An Empirical Analysis of Learner Discourse

Bradley GOODMAN, Robert GAIMARI, Guido ZARRELLA, Frank LINTON
The MITRE Corporation,
202 Burlington Rd., Bedford MA 01803, USA

Abstract. We report the results of an empirical analysis of learner discourse wherein we analyze the utility of domain keywords in determining the topics of conversation. The results provide a foundation for agent-based feedback to learners in a collaborative learning environment. We also track user actions on a graphical user interface to determine learners' progress and the quality of their designs.

1. Introduction

We are developing a collaborative learning environment with an interface consisting of a text chat tool and a shared whiteboard that is specialized for the learning task of object-oriented software analysis and design (OOA&D), specifically, object modeling technique (OMT). The design is based on our observations of learners collaboratively solving software problems in a classroom-based training course. Our ultimate goal is to develop an intelligent agent that will facilitate group interaction, foster deliberative dialog, and optimize learning. In this paper we report progress in interpreting user actions on the whiteboard, and determining the topic of conversation in the text chat tool.

We track users' actions on the graphical user interface to determine the progress and quality of their designs by comparing them to valid solutions.

We report the results of an empirical analysis of learner discourse wherein we analyze the utility of a domain ontology in determining the topics of conversation. The results provide a foundation for agent-based feedback to learners.

1.1 Architecture

Each student in the group interfaces with the other students through Epsilon, a collaborative learning client with a shared agenda, whiteboard, and a chat tool with a speech act (sentence opener) interface [1]. The intelligent agent connects to this system with a modified version of Epsilon and monitors the group's interactions. The agent maintains a set of gauges and graphs representing the state of the group and student models, and can intervene when necessary by drawing on the whiteboard or sending messages to the chat tool.

The architecture of the agent itself is centered around a rule based reasoning engine built with Jess [2]. The agent's interface with Epsilon notifies the reasoner of each new student action. The agent also interprets each action with a number of trainable analysis tools that feed their results into the reasoner. These tools, mainly natural language processing neural networks, analyze the features of the event and provide supplemental information to the agent. The reasoning engine then applies rules to these inputs until a meaningful conclusion is reached. If the conclusion calls for some type of intervention, the software sends a message to the group chat window with feedback from the agent peer.

2. Topic Identification

When an agent is observing a group of learners in problem-solving mode with the goal of intervening to facilitate learning, the agent can intervene productively only to the extent it can observe and interpret the learners' actions. Tutoring a group of learners who are allowed to chat among themselves introduces the difficulties of natural language understanding. In this research we attempt to bypass this problem and explore the potential of keywords to reveal the topic of the conversation turns. We use *topic* to mean one main concept of the domain. A complete exercise is expected to address each topic at some level of complexity.

The introduction to object-oriented analysis and design (OOA&D) that we use in our study has five topics. In the vocabulary of the learners' dialogs, about 20 distinct keywords pertaining to the five topics appear. We hypothesized that the system could determine the current subject of a dialog segment by looking solely for the domain keywords.

To perform the analysis we selected a conversation log for one set of subjects and manually labeled each turn with its keyword related topic. Errors of commission occurred when keywords were used outside of their predicted context. Errors of omission occurred when statements did not include any of the relevant keywords, either because of vocabulary choice (idiosyncratic or exercise-specific word selection, misspellings) or missing context (pronouns, references to previous statements). To address vocabulary related errors, an exercise-specific vocabulary was added to the keyword list. To mitigate contextual errors, referential statements were assumed to be a continuation of the current topic.

The output from topic identification at the individual turn level is further analyzed to determine the topic of the entire dialog segment. By tracking the trend of topics, we avoid false topic changes from short sub-dialogs, misidentified topics, or missing context. The downside is that genuine conversation topic changes are not immediately detected.
We compared the topic-detection algorithm's output to that of an expert human judge, and found a correlation of 0.68.

