Improving interpretation robustness in a tutorial dialogue system

Myroslava O. Dzikovska    Elaine Farrow    Johanna D. Moore

School of Informatics, University of Edinburgh
{m.dzikovska,elaine.farrow,j.moore}@ed.ac.uk
Motivation

Dialogue in Interactive Learning Environments

- Science teaching involves problem-solving and hands-on experiments
- Asking students to verbalize their reasoning can help improve learning
- Support explanation in interactive learning environments
  - Operate in a dynamically changing environment
  - Give students detailed feedback to help them construct correct explanations
  - Feedback: scripted or dynamically generated
Motivation

Dynamic Adaptive Feedback Generation

(Freedman, 2000; Clark et al., 2005; Dzikovska, 2010)

- Semantic interpreter
  - Detailed analyses of correct and incorrect parts of student answer
  - Generally based on a hand-crafted parser

Problem: rule-based interpreters are brittle
Out-of-grammar utterances can cause interpretation failures
Dynamic Adaptive Feedback Generation

(Freedman, 2000; Clark et al., 2005; Dzikovska, 2010)

- Semantic interpreter
  - Detailed analyses of correct and incorrect parts of student answer
  - Generally based on a hand-crafted parser
- Tutorial planner to choose a feedback strategy
  - Prompt, hint, point out error, give away answer
Dynamic Adaptive Feedback Generation

(Freedman, 2000; Clark et al., 2005; Dzikovska, 2010)

- Semantic interpreter
  - Detailed analyses of correct and incorrect parts of student answer
  - Generally based on a hand-crafted parser
- Tutorial planner to choose a feedback strategy
  - Prompt, hint, point out error, give away answer
- Natural language generator to instantiate the strategy
  - Use the state of the environment and the information from interpretation to produce a contextualized feedback message

Problem: rule-based interpreters are brittle
Out-of-grammar utterances can cause interpretation failures
Dynamic Adaptive Feedback Generation

(Freedman, 2000; Clark et al., 2005; Dzikovska, 2010)

- Semantic interpreter
  - Detailed analyses of correct and incorrect parts of student answer
  - Generally based on a hand-crafted parser
- Tutorial planner to choose a feedback strategy
  - Prompt, hint, point out error, give away answer
- Natural language generator to instantiate the strategy
  - Use the state of the environment and the information from interpretation to produce a contextualized feedback message
- **Problem**: rule-based interpreters are brittle
  - Out-of-grammar utterances can cause interpretation failures
Outline

Approach

Experiment

Conclusions
Combining Symbolic and Statistical NLU

(Dzikovska et al, AIED 2013)

- Annotate corpus of student answers with simple accuracy labels responses
  - Correct, partially_correct_incomplete, contradictory, irrelevant, non_domain
- Labels align with common tutoring decisions
- Train a statistical classifier and combine it with the symbolic semantic interpreter
A Tutorial Dialogue System for Basic Electricity and Electronics

Exercise 3
Here is a circuit with 2 switches (Y and Z), either of which could be open or closed.

Dialogue
Tutor: Test your prediction for the fourth case, by closing both switches and observing the bulbs.
Tutor: Yes. When you're ready, go on to the next slide.
Tutor: Did the bulbs behave the way you expected?
Student: no
Tutor: Right. In the second case, you said that bulb B and bulb C would be lit. But they were off.
Tutor: Why was bulb A on when switch Y was open and switch Z was closed?
Student: It was in a closed path
Tutor: Right. Bulb A is contained in a closed path. Keep trying.
Student: I don't know
Tutor: Here's a hint. Your answer should mention a battery.
Student: The bulb got power from the battery
Tutor: Hmmmm... Here's a hint. The battery is contained in something.
Student: Bulb A and the battery were in a closed path
Tutor: That's great.
Tutor: Why was bulb C off when switch Z was open?
Student input
Architecture

**Question:** Why was bulb A on when switch X was open?
**Student Answer:** The battery was in a closed path
**Reference Answer:** Bulb A and the battery were contained in the same closed path
**Question:** Why was bulb A on when switch X was open?

