Mixed Language Processing in the Why2-Atlas Tutoring System

Maxim Makatchev [1], Brian S. Hall, Pamela W. Jordan, Umarani Pappuswamy and Kurt VanLehn

Learning Research and Development Center, University of Pittsburgh

Abstract. When dealing with a natural language interaction about a formal domain, a number of phenomena occur. They include interspersing natural language with formulas, various degrees of formality, and conveying the logical structure of an essay. Capturing these phenomena, to some extent, is necessary for providing relevant tutoring feedback. In this paper we discuss these phenomena, the extent to which we process each of them in the Why2-Atlas tutoring system and directions for future work.

Keywords. Mixed languages, Dialog-based intelligent tutoring systems

1. Introduction

The Why2-Atlas tutoring system [10] is designed to encourage student's self-explanation via a natural language (NL) dialog. After a problem from the domain of qualitative mechanics is presented to the student, she is asked to write an essay containing her answer and explanation (Figure 1). The essay is analyzed for its correctness and coverage [7] and based on the found errors or missing facts or justifications the system starts a dialog. After the points are discussed in the dialog the student is asked to update her essay and the cycle continues until the essay is considered satisfactory.

When is the essay good enough? There are two tutoring objectives that are aimed at by Why2-Atlas. First, the student should use appropriate language to describe domain concepts. Second, the student should develop an understanding of relationships between mechanical concepts represented by laws and formulas and apply these relationships to generate an admissible logical argument that leads to the correct answer. These requirements imply the following criteria for evaluating an essay:

- proper use of physics terms,
- correctness of statements,
- correctness of justifications, namely laws and relationships,
- a significant overlap of the logical structure of the essay with the ideal solution graph (a "proof").

[1] Correspondence to: Maxim Makatchev, LRDC, 3939 O'Hara Street, Pittsburgh, PA 15260, USA. Tel.: +1 412 624 7498; Fax: +1 412 624 7904; E-mail: maxim@pitt.edu.

Problem: A heavy clay ball and a light clay ball are released in a vacuum from the same height at the same time. Which reaches the ground first? Explain.
Explanation: Both balls will hit at the same time. The only force acting on them is gravity because nothing touches them. The net force, then, is equal to the gravitational force. They have the same acceleration, g, because gravitational force=mass*g and f=ma, despite having different masses and net forces. If they have the same acceleration and same initial velocity of 0, they have the same final velocity because acceleration=(final-initial velocity) elapsed time. If they have the same acceleration, final, and initial velocities, they have the same average velocity. They have the same displacement because average velocity=displacementtime. The balls will travel together until the reach the ground.

Figure 1. The statement of the problem and a verbatim student explanation.

All of these evaluation criteria, except for the last one, apply also to the analysis of some of the student turns during the dialog stage. Analyzing an essay or a dialog utterance with respect to these criteria requires dealing with the following language phenomena:

- symbolic expressions: e.g. "a is constant," "f=ma";
- numeric expressions: e.g. "acceleration is 9.8 m/s^2";
- formally and informally described physics terms: e.g. "speeding up" versus "speed is increasing," "move side by side" versus "have equal positions";
- generic and instantiated versions of the physics laws and formulas: "force on the ball equals its mass times g" versus "force is mass times acceleration."
- logical relations being expressed in natural language and omission of logical relations: "*provided* there is no air resistance. . . ," "*if it was true that* the balls fall at the same acceleration, . . . "

Similar language phenomena have been reported in tutoring systems for other formal domains, e. g. mathematical dialogs [1,11] and qualitative process theory [4].

In this paper we describe the extent to which each of these language phenomena are dealt with in the Why2-Atlas tutoring system. In Section 2 we discuss the representation and processing that we use for each of the language phenomena outlined above. Section 3 is dedicated to our the techniques for analyzing the student's essay and dialog turns with respect to the evaluation criteria we listed above. We summarize our current progress and outline possible directions for future work in Section 4.

