Topic Initiative in a Simulated Peer Dialogue Agent[1]

Pamela W. JORDAN [a]
[a] *Learning Research and Development Center, University of Pittsburgh*

Abstract. One goal of our project is to create a dialogue agent that can behave as a student peer and collaborate with a human student to explain or diagnose data structures programs. A human peer must be able to initiate something the agent may not be expecting and the agent must be able to respond to it as a peer would. This paper describes the work we have completed to extend an existing natural language tutorial dialogue system to provide such a capability and the work that remains.

Keywords. linguistics and language technology, communication, collaboration

1. Introduction

One goal of our project[2] is to create a dialogue agent that can behave as a student peer and collaborate with a human student to explain or diagnose data structures programs. To participate in the negotiations that take place during peer collaboration [2], the dialogue agent must be able to both recognize and initiate a number of communicative actions, such as assertions and proposals, about the code being explained or diagnosed. Although considerable attention has been given to the difficult task of intention recognition in planning and dialogue research, relatively less has been devoted to modelling how a peer reacts to intentions that may not immediately fit with her own. Both problems are rendered more manageable by modelling either fixed-initiative interactions or mixed-initiative interactions[3] that allow only one type of initiative to be accepted for a single turn (e.g. a question from a student during system-led tutoring) or one simple, uncontested switching mechanism between system and user-led interaction (e.g. "you do the next step"). The focus of the on-going work described here is to model the much richer negotiations between peers that switch the topic initiative (e.g. from our corpus, "D: that one does need to be head = insertBeginning(88, head). C: let me just draw what it does now, and then we can figure out how to change if after").

A pure finite-state based approach is relatively easy to develop and would allow us to closely model our corpus of human-human peer interactions but does not provide enough flexibility for us to experiment with peer negotiations. Alternatively, principle-based approaches would give us the needed flexibility but would require a prohibitive amount of

[1]This work was funded by NSF grant 0536959
[2]http://andes3.lrdc.pitt.edu/PaL
[3]See [1] for a survey of theories and work on mixed-initiative.

effort to develop the domain and communication knowledge to the point we could use it in experiments with students. We chose to use the existing TuTalk dialogue system for our initial modelling effort since it provides more flexibility than pure finite-state based approaches [4]. It provides a task agenda and discourse history, it tracks discourse obligations and has been used in experiments involving large numbers of students. In addition it's modular architecture will support us in supplementing both the natural language (NL) understanding and generation modules with a human interpreter who will correct the results of both. This human assist will allow us to focus on mixed-initiative issues instead of intention recognition issues. In this paper we describe how we are extending TuTalk to provide a mixed-initiative capability for peer interactions.

2. Extending TuTalk to support topic initiative

TuTalk supports NL tutorial dialogue in which the tutor tries to elicit a main line of reasoning from the student by a series of questions similar to CIRCSIM-tutor's directed-lines of reasoning [3]. Dialogues in TuTalk can be represented as a finite state machine. Each state contains a single tutor turn. The arcs leaving the state correspond to possible student response turns. When creating a state, a dialogue author enters the text for a tutor's turn and defines several classes of student responses as transition arcs, and indicates which state each arc leads to. An arc can also "call" a finite state network, which allows authors to create hierarchical dialogues. The author creates a multi-step hierarchically-organized recipe to a cover a topic. At the leaf nodes a step comprises a state and its arcs. If no student response classes are defined then by default the transition is to the next step of the recipe. Non-terminating nodes of the recipe are a call to a subrecipe. In TuTalk a state is called an initiation and the arcs from a state its responses.

The NL associated with states and arcs is represented in concept definitions. In the simplest case, a concept is a set whose elements are NL phrases. When a string is input, the dialogue manager asks the understanding module to determine what concepts it represents and determines transitions on the basis of the concept labels returned. Likewise when a concept is to be expressed, the dialogue manager asks the generation module to determine how to best express it in NL.

When there is NL input from the dialogue partner, the dialogue manager sends that input and the concepts for the expected responses to the NL understanding module. We extended the dialogue manager so that if the NL input does not match any of the expected responses, then it next sends the NL understanding module the concepts for the initiations in the first step of every recipe. Thus the dialogue manager now sets up tiers of concept labels (i.e. language models) for concept classification where the first tier contains concepts that can potentially fulfill a discourse obligation and the second tier contains likely concepts for initiating a new topic.

When the human dialogue partner produces an initiation for a step, the dialogue agent must decide which of the expected responses to produce. It can randomly select one as with the current implementation or it can use the information the author supplies about correctness and deliberately pick a response that is wrong. In addition, since one of the possible responses is always an *unanticipated-response*, it can indicate that it doesn't know how to respond.

When the human dialogue partner produces an initiation that is the first step in a recipe, we call this topic initiative. We more broadly define topic initiative as any part of a human dialogue agent's turn that matches a concept in the second tier of concepts as long as it was not previously recently matched according to the dialogue history. We make this exception because what was said earlier may have been repeated for increased coherency instead of being a show of initiative. Currently the extended system will always uptake a recognized initiative. If some of the concepts associated with the human partner's turn are in the first tier, and thus are expected, while some are from the second tier, the system will pursue responses to the first tier concepts and will later respond to those in the second tier. When the initiative topic is completed the system will attempt to resume whatever was interrupted. The dialogue manager makes use of discourse obligations to resume interrupted topics after topic initiative. If the human partner interrupts and then resumes an interrupted topic, the dialogue manager prunes any remaining obligations that arose during the interruption.

3. Future Work

There are two extensions that we intend to address next. First we need to add information about the relative importance of topics so that the agent can decide whether to accept, ignore or postpone topic initiative instead of always accepting it. If the dialogue agent's last turn is a step in a topic of equal or greater importance than the human's initiation then the human's initiative would be postponed. Otherwise it is accepted. If the topic is not identified then the initiative is ignored. To properly postpone, the system needs to first offer an acknowledgment of the initiative that communicates it has been understood and will be entertained once the current topic is completed as with "first answer my question and then we'll talk about <topic>". Then it should put the postponed initiative as a goal on the dialogue manager's agenda before retrying the current step. Finally, it must decide where to put the postponed initiative on the agenda.

The second addition addresses resuming an interrupted topic. We need to add information on the relationships between topics to aid in deciding whether to abandon or retry the unfulfilled step, as our model currently does, or to retry some higher level topic goal. Finally, we need to add an appropriate NL transition back to the interrupted topic to reduce the chance of an incoherency because of the topic shift. Once these two additional extensions are completed, we will evaluate our approximate model by comparing task performance and learning gains of students working with either it, a tutor-only version of the agent, or another student.

References

[1] R. Cohen, C. Allaby, C. Cumbaa, M. Fitzgerald, K. Ho, B. Hui, C. Latulipe, F. Lu, N. Moussa, D. Pooley, A. Qian, and S. Siddiqi. What is initiative? *User Modeling and User-Adaptive Interaction*, 8:171–214, 1998.

[2] B. Di Eugenio, P. W. Jordan, R. H. Thomason, and J. D. Moore. The agreement process: an empirical investigation of human-human computer-mediated collaborative dialogues. *International Journal of Human Computer Studies*, 53(6):1017–1076, 2000.

[3] M. Evens and J. Michael. *One-on-One Tutoring by Humans and Computers*. Lawrence Erlbaum Associates, Inc., 2006.

[4] P. W. Jordan, M. Ringenberg, and B. Hall. Rapidly developing dialogue systems that support learning studies. In *Proceedings of ITS06 Workshop on Teaching with Robots, Agents, and NLP*, 2006.