The impact of interpretation problems on tutorial dialogue

Myroslava O. Dzikovska and **Johanna D. Moore**
School of Informatics, University of Edinburgh, Edinburgh, United Kingdom
{m.dzikovska,j.moore}@ed.ac.uk

Natalie Steinhauser and **Gwendolyn Campbell**
Naval Air Warfare Center Training Systems Division, Orlando, FL, USA
{natalie.steihauser,gwendolyn.campbell}@navy.mil

Abstract

Supporting natural language input may improve learning in intelligent tutoring systems. However, interpretation errors are unavoidable and require an effective recovery policy. We describe an evaluation of an error recovery policy in the BEETLE II tutorial dialogue system and discuss how different types of interpretation problems affect learning gain and user satisfaction. In particular, the problems arising from student use of non-standard terminology appear to have negative consequences. We argue that existing strategies for dealing with terminology problems are insufficient and that improving such strategies is important in future ITS research.

1 Introduction

There is a mounting body of evidence that student self-explanation and contentful talk in human-human tutorial dialogue are correlated with increased learning gain (Chi et al., 1994; Purandare and Litman, 2008; Litman et al., 2009). Thus, computer tutors that understand student explanations have the potential to improve student learning (Graesser et al., 1999; Jordan et al., 2006; Aleven et al., 2001; Dzikovska et al., 2008). However, understanding and correctly assessing the student's contributions is a difficult problem due to the wide range of variation observed in student input, and especially due to students' sometimes vague and incorrect use of domain terminology.

Many tutorial dialogue systems limit the range of student input by asking short-answer questions. This provides a measure of robustness, and previous evaluations of ASR in spoken tutorial dialogue systems indicate that neither word error rate nor concept error rate in such systems affect learning gain (Litman and Forbes-Riley, 2005; Pon-Barry et al., 2004). However, limiting the range of possible input limits the contentful talk that the students are expected to produce, and therefore may limit the overall effectiveness of the system.

Most of the existing tutoring systems that accept unrestricted language input use classifiers based on statistical text similarity measures to match student answers to open-ended questions with pre-authored anticipated answers (Graesser et al., 1999; Jordan et al., 2004; McCarthy et al., 2008). While such systems are robust to unexpected terminology, they provide only a very coarse-grained assessment of student answers. Recent research aims to develop methods that produce detailed analyses of student input, including correct, incorrect and missing parts (Nielsen et al., 2008; Dzikovska et al., 2008), because the more detailed assessments can help tailor tutoring to the needs of individual students.

While the detailed assessments of answers to open-ended questions are intended to improve potential learning, they also increase the probability of misunderstandings, which negatively impact tutoring and therefore negatively impact student learning (Jordan et al., 2009). Thus, appropriate error recovery strategies are crucially important for tutorial dialogue applications. We describe an evaluation of an implemented tutorial dialogue system which aims to accept unrestricted student input and limit misunderstandings by rejecting low confidence interpretations and employing a range of error recovery strategies depending on the cause of interpretation failure.

By comparing two different system policies, we demonstrate that with less restricted language input the rate of non-understanding errors impacts both learning gain and user satisfaction, and that problems arising from incorrect use of terminology have a particularly negative impact. A more detailed analysis of the results indicates that, even though we based our policy on an approach ef-

Proceedings of the ACL 2010 Conference Short Papers, pages 43–48,
Uppsala, Sweden, 11-16 July 2010.

fective in task-oriented dialogue (Hockey et al., 2003), many of our strategies were not successful in improving learning gain. At the same time, students appear to be aware that the system does not fully understand them even if it accepts their input without indicating that it is having interpretation problems, and this is reflected in decreased user satisfaction. We argue that this indicates that we need better strategies for dealing with terminology problems, and that accepting non-standard terminology without explicitly addressing the difference in acceptable phrasing may not be sufficient for effective tutoring.

In Section 2 we describe our tutoring system, and the two tutoring policies implemented for the experiment. In Section 3 we present experimental results and an analysis of correlations between different types of interpretation problems, learning gain and user satisfaction. Finally, in Section 4 we discuss the implications of our results for error recovery policies in tutorial dialogue systems.

2 Tutorial Dialogue System and Error Recovery Policies

This work is based on evaluation of BEETLE II (Dzikovska et al., 2010), a tutorial dialogue system which provides tutoring in basic electricity and electronics. Students read pre-authored materials, experiment with a circuit simulator, and then are asked to explain their observations. BEETLE II uses a deep parser together with a domain-specific diagnoser to process student input, and a deep generator to produce tutorial feedback automatically depending on the current tutorial policy. It also implements an error recovery policy to deal with interpretation problems.