2. Recognizing and representing mixed language expressions

2.1. Symbolic and numeric expressions

Some examples of the expressions that fall into this category are:

a. "acceleration is final velocity minus initial velocity over elapsed time"
b. "net force is mass times acceleration"
c. "9.8 m/s^2"
d. "a = 9.8 m/s^2"
e. "the equation <net force = m * a>"

Formulas can be expressed in natural language *(a), (b),* in algebraic form *(c), (d),* or in natural language mixed with algebraic symbols *(e)*. In all cases these expressions are treated as semantic units and are marked in the student input by an equation identifier. The equation identifier matches the student's text with the stored representations of commonly occurring correct and erroneous formulas.

One of the observed differences between explanations in the subdomain of qualitative physics and the domain of mathematics is that much of the reasoning consists of application of a small number (37) of physics principles (10 of which are vector relations between physical quantities, 12 are their derivatives, including qualitative relations, and the rest are rules of idealization, e.g. "Possible forces are either contact forces or the gravitational force"). This allows us to match directly against the representations relevant to the correct and most common buggy formulas corresponding to the physics principles, as opposed to compositional representations of formulas as presented in [1]. In all, twelve legitimate and seven buggy (including "unrecognizable") mathematical forms are identified. The identifier is implemented as a series of regular expressions applied to the student input after spelling correction, but before invocation of any of our language understanding modules. The resulting text, with formulas replaced by tags can be passed through a parser such as CARMEL [8] or MINIPAR/Rappel [5,3] since both parsers are robust enough to skip unknown words (CARMEL), or treat them as nouns (MINIPAR), without significant anticipated performance loss. The equation identifier is currently being tested as a part of the evaluation of the Why2-Atlas system.

Since fragments of algebraic expressions can be interspersed with text, as in *(e)*, some degree of flexibility is needed in defining an acceptable syntax for formulas. The use of angle brackets as an enclosure for algebraic forms is implicitly suggested by incorporating this convention into the tutoring materials presented to the student. These grouping characters, together with parentheses, are recognized but optional (and need not even be balanced). Since mathematical forms in prose would tend to omit them even if the result was mathematically ambiguous, as in *(a)*, the equation identifier must tolerate this type of ill-formedness for all modes of expression. Currently lower and upper case distinctions are also ignored, even though it means losing the distinction between "G" (universal gravitational constant) and "g" (acceleration due to gravity).

The equation identifier is not entirely forgiving of ill-formedness, however; it detects a small set of common buggy formulas: for example, "a = m / f" instead of the correct "a = f / m." The idea is to identify common algebraic errors, and errors of omission such as "velocity" instead of "average velocity." The final step is a catch-all pattern to identify strings like "x = y * z" in which one or more of the variables cannot be recognized, as a generic buggy equation or perhaps more accurately, "unrecognizable mathematical form."

Example *(c)* is identified as a synonym for "g" gravitational acceleration, by virtue of both its numeric value and its units – the expression "9.8" on its own would not be given such privileged status. Such a numeric expression can, of course, be embedded in an equation proper, as in *(d)*.

Once the correct or buggy formula is identified, it is represented as an atom in the first-order predicate logic-based knowledge representation, for example
`(math-form mf1 a-equals-f-over-m)` and
`(math-form mf2 buggy-a-equals-f-over-m-inverted),`
where `mf1` and `mf2` are the atom identifiers used for cross-referencing.

Category	Example of natural language expression
relative position	"keys are behind (in front of, above, under, close, far from, etc.) man"
motion	"move slower," "slow down," "moves along a straight line"
dependency	"horizontal speed will not depend on the force"
direction	"the force is downward"
interaction	"the man pushes the pumpkin," "the gravity pulls the ball"

Table 1. Categories of informal physics expressions.

2.2. Formal and informal physics

As with many other domains, some terms from the domain of mechanics have a much less formal use in everyday language, for example "this will force the objects to move at constant speed." This problem can normally be handled by filtering of syntactic categories. Conversely, many mechanics phenomena may be described in an informal language, for example "speed up" instead of "accelerate," "push" instead of "apply a force," "the force is downward" instead of "the force is negative." This type of informality has proven to be a significant problem.

