An Intelligent Discussion-Bot for Guiding Student Interactions in Threaded Discussions

Jihie Kim
Erin Shaw
Grace Chern
Donghui Feng

University of Southern California
Information Sciences Institute
Outline

• Discussion-Bot Framework
• Modeling Student Interactions in On-Line Discussions
• Modeling Student Interactions with Speech Act Classifiers
• Current Results
• Summary and Future Work
### USC/DEN ISI DB Online Learning

![Screenshot of the USC/DEN ISI DB Online Learning platform](image)

#### Courses

<table>
<thead>
<tr>
<th>Discussions</th>
<th>Topics</th>
<th>Posts</th>
<th>Last Post</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSCI402x_20053</td>
<td>7</td>
<td>25</td>
<td>17 Sep 2005 23:49 (noles)</td>
</tr>
<tr>
<td>ISI Discussion Board Questions</td>
<td>2</td>
<td>9</td>
<td>16 Sep 2005 14:08 (caro/aiine)</td>
</tr>
<tr>
<td>Looking for group partners</td>
<td>40</td>
<td>99</td>
<td>16 Sep 2005 00:51 (shah)</td>
</tr>
<tr>
<td>Grader/TA Office Hour Notices</td>
<td>8</td>
<td>10</td>
<td>16 Sep 2005 12:44 (nick)</td>
</tr>
<tr>
<td><strong>Nachos General Questions</strong></td>
<td>12</td>
<td>27</td>
<td>11 Sep 2005 11:10 (noles)</td>
</tr>
<tr>
<td>402 Humor</td>
<td>5</td>
<td>9</td>
<td>16 Sep 2005 10:44 (shaf)</td>
</tr>
<tr>
<td>Project 1 Questions</td>
<td>36</td>
<td>101</td>
<td>10 Sep 2005 10:21 (shaf)</td>
</tr>
<tr>
<td>Project 2 Questions</td>
<td>0</td>
<td>0</td>
<td>No Posts</td>
</tr>
<tr>
<td>Project 3 Questions</td>
<td>0</td>
<td>0</td>
<td>No Posts</td>
</tr>
<tr>
<td>Project 4 Questions</td>
<td>0</td>
<td>0</td>
<td>No Posts</td>
</tr>
</tbody>
</table>

**New Topics**
- Reading list #3 (09/19/05 13:01)
- DNS Generation (09/18/05 13:19)
- Laptop failure (09/18/05 12:44)
- Anyone have another (09/16/05 10:44)
- Assertion failed Err... (09/14/05 20:41)

**RSS Feed**
- [http://www.spnews.com/](http://www.spnews.com/)

**Recent Topics**
- [CSCI402] Discussion (09/04/05 18:05)
- [csci602] New (09/14/05 11:01)
Courses Involved

Past
• Five semesters of Undergraduate CS Operating System, USC
• Two semesters of Graduate CS Advanced Operating Systems, USC
• One semester of Psychology of Women course at the University of Massachusetts
• One semester of Engineering Negotiation for Collaborative Product Development, USC

Ongoing
• Undergraduate CS Operating System, USC
• Graduate Security Systems, USC
• Formal Languages and Automata, UC Irvine
• Undergraduate CS Operating System, Michigan Technological Univ.

~500 past students, ~150 current students
Over 7000 messages
Discussion-Bot Framework

• Modeling and assessing student interactions in online discussions
• Handling many student queries
• Guiding/scaffolding student interactions
Discussion-Bot: Responding to Student Queries

(Feng, Shaw, Kim, Hovy IUI-2006)
Modeling Discussion

Individual Messages

Response/Replies

Discussion threads
The Professor gave us 2 methods for forking threads from the main program. One was ....... The other was to ........ When you fork a thread where does it get created and take its 8 pages from? Do you have to calculate ......? If so how? Where does it store its PCReg ......? Any suggestions would be helpful.

read the student documentation for the Fork syscall

I am still confused. I understand it is in the same address space as the parent process, where do we allocate the 8 pages of mem for it? And how do we keep track of .....? … I am sure it is a simple concept that I am just missing.