Students currently communicate with the system via a typed chat interface. While typing removes the uncertainty and errors involved in speech recognition, expected student answers are considerably more complex and varied than in a typical spoken dialogue system. Therefore, a significant number of interpretation errors arise, primarily during the semantic interpretation process. These errors can lead to *non-understandings*, when the system cannot produce a syntactic parse (or a reasonable fragmentary parse), or when it does not know how to interpret an out-of-domain word; and *misunderstandings*, where a system arrives at an incorrect interpretation, due to either an incorrect attachment in the parse, an incorrect word sense assigned to an ambiguous word, or an incorrectly resolved referential expression.

Our approach to selecting an error recovery policy is to prefer non-understandings to misunderstandings. There is a known trade-off in spoken dialogue systems between allowing misunderstandings, i.e., cases in which a system accepts and acts on an incorrect interpretation of an utterance, and non-understandings, i.e., cases in which a system rejects an utterance as uninterpretable (Bohus and Rudnicky, 2005). Since misunderstandings on the part of a computer tutor are known to negatively impact student learning, and since in human-human tutorial dialogue the majority of student responses using unexpected terminology are classified as incorrect (Jordan et al., 2009), it would be a reasonable approach for a tutorial dialogue system to deal with potential interpretation problems by treating low-confidence interpretations as non-understandings and focusing on an effective non-understanding recovery policy.[1]

We implemented two different policies for comparison. Our baseline policy does not attempt any remediation or error recovery. All student utterances are passed through the standard interpretation pipeline, so that the results can be analyzed later. However, the system does not attempt to address the student content. Instead, regardless of the answer analysis, the system always uses a neutral acceptance and bottom out strategy, giving the student the correct answer every time, e.g., "OK. One way to phrase the correct answer is: the open switch creates a gap in the circuit". Thus, the students are never given any indication of whether they have been understood or not.

The full policy acts differently depending on the analysis of the student answer. For correct answers, it acknowledges the answer as correct and optionally restates it (see (Dzikovska et al., 2008) for details). For incorrect answers, it restates the correct portion of the answer (if any) and provides a hint to guide the student towards the completely correct answer. If the student's utterance cannot be interpreted, the system responds with a help message indicating the cause of the problem together with a hint. In both cases, after 3 unsuccessful attempts to address the problem the system uses the bottom out strategy and gives away the answer.

[1] While there is no confidence score from a speech recognizer, our system uses a combination of a parse quality score assigned by the parser and a set of consistency checks to determine whether an interpretation is sufficiently reliable.

The content of the bottom out is the same as in the baseline, except that the full system indicates clearly that the answer was incorrect or was not understood, e.g., "Not quite. Here is the answer: the open switch creates a gap in the circuit".

The help messages are based on the Targeted-Help approach successfully used in spoken dialogue (Hockey et al., 2003), together with the error classification we developed for tutorial dialogue (Dzikovska et al., 2009). There are 9 different error types, each associated with a different targeted help message. The goal of the help messages is to give the student as much information as possible as to why the system failed to understand them but without giving away the answer.

In comparing the two policies, we would expect that the students in both conditions would learn something, but that the learning gain and user satisfaction would be affected by the difference in policies. We hypothesized that students who receive feedback on their errors in the full condition would learn more compared to those in the baseline condition.

3 Evaluation

We collected data from 76 subjects interacting with the system. The subjects were randomly assigned to either the baseline (BASE) or the full (FULL) policy condition. Each subject took a pretest, then worked through a lesson with the system, and then took a post-test and filled in a user satisfaction survey. Each session lasted approximately 4 hours, with 232 student language turns in FULL ($SD = 25.6$) and 156 in BASE ($SD = 2.02$). Additional time was taken by reading and interacting with the simulation environment. The students had little prior knowledge of the domain. The survey consisted of 63 questions on the 5-point Likert scale covering the lesson content, the graphical user interface, and tutor's understanding and feedback. For purposes of this study, we are using an averaged tutor score.

The average learning gain was 0.57 ($SD = 0.23$) in FULL, and 0.63 ($SD = 0.26$) in BASE. There was no significant difference in learning gain between conditions. Students liked BASE better: the average tutor evaluation score for FULL was 2.56 out of 5 ($SD = 0.65$), compared to 3.32 ($SD = 0.65$) in BASE. These results are significantly different (t-test, $p < 0.05$). In informal comments after the session many students said that they were frustrated when the system said that it did not understand them. However, some students in BASE also mentioned that they sometimes were not sure if the system's answer was correcting a problem with their answer, or simply phrasing it in a different way.