The difficulty is partially in defining the boundaries of equivalence classes for informal expressions about physics concepts. Consider the example "The object will slow down." The correctness of this statement with respect to a particular context can be evaluated, despite its somewhat informal language, by representing it as a velocity with a decreasing magnitude, a decreasing speed, or a negative acceleration. When evaluating the coverage of this statement, however, it may not be desirable to represent this statement in terms of formal physics, since this may attribute to the student more knowledge than she has actually expressed.

We address this problem by adopting representations for different levels of formality. The formal physics concepts are represented by predicates for vector quantities (position, displacement, velocity, acceleration, force, total force, momentum), scalar quantities (mass, speed, distance, duration), states (for example contact state, being in vacuum, freefall), relations (comparison of magnitudes and directions) and a predicate defining the order of the time instants [6]. Informal expressions can be grouped into categories, some of which are shown in Table 1.

In the current version of our system we use a dedicated predicate for representing each of these informal categories, except for the interaction category which we are considering implementing at a later stage. Below we include the somewhat abridged representation of the sentences "there is a downward force of gravity"

```
 (force f1 ?body1 ?body2 ?comp1 ?d-mag1 ?d-mag-num1
      ?mag-zero1 ?mag-num1 ?dir1 ?dir-num1 ?d-dir1 ?time1 ?time2)
(due-to dt1 f1 gravity)
(coordinate-system cs1 ?dir1 down)
```

and "the keys are behind the man"

```
 (rel-position rp1 behind keys man ?time3 ?time4)
```

In the former example arguments that are equal across predicates are represented via shared variables.

2.3. Generic and instantiated physical laws

The qualitative rules of physics can be described in a generic form, e.g. “for two objects, if the acceleration, initial velocity and duration are the same, so is final velocity,” or in an instantiated form “the balls have the same initial velocity and acceleration so their velocity will be the same at all times.” Formulas (both in algebraic and NL form) can also be generic “a=(vf-vi)/t” or specific (for example, “a1=vf1-vi1/t1,” “acceleration of the heavy ball equals to final velocity minus initial velocity over t.”)

Generic formulas are recognized and represented using the framework described in Section 2.1. While we have not seen many specific algebraic expressions in our corpus, there are cases when formulas expressed in natural language refer to specific bodies. This imbalance can be explained by two facts: first, there are no examples of specific algebraic expressions used by the tutor; second, specific algebraic expressions require introducing new notations, while specific formulas expressed in NL do not. Due to the small number of observed examples in our corpus, currently we do not attempt to represent or recognize instantiated formulas in either algebraic or NL form.

However we do represent to a certain degree both generic and specific versions of the qualitative relations that are consequences of the formulas. Consider the examples of a generic and its respective specific relation from the beginning of the section: “For two objects, if the acceleration, initial velocity and duration are the same, so is final velocity,” and ‘the balls have the same initial velocity and acceleration so their velocity will be the same at all times.” The difference between these two expressions is in the specificity of the subject of the sentence: a ball is a subclass of an object. This nuance can be represented by appropriate type restrictions on the corresponding variables and constants, and by using a universal quantifier. Our current knowledge representation, while allowing for typing of variables, does not allow quantifiers or Skolem constants, for the sake of efficiency of reasoning. Instead, all the variables in standalone atoms are assumed to be existentially quantified and all variables in the rules (except for variables present only in the consequent of a rule) are assumed to be universally quantified [6]. Therefore currently we settle for an imprecise representation of the generic propositions, via an existentially quantified variable of a generic type. While this would lead to ambiguity, in that an existentially quantified version of the statement would have a representation that is identical with that of the universally quantified version, lack of the existentially quantified expressions in our corpus (of the sort “there are two objects that have the same velocity”) alleviates the potential problem. Incorporating a syntax for both quantifiers is planned for a future version of the knowledge representation.