**Student Answer:** The battery was in a closed path

**Reference Answer:** Bulb A and the battery were contained in the same closed path

**Interpreter**

**Partially Correct Incomplete**

- **Correct:** (Bulb B1) (Path _P1)
  - (is-closed _P1 true) (contains _P1 B1)

- **Missing:** (Battery B2) (contains _P1 B2)

- **Contradictory:** ()

**Hint object:** Your answer should mention a bulb

**Hint relation:** Bulb A was contained in something
**Architecture**

**Question:** Why was bulb A on when switch X was open?

**Student Answer:** The battery was in a closed path

**Reference Answer:** Bulb A and the battery were contained in the same closed path

**Interpreter**

**Partially Correct Incomplete**

- **Correct:** (Bulb B1) (Path P1)
  - (is-closed P1 true) (contains P1 B1)
- **Missing:** (Battery B2) (contains P1 B2)
- **Contradictory:** ()

**Classifier**

**Partially Correct Incomplete**

**Hint object:** Your answer should mention a bulb

**Hint relation:** Bulb A was contained in something

**Contentless Prompt:** Right. Anything else?
**Question:** Why was bulb A on when switch X was open?
**Student Answer:** The battery was in a closed path
**Reference Answer:** Bulb A and the battery were contained in the same closed path

**Interpreter**

**Partially Correct Incomplete**

**Correct:** (Bulb B1) (Path P1)
(is-closed P1 true) (contains P1 B1)

**Missing:** (Battery B2) (contains P1 B2)

**Contradictory:** ()

**Hint object:** Your answer should mention a bulb
**Hint relation:** Bulb A was contained in something

**Classifier**

**Partially Correct Incomplete**

**Contentless Prompt:** Right. Anything else?
Initial Evaluation

(Dzikovska et al., AIED 2013)

- Combine interpreter with classifier trained on approximately 3000 student responses
- Significant improvement in interpretation quality compared to semantic interpreter alone
- Best combination policy: use the output of the classifier only if semantic interpretation fails
Initial Evaluation

(Dzikovska et al., AIED 2013)

• Combine interpreter with classifier trained on approximately 3000 student responses
• Significant improvement in interpretation quality compared to semantic interpreter alone
• Best combination policy: use the output of the classifier only if semantic interpretation fails
• Next step: analyze non-interpretable utterance subset in more detail
Outline

Approach

Experiment

Conclusions
Distribution of labels in interpretable and non-interpretable utterances
A combination policy that uses a classifier only for interpretation failures can benefit from a classifier specific to non-interpretable utterances.
Experimental Setup

- Beetle portion of Student Response Analysis Corpus (Dzikovska et al., 2013)
  - \( \sim 3000 \) student answers to explanation questions in **BEETLE II** system
  - 36% of utterances rejected as non-interpretable
- 10-fold cross-validation
Classifiers and Combination Policies

- Classifiers: same features, different training sets
  - 20 lexical overlap and negation features
  - Sim20: trained on all data in the training folds
  - Sim20NI: trained on Non-Interpretable utterances only
Experiment

Classifiers and Combination Policies

- Classifiers: same features, different training sets
  - 20 lexical overlap and negation features
  - Sim20: trained on all data in the training folds
  - Sim20NI: trained on Non-Interpretable utterances only
- Two policies applied in case of interpretation failure
  - Best policy (in the talk): whenever interpretation fails, use the classifier result
  - Additional policy (reported in the paper): preserve some of the “non-understanding” messages
Evaluation metric: per-class and macro-averaged $F_1$ score

|               | Standalone | Interpreter + Classifier |
|---------------|------------|--------------------------|
|               | Interp.    | Sim20                    | Sim20          | Sim20NI       |
| correct       | 0.66       | 0.71                     | 0.70           | 0.70          |
| pc_incomplete | 0.48       | 0.40                     | 0.51           | 0.50          |
| contradictory | 0.27       | 0.45                     | 0.47           | 0.51          |
| irrelevant    | 0.21       | 0.08                     | 0.22           | 0.22          |
| non_domain    | 0.65       | 0.78                     | 0.83           | 0.83          |
| macro average | 0.45       | 0.48                     | **0.55**       | **0.55**      |
Outline

Approach

Experiment

Conclusions
Conclusions

- Investigated using a classifier to provide robustness where rule-based semantic interpretation fails
Conclusions

- Investigated using a classifier to provide robustness where rule-based semantic interpretation fails
- Utterances causing interpretation failures look different from interpretable utterances
Conclusions

- Investigated using a classifier to provide robustness where rule-based semantic interpretation fails
- Utterances causing interpretation failures look different from interpretable utterances
- The “non-interpretable” subset can be exploited to help system robustness
  - Retain the benefits of dynamic feedback generation on interpretable utterances
  - Target annotation to utterances known to be difficult for the symbolic interpreter
Future Work

- Show that the pattern we observed (contradictory utterances more problematic, can be used to train a classifier) applies to other domains
- Test classifiers with more sophisticated features
- Evaluate the combined system in user trials
Conclusions

Further Work on Understanding Student Explanations

Come see the posters from Student Response Analysis and Recognizing Textual Entailment Challenge on Saturday!