If you use the first implementation...., then you'll have a hard limit on the number of threads....If you use the second implementation, you need to....

Either way, you'll need to implement the AddrSpace::NewStack() function and make sure that there is memory available.

...
Modeling and Assessing Student Interactions

• Contribution content
  ✓ • Topic of the discussion, topic coherence
  ✓ • Quality of the content (e.g. technical term uses)

• Role of each participant and his/her contribution
  e.g. person who asks many questions on a particular topic

• Interaction patterns in threads
  ✓ e.g. long vs. short discussions
  ✓ e.g. threads that reach an agreement on a topic versus threads that have unanswered queries
  e.g. effect of instructor intervention

• Interaction changes over time
  e.g. topic changes over a semester
- Discussion threads are often very short, many consisting of only one or two messages
- Student jump into programming details without understanding what is to be programmed or related technical concepts
- TA and instructors are not always available to fully guide interactions
Discussion Topic Analysis

- Automatically classify discussion *threads topics* and model *topic shifts* within each thread.

(Feng, Kim, Shaw, Hovy, AAAI-2006)
### Modeling Interactions: Speech Act Categories

*(Feng, Shaw, Kim, Hovy, HLT/NAACL 2006)*

<table>
<thead>
<tr>
<th>Speech Act</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACK</td>
<td>Acknowledge</td>
<td>Confirm or acknowledge</td>
</tr>
<tr>
<td>CANS</td>
<td>Complex Answer</td>
<td>Give answer requiring a full description of procedures, reasons, etc.</td>
</tr>
<tr>
<td>ANNO</td>
<td>Command</td>
<td>Command or announce</td>
</tr>
<tr>
<td>COMP</td>
<td>Compliment</td>
<td>Praise an argument or suggestion</td>
</tr>
<tr>
<td>CORR</td>
<td>Correct</td>
<td>Correct a wrong answer or solution</td>
</tr>
<tr>
<td>CRT</td>
<td>Criticize</td>
<td>Criticize an argument</td>
</tr>
<tr>
<td>ELAB</td>
<td>Elaborate</td>
<td>Elaborate on a previous argument or question</td>
</tr>
<tr>
<td>OBJ</td>
<td>Object</td>
<td>Object to an argument or suggestion</td>
</tr>
<tr>
<td>QUES</td>
<td>Question</td>
<td>Ask question about a specific problem</td>
</tr>
<tr>
<td>SANS</td>
<td>Simple Answer</td>
<td>Answer with a short phrase or few words</td>
</tr>
<tr>
<td>SUG</td>
<td>Suggest</td>
<td>Give advice or suggest a solution</td>
</tr>
<tr>
<td>SUP</td>
<td>Support</td>
<td>Support an argument or suggestion</td>
</tr>
</tbody>
</table>

**Inspired by**

*(Austin, 1962; Searle, 1969)*
Speech Act Categories Explored

Code 1
% agreement: 63
Kappa: 0.54

Code 2
% agreement: 92
Kappa: 0.58

Code 3
% agreement: 81
Kappa: 0.70

Name
- QUES: Question
- ANNO: Announcement
- CANS: Complex Answer
- SANS: Simple Answer
- SUG: Suggest
- ELAB: Elaborate
- CORR: Correct
- OBJ: Object
- CRT: Criticize
- SUP: Support
- ACK: Acknowledge
- COMP: Complement

Code 1

Code 2

Code 3

Code 1

Code 2

Code 3

% agreement: 81
Kappa: 0.70

% agreement: 63
Kappa: 0.54

% agreement: 92
Kappa: 0.58

Kappa = \frac{\text{Observed agreement} - \text{Chance agreement}}{\text{Total observed} - \text{Chance agreement}}
Speech Acts in a Discussion Thread

S1
I am still confused. I understand it is in the same address space as the parent process, where do we allocate the 8 pages of mem for it? And how do we keep track of .....? ... I am sure it is a simple concept that I am just missing.