3. Complete and Correct Solutions

One of the tutor agent's goals is that the students' work on a topic should be complete and correct for reasons discussed in [3]. In the case of OOA&D, a domain knowledge module can determine the extent to which a topic is currently complete and correct by interpreting the diagram of the solution as the learners are constructing it on the whiteboard drawing tool. Implementing model tracing [4] in the rule based architecture described above has been fairly straightforward for limited OOA&D.

Obtaining a correct and complete solution is only an enabling goal in service of the more important goal of having the learners master the skills and knowledge they are attempting to acquire. The agent will use its assessment of the completeness and correctness of the solution together with other measures to determine interventions that promote mastery. For example, if a topic has just been correctly completed on the whiteboard, and one learner's student model indicates she or he does not yet have a good understanding of the topic, the agent might prompt that learner for an explanation as to why that solution was chosen. The agent will use these results, along with an analysis of each learner's participation and contribution to the topic, to update each individual's student model.

The domain of OOA&D is comprised of objects that are grouped in similar *classes*, which in turn share *attributes*.

The relationships among classes are indicated by *links* with indicators of *multiplicity*. The agent stores a representation of the expert solution as a set of the necessary components of the solution. Each element is defined by the collection of attached criteria. A class' criteria are *name*, a list of *attributes*, and a list of *operators*. A link's criteria are *name*, *multiplicity*, and the two *classes associated by the link*.

As students add or edit parts of their solution on the whiteboard, rules in the reasoning system attempt to match each component to an element in an expert solution. If the defining criteria of two elements are functionally identical, the match between them is labeled as "exact". Partial matches are labeled as "likely" or "possible" depending on the number of criteria that are met.

The system can use this information to track both the overall correctness of the solution and the group's proficiency within each individual topic area of the problem. It can also measure the quantity and accuracy of contributions to the solution by each student. This data forms the basis for suggestions to the group or an individual about potential problem areas, and provides a context for interpreting the information extracted from the text chat.

4. Concluding Remarks

We have presented an architecture for a collaborative learning environment with an intelligent agent. To intervene effectively in the learning process, the agent must understand not only the learners' solutions to the exercises they produce on the specialized whiteboard, but also the topics of their conversation. Conventional model tracing allows the agent to interpret learner's actions on the whiteboard, but the task of interpreting learners' conversation is more difficult without incurring the overhead of deep natural language understanding. Instead, we use a speech act interface [1], combined with keyword analysis, to capture learner's intent and to determine the current topic of their conversation.
Ultimately, the agent will combine its knowledge of the completeness and correctness of each topic with its assessment of the learners' deliberation to make intervention decisions regarding the quality of the solution and the nature of the learners' small group interactions. The agent will also combine its knowledge of the correctness of each topic with its assessment of each learner's capability to perform that step, to make intervention decisions that either improve the learner's skill or to move on to another topic.

The agent will keep the conversation coherent and focusing on one topic at a time, while still permitting learner flexibility in topic selection. It will also help the learners reach a complete correct solution by encouraging discussion of incomplete or incorrect topics, and encourage each learner to increase their knowledge by measuring their degree of understanding and coaching them to interact with their peers.

References

[1] Soller, A., Goodman, B., Linton, F., and Gaimari, R. (1998). Promoting effective peer interaction in an intelligent collaborative learning environment. *Proceedings of the Fourth International Conference on Intelligent Tutoring Systems (ITS 98)*, San Antonio, TX, 186-195.

[2] Jess. http://herzberg.ca.sandia.gov/jess/

[3] Corbett, A.T. and Anderson, J.R. (1989). "Feedback timing and student control in the LISP Intelligent Tutoring System". Proceedings of the Fourth International Conference on AI and Education, 64-72.

[4] Anderson, J. R., Boyle, C. F., Corbett, A., & Lewis, M. (1990). Cognitive modeling and intelligent tutoring. Artificial Intelligence, 42, 7-49.