

# CSSE 413: Artificial Intelligence

## Dialog NLP Lab Manual

Name: \_\_\_\_\_Doris Chen\_\_\_\_\_

### 1) Parsing.

- a. [8 pts] The ability to tell the robot to repeat a command, such as: "again," "do that again," "more left," "further left."

```
> down
Got it.
I think you want me to move down
> again
Ja ja.
doris is going to MOVE_DOWN
> repeat
OK!
Got it! doris will immediately MOVE_DOWN
> do that again
[(obj,that/DT), (advmod,again/RB)]
Got it.
Sure, doris is going to MOVE_DOWN
> more down
Got it.
I think you want me to move down
> further down
10-4.
Sure, doris is going to move down
```

- b. [8 pts] The ability to tell the robot to undo the prior move command, such as: "undo," "go back."

```
> right
OK!
I think you want me to move right
> undo
Yes, undo.(VB)
Ja ja.
I think you want me to move left
> go left
[(advmod,left/RB)]
OK!
Sure, doris is going to move left
> go back
[(advmod,back/RB)]
Got it.
Sure, doris is going to Yes, going back
Ja ja.
Got it! doris will immediately move right
.
```

- c. [10 pts] The ability to deal with negation. If the user inputs "not left," or "do not go left," etc. the robot should not go left. Explain how you implemented this feature.

```
> not left
Roger that.
Sure, do nothing.
> do not left
10-4.
Sure, do nothing.
> do not go to left
OK!
Sure, do nothing.
```

I simply used a `String.contains()` method at where the input is a string instead of a semantic graph, and I checked if there is “not” in the command. If there is, then do nothing.

```

125     if (name.contains("not")) {
126         System.out.println(getRandom(this.reponses));
127         System.out.println("Sure, do nothing.");
128         this.prevAct = Action.DO_NOTHING;
129         return Action.DO_NOTHING;
130     }

```

- d. [30 pts] Your software should be able to parse any sentence in which the VB is "clean" or "move", but only those sentences that make sense. Please explain how you implemented this feature.

I looked up the graph structure when I typed these sentences:

> Please move to the right space.

> Please move to the space on your right.

> Please move to the space on you right.

“clean” or “move” is either in the pair of the root, or in the pair of children of root.

For 1) “clean” or “move” is in the pair of the root, I looked for “clean” and “move” in looping through second one of `childPairs(root)`

For 2) “clean” or “move” is in the pair of the one of root’s children, I did: for each child of root, check “clean” or “move” of the second one of child’s pair(`childPairs(pair.second)`), using a nested for loop.

```

> Please move to the right space.
[(discourse,Please/UH), (obl:to,space/NN), (punct,./.)]
Got it.
I think you want me to go right
> Please move to the space on your right.
[(discourse,Please/UH), (obl:to,space/NN), (punct,./.)]
Ja ja.
doris is going to go right
> Please move to the space on you right.
[(discourse,Please/UH), (obl:to,space/NN), (advmod,right/RB), (punct,./.)]
10-4.
Ok, doris will move right

```

- 2) [20 pts] **Keyword Search.** List the *five* variations that your system uses in case of a successful keyword search.

Even I typed sth wrong in the sentence, the robot will find the keyword which is “clear” or “move”

`this.myname` is robot’s name the user gave

```

"I think you want me to ";
this.myname + " is going to ";
"Ok, " + this.myname + " will ";
"Sure, " + this.myname + " is going to ";
"Got it! " + this.myname + " will immediately ";

```

Ex:

```

> please go to the left
Got it.
Ok, doris will move left
> Please clean the tile.
OK!
Got it! doris will immediately clean.

```

```
> plase fo clean
Roger that.
Sure, doris is going to clean.
```

- 3) [10 pts] **Random Responses.** List the *ten* phrases that your system uses to request clarification.  
 this.myname is robot's name the user gave

```
"I'm sorry but I'm not sure I understand. Could you say it in another way for " + this.myname + "?";
"I didn't quite get that. Can you clarify that for " + this.myname + "?";
"Sorry, could you rephrase that for " + this.myname + "?";
"I didn't catch that, could you please try one more time?";
"Sorry, could you elaborate on that?";
"I didn't understand that. Please try something else.";
"Sorry, what do you want " + this.myname + " to do?";
"Sorry, could you be more specific?";
"Sorry, " + this.myname + " is confused about what you said, could you please try again?";
"I'm not sure what you want me to do, could you make it more clear?";
> ohhhhhhhhhhhhh
I'm sorry but I'm not sure I understand. Could you say it in another way for doris?
> Pick up right
No, pick isn't valid.
I didn't quite get that. Can you clarify that for doris?
> No no no
Sorry, what do you want doris to do?
```

- 4) [14 pts] **Human-robot Interaction (HRI).**  
 a. Does your robot have a name and can they report it?

Yes. My robot will have a name which user gives, and they can report the name when user ask.

```
> What is your name?
Hello! My name is doris, and I am a intelligent cleaner!!
10-4.
> Who are you?
Hello! My name is doris, and I am a intelligent cleaner!!
Ja ja.
```

- b. Can your system name the robot and can the robot remember it?

Yes. My robot doesn't have name at the beginning, and it will ask for a name once the system started.

```
Hello! I am your private cleaner, would you please give me a name?
> doris
Got it.
Got the name!
```

- c. Please list the (at least three) different praise phrases.  
 1) Good!  
 2) Nice work!  
 3) I like your work

4) Well done!  
5) Nice!  
> Good!  
Happy to work with you!  
Ja ja.  
> Nice work  
Thanks, nice to meet you!  
OK!  
> Well done!  
Thank you! I've tried my best!  
Roger that.

d. Please list the (at least five) different acknowledgement phrases:

1) Got it.  
2) Roger that.  
3) 10-4  
4) Ja ja.  
5) OK!  
> Move to right  
[(xcomp,right/RB)]  
10-4.  
Ok, doris will move right  
> up  
Ja ja.  
doris is going to move up  
> down  
Got it.  
I think you want me to move down

5) [10 pts] **Extra credit:** Add a speech-to-text component.

Did you implement this component? \_\_No\_\_

6) [10 pts] **Extra credit:** Add a text-to-speech component.

Did you implement this component? \_\_No\_\_

Paste an annotated sample dialog here. It should demonstrate your systems capabilities as specified above. Ensure that the dialog covers at least one phrase from each of the items (1) – (4). Please use a different font (or bold face or color) to explain which of the features a dialog item demonstrates. Feel free to err on the side of plenty.