2.4. Recovering the logical structure

Ideally, we would like to treat an essay not as a set of unordered propositions about the domain, but to capture and represent the causal and dependency relations between these propositions. Representing causal relations is complicated by the fact that there are potentially multiple causes for a particular consequence. Aside from the difficulty of recognizing the causes from the text, there is the problem of representing the cause-effect relation in a form that (1) can be flexible enough to account for a variable number of causes, (2) can be efficiently processed. One solution that satisfies both of these criteria would require a predicate of variable arity:

`(cause c1 cause1 cause2 cause3 ...causeN effect1)`.
This representation is considered superior to one with N binary predicates of the form
`(cause c1 causei effect1)`
that increases the number of cross-referenced atoms to be matched against stored representations. Matching cross-referenced atoms is an expensive procedure: the time complexity of the algorithm is $O(2^n n^3)$, where n is the number of input atoms [9].

Another nuance of cause-effect representation is *asserting* versus *non-asserting* conditions. Consider the following examples, "if there was air resistance, the larger ball would fall faster," and "since there is no air resistance, the balls fall at the same speed." Clearly there is a difference in the speaker's belief about whether the condition actually holds or not. The logical structures corresponding to these sentences can be represented as $A \rightarrow B$ and $A \wedge (A \rightarrow B)$ respectively.

Currently we do not recognize this type of logical structure in an essay, and instead match the unordered set of cross-referenced atoms with the stored representations for facts and physics rules and analyze the intersection of the matched facts and rules with the statements in the nodes of an ideal "proof" graph (more details are in Section 3.1).

One aspect of essay structure that we do represent is dependencies between physical quantities. This is relevant for representing such physics statements as "the freefall acceleration does not depend on mass," "the horizontal velocity does not depend on the vertical force," etc., and was discussed in Section 2.2. We are considering implementing a more sophisticated mechanism for reasoning about the logical structure of an essay in a future version of the system.

3. Evaluating essays and dialog turns

3.1. Coverage

Each of the four physics problems implemented in the system has an ideal "proof" designed by expert physics tutors that contains steps of reasoning, i.e. facts and their justifications, and ends with the correct answer. A fragment of the proof for the Clay Balls problem stated in Figure 1 is given in Figure 2. This proof is represented in the system by a graph such that an edge (a, b) is in the graph if a is a justification rule or a fact that is used as a premise (antecedent) of a rule that derives b. The justification in the ideal proof corresponds to the generic physics rules, in their algebraic form and/or qualitative form. The representation of a student essay as a set of cross-referenced first-order predicate logic atoms is matched against the nodes of the ideal graph, some of which are marked as required. Based on the overlap with the required nodes and taking into account the structure of the graph, the system initiates a dialog to elicit the remaining required points [10].

In addition to the manually generated ideal proof graph, each problem has a corresponding automatically generated assumption-based truth maintenance system (ATMS) that includes a large set of possible correct and buggy facts that can be derived from the problem givens using correct and buggy rules. This allows us to compute the inferential proximity metric between the student utterance and a required fact [7]. If the utterance is within a single inference step from a required fact the student may be given partial credit and appropriate feedback to help bridge the gap between the two statements. Providing this kind of a feedback is being considered for a future version of our system [2].

Step	Proposition	Justification
1	Both balls are near earth	Unless the problem says otherwise, assume objects are near earth
2	Both balls have a gravitational force on them due to the earth	If an object is near earth, it has a gravitational force on it due to the earth
3	There is no force due to air friction on the balls	When an object is in a vacuum, no air touches it
4	The only force on the balls is the force of gravity	Forces are either contact forces or the gravitational force
5	The net force on each ball equals the force of gravity on it	[net force = sum of forces], so if each object has only one force on it, then the object's net force equals the force on it
6	**Gravitational force is w = m*g for each ball**	**The force of gravity on an object has a magnitude of its mass times g, where g is the gravitational acceleration**
⋮	⋮	⋮
18	**The balls have the same initial vertical position**	given
19	The balls have the same vertical position at all times	[Displacement = difference in position], so if the initial positions of two objects are the same and their displacements are the same, then so is their final position
20	**The balls reach the ground at the same time**	

Figure 2. A fragment of an ideal "proof" for the Clay Balls problem from Figure 1. The required points are in bold.