We used mean frequency of non-interpretable utterances (out of all student utterances in each session) to evaluate the effectiveness of the two different policies. On average, 14% of utterances in both conditions resulted in non-understandings.[2] The frequency of non-understandings was negatively correlated with learning gain in FULL: $r = -0.47, p < 0.005$, but not significantly correlated with learning gain in BASE: $r = -0.09, p = 0.59$. However, in both conditions the frequency of non-understandings was negatively correlated with user satisfaction: FULL $r = -0.36, p = 0.03$, BASE $r = -0.4, p = 0.01$. Thus, even though in BASE the system did not indicate non-understanding, students were negatively affected. That is, they were not satisfied with the policy that did not directly address the interpretation problems. We discuss possible reasons for this below.

We investigated the effect of different types of interpretation errors using two criteria. First, we checked whether the mean frequency of errors was reduced between BASE and FULL for each individual strategy. The reduced frequency means that the recovery strategy for this particular error type is effective in reducing the error frequency. Second, we looked for the cases where the frequency of a given error type is negatively correlated with either learning gain or user satisfaction. This is provides evidence that such errors are negatively impacting the learning process, and therefore improving recovery strategies for those error types is likely to improve overall system effectiveness,

The results, shown in Table 1, indicate that the majority of interpretation problems are not significantly correlated with learning gain. However, several types of problems appear to be particularly significant, and are all related to improper use of domain terminology. These were *irrelevant_answer*, *no_appr_terms*, *selectional_restriction_failure* and *program_error*.

An *irrelevant_answer* error occurs when the student makes a statement that uses domain termi-

[2] We do not know the percentage of misunderstandings or concept error rate as yet. We are currently annotating the data with the goal to evaluate interpretation correctness.

	full			baseline		
error type	mean freq. (std. dev)	satisfaction r	gain r	mean freq (std. dev)	satisfaction r	gain r
irrelevant_answer	0.008 (0.01)	-0.08	-0.19	0.012 (0.01)	-0.07	-0.47**
no_appr_ terms	0.005 (0.01)	-0.57**	-0.42**	0.003 (0.01)	-0.38**	-0.01
selectional_restr_failure	0.032 (0.02)	-0.12	-0.55**	0.040 (0.03)	0.13	0.26*
program_error	0.002 (0.003)	0.02	0.26	0.003 (0.003)	0	-0.35**
unknown_word	0.023 (0.01)	0.05	-0.21	0.024 (0.02)	-0.15	-0.09
disambiguation_failure	0.013 (0.01)	-0.04	0.02	0.007 (0.01)	-0.18	0.19
no_parse	0.019 (0.01)	-0.14	-0.08	0.022(0.02)	-0.3*	0.01
partial_interpretation	0.004 (0.004)	-0.11	-0.01	0.004 (0.005)	-0.19	0.22
reference_failure	0.012 (0.02)	-0.31*	-0.09	0.017 (0.01)	-0.15	-0.23
Overall	0.134 (0.05)	-0.36**	-0.47**	0.139 (0.04)	-0.4**	-0.09

Table 1: Correlations between frequency of different error types and student learning gain and satisfaction. ** - correlation is significant with $p < 0.05$, * - with $p <= 0.1$.

nology but does not appear to answer the system's question directly. For example, the expected answer to "In circuit 1, which components are in a closed path?" is "the bulb". Some students misread the question and say "Circuit 1 is closed." If that happens, in FULL the system says "Sorry, this isn't the form of answer that I expected. I am looking for a component", pointing out to the student the kind of information it is looking for. The BASE system for this error, and for all other errors discussed below, gives away the correct answer without indicating that there was a problem with interpreting the student's utterance, e.g., "OK, the correct answer is the bulb."

The *no_appr_terms* error happens when the student is using terminology inappropriate for the lesson in general. Students are expected to learn to explain everything in terms of connections and terminal states. For example, the expected answer to "What is voltage?" is "the difference in states between two terminals". If instead the student says "Voltage is electricity", FULL responds with "I am sorry, I am having trouble understanding. I see no domain concepts in your answer. Here's a hint: your answer should mention a terminal." The motivation behind this strategy is that in general, it is very difficult to reason about vaguely used domain terminology. We had hoped that by telling the student that the content of their utterance is outside the domain as understood by the system, and hinting at the correct terms to use, the system would guide students towards a better answer.

Selectional_restr_failure errors are typically due to incorrect terminology, when the students phrased answers in a way that contradicted the system's domain knowledge. For example, the system can reason about damaged bulbs and batteries, and open and closed paths. So if the student says "The path is damaged", the FULL system would respond with "I am sorry, I am having trouble understanding. Paths cannot be damaged. Only bulbs and batteries can be damaged."

