Tacitus-lite+ to construct proofs for the sentences and reviewed each proof to see how many bugs were found. If a proof has a relatively high assumption cost that can mean either an insufficient knowledge base or representation, or processing bounds that are too restrictive.

Of the 6 incorrect statements, 2 were correctly found to have bugs. In one case the best available bug was found at a low cost but in the other the cost was relatively high and there was a more suitable bug available. In one of the remaining 4 incorrect sentences, the prover was not able to prove anything because the rule base was incomplete with respect to the sentence. For the remaining 3 incorrect sentences the prover was able to make assumptions at low costs without using any buggy rules. In one case the sentence could not be adequately represented and in the others the prover was able to make inexpensive assumptions. This was because there are weak limits on introducing new bodies with arbitrary properties. This problem is related to the lack of negation in our implementation and to the fact that the cost of a hypothesis is smaller when the hypothesis is assumed at a deeper level of the proof (according to the cost propagation formula).

In the case of the 1 sentence that was ambiguous, a plausible bug was found at a low cost and for the 4 correct statements, no bugs were found. But since expensive assumptions were made in 3 of the 4 this indicates that our rule base P is more complete for entailing incorrect statements than for proving correct statements. Overall, the prover performed its task in a promising manner since if found 3 of 6 incorrect sentences that it should have been able to find with no false positives. This preliminary evaluation has shown us that the problems so far are primarily with incompleteness of the knowledge base and the representation and not with the reasoner.

6 Conclusions

In this paper we argued for a need to achieve a deeper understanding of students' explanations than can be afforded by superficial sentence-level semantics. We presented abductive proofs, that are based on students essays, as a way to model students' beliefs and knowledge and described how we adapted a weighted-abduction reasoning framework for the task of building proofs of student essays. We developed a combination of heuristics to assist in choosing the best proof and hence the best model of the student by having these heuristics approximate selection criteria that are based on measures of utility and plausibility of a candidate model.

[4]This measure includes all of the system processing time, not just that of Tacitus-lite+, and counts just those cases where there is something new in the essay for which the system can obtain sentence propositions.

[5]We start with single sentences because the correctness of an essay's proof depends on the correctness of its component proofs.

References

[1] Vincent Aleven, Octav Popescu, and Kenneth Koedinger. Pilot-testing a tutorial dialogue system that supports self-explanation. In *Proceedings of Intelligent Tutoring Systems Conference*, volume 2363 of *LNCS*, pages 344–354. Springer, 2002.

[2] Douglas Appelt and Martha Pollack. Weighted abduction for plan ascription. *User Modeling and User-Adapted Interaction*, 2(1 – 2):1 – 25, 1992.

[3] Eugene Charniak and Solomon E. Shimony. Probabilistic semantics for cost based abduction. In *Proceedings of AAAI-90*, pages 106–111, 1990.

[4] Michelene T. H. Chi, Nicholas de Leeuw, Mei-Hung Chiu, and Christian LaVancher. Eliciting self-explanations improves understanding. *Cognitive Science*, 18:439–477, 1994.

[5] Kave Eshghi. A tractable class of abduction problems. In *Proceedings 13th International Joint Conference on Artificial Intelligence*, pages 3–8, Chambery, France, 1993.

[6] Kave Eshghi and Robert A. Kowalski. Abduction compared with negation by failure. In *Proceedings of the 6th International Conference on Logic Programming (ICLP '89)*, pages 234–254, 1989.

[7] Pamela Jordan and Kurt VanLehn. Discourse processing for explanatory essays in tutorial applications. In *Proceedings of the 3rd SIGdial Workshop on Discourse and Dialogue*, July 2002.

[8] Antonis Kakas, Robert A. Kowalski, and Francesca Toni. The role of abduction in logic programming. In Dov M. Gabbay, C. J. Hogger, and J. A. Robinson, editors, *Handbook of logic in Artificial Intelligence and Logic Programming*, volume 5, pages 235–324. Oxford University Press, 1998.

[9] Thomas K. Landauer, Peter W. Foltz, and Darrell Laham. An introduction to latent semantic analysis. *Discourse Processes*, 25:259–284, 1998.

[10] Andrew McCallum and Kamal Nigam. A comparison of event models for naive bayes text classification. In *Proceeding of AAAI/ICML-98 Workshop on Learning for Text Categorization*. AAAI Press, 1998.

[11] R. Charles Murray and Kurt VanLehn. DT Tutor: A dynamic decision-theoretic approach for optimal selection of tutorial actions. In *Proceedings of Intelligent Tutoring Systems Conference*, volume 1839 of *LNCS*, pages 153–162. Springer, 2000.

[12] David Poole. Probabilistic Horn abduction and Bayesian networks. *Artificial Intelligence*, 64(1):81–129, 1993.

[13] Carolyn Rosé, Dumisizwe Bhembe, Antonio Roque, Stephanie Siler, Ramesh Srivastava, and Kurt VanLehn. A hybrid understanding approach for robust selection of tutoring goals. In *Proceedings of Intelligent Tutoring Systems Conference*, volume 2363 of *LNCS*, pages 552–561. Springer, 2002.

[14] Carolyn Rosé, Antonio Roque, Dumisizwe Bhembe, and Kurt VanLehn. An efficient incremental architecture for robust interpretation. In *Proceedings of Human Language Technology Conference*, San Diego, CA., 2002.

[15] James D. Slotta, Michelene T.H. Chi, and Elana Joram. Assessing students' misclassifications of physics concepts: An ontological basis for conceptual change. *Cognition and Instruction*, 13(3):373–400, 1995.

[16] Mark Stickel. A Prolog-like inference system for computing minimum-cost abductive explanations in natural-language interpretation. Technical Report 451, SRI International, 333 Ravenswood Ave., Menlo Park, California, 1988.

[17] Kurt VanLehn, Pamela Jordan, Carolyn Rosé, Dumisizwe Bhembe, Michael Böttner, Andy Gaydos, Maxim Makatchev, Umarani Pappuswamy, Michael Ringenberg, Antonio Roque, Stephanie Siler, and Ramesh Srivastava. The architecture of Why2-Atlas: A coach for qualitative physics essay writing. In *Proceedings of Intelligent Tutoring Systems Conference*, volume 2363 of *LNCS*, pages 158–167. Springer, 2002.

[18] Ingrid Zukerman, Richard McConachy, and Kevin B. Korb. Using argumentation strategies in automated argument generation. In *Proceedings of the 1st International Natural Language Generation Conference*, pages 55–62, 2000.