3.2. Correctness

Domain statements that are recognized from the student input, namely facts (including instantiated rules) and generic rules, are also analyzed for correctness. This is done via a two step process: first, the statements are matched against known common buggy statements; second, they are matched against nodes of the ATMS. The latter match allows us to determine that a statement is buggy even when there is no corresponding bug in the list of common bugs, because the only ATMS environments in which it holds true include buggy assumptions. More details on this, including preliminary evaluation results, can be found in [7].

4. Conclusion and future work

Natural language interaction with a student about a formal domain produces a number of interesting natural language phenomena that need to be processed to generate adequate tutoring feedback. In our system we demonstrated the feasibility of capturing those phenomena that require limited knowledge about the domain and are undamaged by our treatment of an essay as a set of individual sentences. This already provides the system with useful information for generating richer tutoring feedback. Among the challenging phenomena that we hope to tackle in future versions of the system are the logical structure of an essay and more complete treatment of informal statements about the domain.

Acknowledgments

This research has been supported under NSF grant 0325054 and ONR grant N00014-00-1-0600. The authors would like to thank present and past members of the Natural Language Tutoring group for their work on the Why2-Atlas tutoring system.

References

[1] Helmut Horacek and Magdalena Wolska. Interpreting semi-formal utterances in dialogs about mathematical proofs. In F. Meziane and E. Métais, editors, *Natural Language Processing and Information Systems*, volume 3136 of *LNCS*, pages 26–38. Springer, 2004.

[2] Pamela W. Jordan. Using student explanations as models for adapting tutorial dialogues. In *Proceedings of 17th International FLAIRS Conference*, 2004.

[3] Pamela W. Jordan, Maxim Makatchev, and Kurt VanLehn. Combining competing language understanding approaches in an intelligent tutoring system. In *Proceedings of Intelligent Tutoring Systems Conference*, volume 3220 of *LNCS*, pages 346–357, Maceió, Alagoas, Brazil, 2004. Springer.

[4] Sven E. Kuehne and Kenneth D. Forbus. Capturing QP-relevant information from natural language text. In Johan de Kleer and Kenneth D. Forbus, editors, *Proceedings of the 18th International Workshop on Qualitative Reasoning*, pages 25–32, Evanston, USA, 2004. Lawrence Erlbaum Associates.

[5] Dekang Lin. Dependency-based evaluation of MINIPAR. In *Proc. of Workshop on the Evaluation of Parsing Systems*, Granada, Spain, May 1998.

[6] Maxim Makatchev, Pamela W. Jordan, and Kurt VanLehn. Abductive theorem proving for analyzing student explanations to guide feedback in intelligent tutoring systems. *Journal of Automated Reasoning, Special issue on Automated Reasoning and Theorem Proving in Education*, 32:187–226, 2004.

[7] Maxim Makatchev and Kurt VanLehn. Analyzing completeness and correctness of utterances using an ATMS. In *Proceedings of Int. Conference on Artificial Intelligence in Education, AIED2005*. IOS Press, July 2005.

[8] Carolyn P. Rosé. A framework for robust semantic interpretation. In *Proceedings of the First Meeting of the North American Chapter of the Association for Computational Linguistics*, pages 311–318, 2000.

[9] Kim Shearer, Horst Bunke, and Svetha Venkatesh. Video indexing and similarity retrieval by largest common subgraph detection using decision trees. *Pattern Recognition*, 34(5):1075–1091, 2001.

[10] 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.

[11] Magdalena Wolska and Ivana Kruijff-Korbayová. Analysis of mixed natural and symbolic language input in mathematical dialogs. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics*, pages 25–32, Barcelona, Spain, 2004.