

Annotation of Initiative

Edinburgh Tutorial Dialogue Group*

January 22, 2002

The goal of this annotation is to see how initiative (control of the dialogue) shifts as the dialogue progresses.

1 Form

According to Whittaker and Stenton (1988), initiative can be determined if it is known whether an utterance is a question, command, statement, or positive-response. The first phase of annotation is labeling utterances with this information.

questions include yes-no questions (*e.g.*, “did you connect the leads?”), wh-questions (*e.g.*, “how should you connect the leads?”, “where should you connect the leads?”), requests for confirmation (*e.g.*, “the red lead goes on tab five right?”), and requests for information (*e.g.*, “explain how you computed that”). Note, not every utterance ending in a question mark is a question. The question mark may simply indicate uncertainty in a student answer and not constitute a question:

```
question T: What is the correct value for the resistor?  
statement S: 10 /4000 ?
```

Paraphrases are also considered questions because the listener should confirm or deny the correctness of the paraphrase.

```
statement T: a load is "using current up"  
statement     therefore a load is the opposite of a source  
question     S: a load consumes energy  
prompt       T: very good!
```

Note, although syntax is a useful guide, speakers can intend to use commands such as the ones below to elicit information from the student.

*Note, originally, the tag prompt was called PosResponse but we changed the name to better fit the terminology in the literature. We also removed some Edinburgh-specific details in the tool section.

statement T: voltage should not considered a given
question look again at the information that is given in the lab instructions,
and think again about what your first step should be
statement S: multiply the 4 miliamps by the values of the resistor divided by 1000

commands include imperatives (*e.g.*, “please go ahead and try that in the simulation”) as well as more indirect commands (*e.g.*, “you may continue”). Commands include commands to perform mental actions such as “remember that polarity must be observed”.

statements are declarative utterances (*e.g.*, “the red lead goes on tab 5”) as well as answers to questions:

question T: where should you connect the red lead?
statement S: tab 5

Any signals of politeness (“hello”, “sorry”, “thanks”) should be labeled statements.

prompts are short positive responses (*e.g.* “yes”, “correct”, “that’s right”, “okay”, “oh”, “I see”). Note, prompts such as “okay” might not be meant to convey acceptance. A better name for this category might be positive and lukewarm responses.

Do not get these responses mixed up with answers to questions which are statements.

question S: the red lead goes on tab five right?
statement T: yes
statement that’s right

If the first utterance of the dialogue is a word such as “okay” it is a statement because there is nothing to respond to.

2 Dialogue End

The corpus is made up of individual tutoring sessions. Each session is broken into turns: sequences of utterances by the same speaker. However we need to further divide the session. Consider the following dialogue:

10:09:33 1 S: please let me try again
10:09:39 2 T: Sure,
10:09:39 3 go ahead
10:10:56 4 If you hook up the red lead of the ammeter to the negative end
of the battery, what problem do you see?

Utterances 2-4 are all part of the same turn but are part of different dialogues. In utterances 2-3, the tutor gives the student permission to try again. There is then a pause of 1 minute (and 17 seconds) and the tutor asks a question. Presumably in this time, the student has tried to solve the problem again

in the circuit simulation. You should thus mark utterance 3 as the end of a dialogue.

Your goal in this annotation is to mark **dialogue endings**; we define dialogue endings as places where the student returns to the circuit simulation or quiz program. Unfortunately we have no record of the student's interaction with the simulation and quiz program. Thus, you will need to look for examples such as the one above where it does not make sense for a group of utterances to be together in one turn. In the above example, it does not make sense to give a command ("go ahead") and then immediately ask a question in the same dialogue. You can look at time stamps to verify that a significant pause occurred after the potential dialogue ending.

Another problem with not having a trace of the student's actions is that the computer may ask the student a question such as "what is the voltage in a particular situation". Thus, a student may start a dialogue simply by saying "2790.4" (*i.e.*, giving the answer to the human tutor instead of the computer). These should always occur at the start of dialogues.

3 How to use the tool

Each student in our corpus of tutoring dialogues generally was tutored twice resulting in two logfiles. To annotate a logfile, open a terminal window and type:

```
xhost +sorley
rlogin sorley
setenv DISPLAY urquhart:0
cd /projects/ltg/projects/Mate/java
```

To annotate a logfile type:

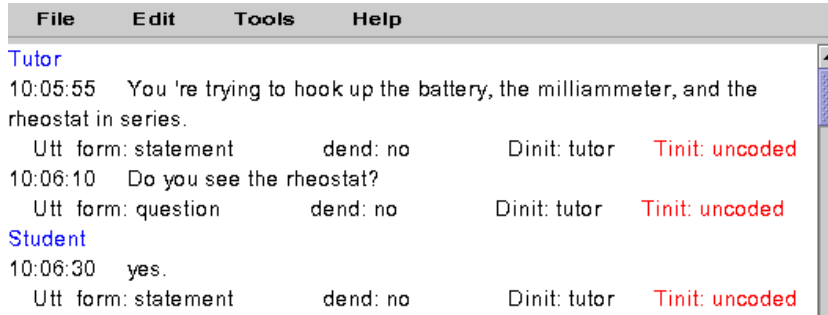
```
java1.2.1 mate.Workbench NewProjects/bee/<student-name>/<log>-<username>/bee.mp
```

< *log* > is either log1 or log2 depending on whether you are annotating the first or second tutoring session for a particular student.

< *student - name* > takes on the following values:

```
stud1, stud10, stud11, stud12, stud13, stud14, stud16, stud17, stud18,
stud19, stud23, stud25, stud27, stud28, stud29, stud3, stud31
```

After the program finishes loading you should see a window with the dialog:



Each turn of the dialogue is labeled with its speaker. In the example above, the tutor starts the dialogue. After the speaker's name is the first utterance of the turn (in black). The utterance is preceded by its time stamp.

Underneath the utterance are the labels for that utterance. Currently, you will only be worrying about the **form** and **dend** (dialogue end) labels. We have tried to automatically label form and dend. Sometimes though, what is labeled is wrong and you will need to correct it. To change a form label, use the left mouse button to click on the label. For example, if form is question and you want to change it to statement, then click on the word question. A menu with the options (question, prompt, command, statement, uncoded, and Remove utterance) should appear. Click on your choice but please ask me before you click on Remove utterance. Note, unlike menus in Netscape and other programs you should not hold the mouse button down to activate the menu. Just click once to get the menu and click a second time to make your selection. This same procedure holds true for dend except that the menu for dend has the options: yes, no, and Remove utterance. We have colored cases where the computer labeled dend as "yes" red to make them easier to spot.

You can save your annotation at any time by selecting "save coding" from the file menu. You can quit at any time by selecting "close coding" from the file menu: WARNING if you have not saved your data and you close the tool you will lose your changes.

If you want, you may save your data, close the tool, and finish annotation of the file another time. First backup your data, by typing the following (all on one line):

```
~/markc/bin/BEE/backup <student-name> <log>
```

Next, just repeat the steps outlined above for loading the program.