S2
read the student documentation for the Fork syscall

S3
If you use the first implementation, then you'll have a hard limit on the number of threads...If you use the second implementation, you need to.... Either way, you'll need to implement the AddrSpace::NewStack() function and make sure that there is memory available.
Statistics of Speech Acts

<table>
<thead>
<tr>
<th>Speech Act</th>
<th>Description</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACK-SUP-COMP</td>
<td>An acknowledgement, compliment or support in response to a previous message</td>
<td>8.5</td>
</tr>
<tr>
<td>INFORM</td>
<td>Information, Command or Announcement</td>
<td>6.7</td>
</tr>
<tr>
<td>ANS-SUG</td>
<td>A simple or complex answer to a previous question. Suggestion or advice</td>
<td>37.8</td>
</tr>
<tr>
<td>CORR-OBJ</td>
<td>A correction or objection (or complaint) to/on a previous message</td>
<td>9.7</td>
</tr>
<tr>
<td>ELAB</td>
<td>An elaboration (of a previous message) or description, including elaboration of a question or an answer</td>
<td>8.1</td>
</tr>
<tr>
<td>QUES</td>
<td>A question about a problem, including question about a previous message</td>
<td>29.2</td>
</tr>
</tbody>
</table>
Automatic SA Classifiers

- Cleaning/preprocessing/transformation of raw data
- N-gram features and Linear SVM

<table>
<thead>
<tr>
<th>Category</th>
<th>1-gram</th>
<th>2-grams</th>
<th>3-grams</th>
<th>4-grams</th>
</tr>
</thead>
<tbody>
<tr>
<td>Question Classifier (QC)</td>
<td>? [categ_wh ] will do confused</td>
<td>do [categ_person] [tech_term] ? can [categ_person] is there ? thanks</td>
<td>[categ_wh] should wondering [or/and] do [categ_person] is there a [tech_term] ?</td>
<td>do [categ_person] have to do [categ_person] need to [tech_term] [tech_term] [tech..] ? is there a better does this mean that</td>
</tr>
<tr>
<td>Answer Classifier (AC)</td>
<td>yes am helps but depends</td>
<td>look at [or/and] do seems like in [tech_term] stated in</td>
<td>look at the for example , . [categ_person] should let [me/him/her/us] know not seem to</td>
<td>[categ_person] am a [tech_term] do [categ_person] have to look at the [tech_term] in the same [tech_term]</td>
</tr>
</tbody>
</table>

- Accuracy: QC – 88% and AC – 73%
Thread Classification with SA Classifiers

1) whether the given thread contains questions
2) whether the questions were answered or not.

- 70-75% of the predictions from the system were consistent with human answers (Ravi & Kim, AIED 2007)
Related Work

- Email Speech Act analysis (Carvalho and Cohen 2005)
- Dialogue analysis for intelligent tutoring systems (Graesser et al., 2001)
- Dialogue act analysis, surface cues (Samuel 2000; Hirschberg and Litman 1993)
- Topic analysis (Joachims, 1997; Liu et al., 2004; Yang et al., 2005)
- Improving Questions Answering with Speech Act Classifiers (Feng, Shaw, Kim, Hovy HLT-NAACL 2006)
- Thread summarization (Zhou and Hovy 2005)
- Predicting the likelihood of a message receiving a reply (Arguello et al., 2006)
- Computer supported collaborative argumentation (Shum 2000)
- Collaborative interaction in learning (Soller and Lesgold 2003)
Summary

• Modeling and Guiding Student Interactions in On-Line Discussions
  • Modeling student interactions with SA classifiers
  • Finding discussion threads that may need instruction attention

• Ongoing Work
  • Classifiers for other speech act types
  • Integration of interaction modeling and question answering: when to intervene
  • Developing scaffolding techniques