Program_error were caused by faults in the underlying network software, but usually occurred when the student was using extremely long and complicated utterances.

Out of the four important error types described above, only the strategy for *irrelevant_answer* was effective: the frequency of *irrelevant_answer* errors is significantly higher in BASE (t-test, $p < 0.05$), and it is negatively correlated with learning gain in BASE. The frequencies of other error types did not significantly differ between conditions.

However, one other finding is particularly interesting: the frequency of *no_appr_terms* errors is negatively correlated with user satisfaction in BASE. This indicates that simply accepting the student's answer when they are using incorrect terminology and exposing them to the correct answer is not the best strategy, possibly because the students are noticing the unexplained lack of alignment between their utterance and the system's answer.

4 Discussion and Future Work

As discussed in Section 1, previous studies of short-answer tutorial dialogue systems produced a counter-intuitive result: measures of interpretation accuracy were not correlated with learning gain. With less restricted language, misunderstandings

negatively affected learning. Our study provides further evidence that interpretation quality significantly affects learning gain in tutorial dialogue. Moreover, while it has long been known that user satisfaction is negatively correlated with interpretation error rates in spoken dialogue, this is the first attempt to evaluate the impact of different types of interpretation errors on task success and usability of a tutoring system.

Our results demonstrate that different types of errors may matter to a different degree. In our system, all of the error types negatively correlated with learning gain stem from the same underlying problem: the use of incorrect or vague terminology by the student. With the exception of the *irrelevant_answer* strategy, the targeted help strategies we implemented were not effective in reducing error frequency or improving learning gain. Additional research is needed to understand why. One possibility is that *irrelevant_answer* was easier to remediate compared to other error types. It usually happened in situations where there was a clear expectation of the answer type (e.g., a list of component names, a yes/no answer). Therefore, it was easier to design an effective prompt. Help messages for other error types were more frequent when the expected answer was a complex sentence, and multiple possible ways of phrasing the correct answer were acceptable. Therefore, it was more difficult to formulate a prompt that would clearly describe the problem in all contexts.

One way to improve the help messages may be to have the system indicate more clearly when user terminology is a problem. Our system apologized each time there was a non-understanding, leading students to believe that they may be answering correctly but the answer is not being understood. A different approach would be to say something like "I am sorry, you are not using the correct terminology in your answer. Here's a hint: your answer should mention a terminal". Together with an appropriate mechanism to detect paraphrases of correct answers (as opposed to vague answers whose correctness is difficult to determine), this approach could be more beneficial in helping students learn. We are considering implementing and evaluating this as part of our future work.

Some of the errors, in particular instances of *no_appr_terms* and *selectional_restr_failure*, also stemmed from unrecognized paraphrases with non-standard terminology. Those answers could conceivably be accepted by a system using semantic similarity as a metric (e.g., using LSA with pre-authored answers). However, our results also indicate that simply accepting the incorrect terminology may not be the best strategy. Users appear to be sensitive when the system's language does not align with their terminology, as reflected in the decreased satisfaction ratings associated with higher rates of incorrect terminology problems in BASE. Moreover, prior analysis of human-human data indicates that tutors use different restate strategies depending on the "quality" of the student answers, even if they are accepting them as correct (Dzikovska et al., 2008). Together, these point at an important unaddressed issue: existing systems are often built on the assumption that only incorrect and missing parts of the student answer should be remediated, and a wide range of terminology should be accepted (Graesser et al., 1999; Jordan et al., 2006). While it is obviously important for the system to accept a range of different phrasings, our analysis indicates that this may not be sufficient by itself, and students could potentially benefit from addressing the terminology issues with a specifically devised strategy.

Finally, it could also be possible that some differences between strategy effectiveness were caused by incorrect error type classification. Manual examination of several dialogues suggests that most of the errors are assigned to the appropriate type, though in some cases incorrect syntactic parses resulted in unexpected interpretation errors, causing the system to give a confusing help message. These misclassifications appear to be evenly split between different error types, though a more formal evaluation is planned in the future. However from our initial examination, we believe that the differences in strategy effectiveness that we observed are due to the actual differences in the help messages. Therefore, designing better prompts would be the key factor in improving learning and user satisfaction.

Acknowledgments

This work has been supported in part by US Office of Naval Research grants N000140810043 and N0001410WX20278. We thank Katherine Harrison, Leanne Taylor, Charles Callaway, and Elaine Farrow for help with setting up the system and running the evaluation. We would like to thank anonymous reviewers for their detailed feedback.

References

V. Aleven, O. Popescu, and K. R. Koedinger. 2001. Towards tutorial dialog to support self-explanation: Adding natural language understanding to a cognitive tutor. In *Proceedings of the 10^{th} International Conference on Artificial Intelligence in Education (AIED '01)"*.

Dan Bohus and Alexander Rudnicky. 2005. Sorry, I didn't catch that! - An investigation of non-understanding errors and recovery strategies. In *Proceedings of SIGdial-2005*, Lisbon, Portugal.

Michelene T. H. Chi, Nicholas de Leeuw, Mei-Hung Chiu, and Christian LaVancher. 1994. Eliciting self-explanations improves understanding. *Cognitive Science*, 18(3):439–477.

Myroslava O. Dzikovska, Gwendolyn E. Campbell, Charles B. Callaway, Natalie B. Steinhauser, Elaine Farrow, Johanna D. Moore, Leslie A. Butler, and Colin Matheson. 2008. Diagnosing natural language answers to support adaptive tutoring. In *Proceedings 21st International FLAIRS Conference*, Coconut Grove, Florida, May.

Myroslava O. Dzikovska, Charles B. Callaway, Elaine Farrow, Johanna D. Moore, Natalie B. Steinhauser, and Gwendolyn C. Campbell. 2009. Dealing with interpretation errors in tutorial dialogue. In *Proceedings of SIGDIAL-09*, London, UK, Sep.

Myroslava O. Dzikovska, Johanna D. Moore, Natalie Steinhauser, Gwendolyn Campbell, Elaine Farrow, and Charles B. Callaway. 2010. Beetle II: a system for tutoring and computational linguistics experimentation. In *Proceedings of ACL-2010 demo session*.

A. C. Graesser, P. Wiemer-Hastings, P. Wiemer-Hastings, and R. Kreuz. 1999. Autotutor: A simulation of a human tutor. *Cognitive Systems Research*, 1:35–51.

Beth Ann Hockey, Oliver Lemon, Ellen Campana, Laura Hiatt, Gregory Aist, James Hieronymus, Alexander Gruenstein, and John Dowding. 2003. Targeted help for spoken dialogue systems: intelligent feedback improves naive users' performance. In *Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics*, pages 147–154, Morristown, NJ, USA.

Pamela W. Jordan, Maxim Makatchev, and Kurt VanLehn. 2004. Combining competing language understanding approaches in an intelligent tutoring system. In James C. Lester, Rosa Maria Vicari, and Fábio Paraguaçu, editors, *Intelligent Tutoring Systems*, volume 3220 of *Lecture Notes in Computer Science*, pages 346–357. Springer.

Pamela Jordan, Maxim Makatchev, Umarani Pappuswamy, Kurt VanLehn, and Patricia Albacete. 2006. A natural language tutorial dialogue system for physics. In *Proceedings of the 19th International FLAIRS conference*.

Pamela Jordan, Diane Litman, Michael Lipschultz, and Joanna Drummond. 2009. Evidence of misunderstandings in tutorial dialogue and their impact on learning. In *Proceedings of the 14th International Conference on Artificial Intelligence in Education (AIED)*, Brighton, UK, July.

Diane Litman and Kate Forbes-Riley. 2005. Speech recognition performance and learning in spoken dialogue tutoring. In *Proceedings of EUROSPEECH-2005*, page 1427.

Diane Litman, Johanna Moore, Myroslava Dzikovska, and Elaine Farrow. 2009. Generalizing tutorial dialogue results. In *Proceedings of 14th International Conference on Artificial Intelligence in Education (AIED)*, Brighton, UK, July.

Philip M. McCarthy, Vasile Rus, Scott Crossley, Arthur C. Graesser, and Danielle S. McNamara. 2008. Assessing forward-, reverse-, and average-entailment indeces on natural language input from the intelligent tutoring system, iSTART. In *Proceedings of the 21st International FLAIRS conference*, pages 165–170.

Rodney D. Nielsen, Wayne Ward, and James H. Martin. 2008. Learning to assess low-level conceptual understanding. In *Proceedings 21st International FLAIRS Conference*, Coconut Grove, Florida, May.

Heather Pon-Barry, Brady Clark, Elizabeth Owen Bratt, Karl Schultz, and Stanley Peters. 2004. Evaluating the effectiveness of SCoT: A spoken conversational tutor. In J. Mostow and P. Tedesco, editors, *Proceedings of the ITS 2004 Workshop on Dialog-based Intelligent Tutoring Systems*, pages 23–32.

Amruta Purandare and Diane Litman. 2008. Content-learning correlations in spoken tutoring dialogs at word, turn and discourse levels. In *Proceedings 21st International FLAIRS Conference*, Coconut Grove, Florida, May.