EPFL

CHILI
computer-human interaction
in learning and instruction

LEARN
Center for Learning Sciences

Help a humanoid robot understand my verbalised intentions!

Author:
Yiwen Ma

Supervisor(s):
Barbara Bruno , Utku Norman

Professor:
Pierre Dillenbourg

January 7, 2022

Contents

1 Introduction

In a research line of JUSThink project[1], the mutual understanding skills for a humanoid robot have been developed in the context of a collaborative problem-solving activity – "JUSThink", that aims to improve the computational thinking skills of human, by applying abstract and algorithmic reasoning to solve an unfamiliar problem on networks [2]

The robot currently relies on the human's use of the touch screen and acts via direct commands to the activity as well as verbalizing its intention and actions. The aim of this project is to endow a humanoid robot Reachy with the ability to understand the verbalized intentions of the human, in order to enhance the interactions. The goal of the improvement of the verbal skills of the robot includes:

1. recognise the verbal content of speech via a speech-to-text tool like Google reliably in real-time (i.e. automatic speech recognition), and
2. detect the intention of the human within the context of a learning activity (i.e. natural language understanding).

2 Background Scenarios

The project is based on a human-robot collaborative learning activity, JUSThink World [2], whose scenario aims to improve the computational thinking skills of the school children by applying abstract and algorithmic reasoning to solve an unfamiliar problem on networks.

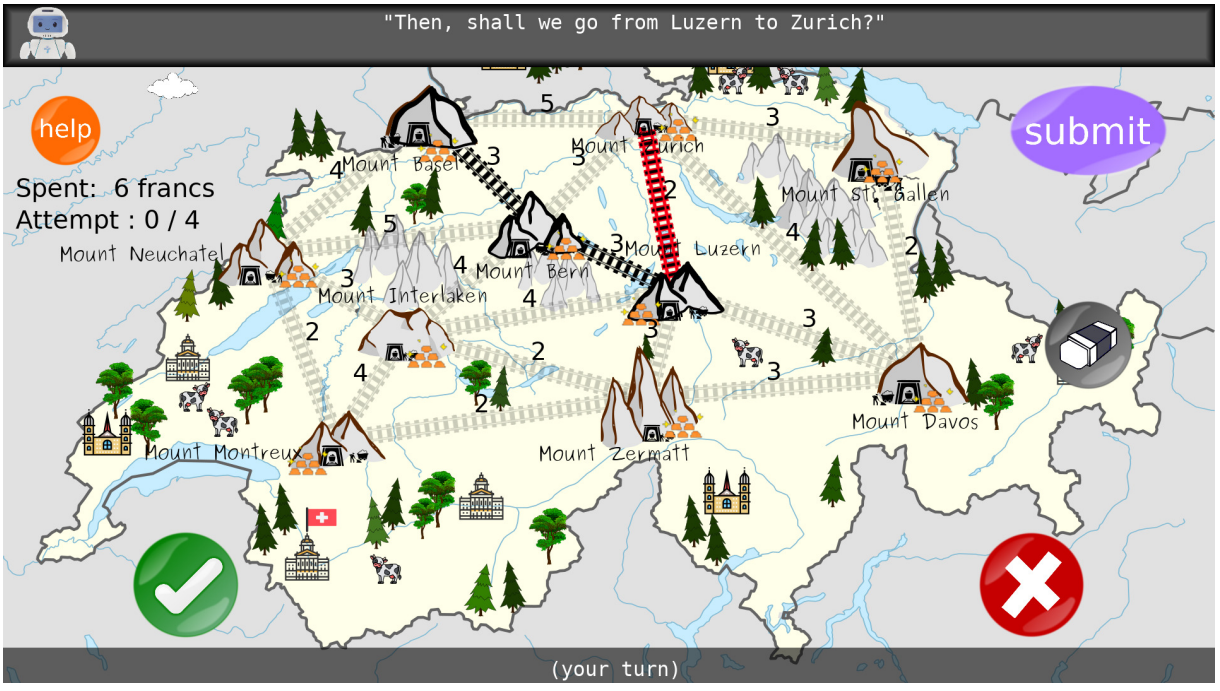

Figure 1: A snapshot of the JUSThink activity

[1] *JUSThink Project - CHILI - EPFL* https://www.epfl.ch/labs/chili/index-html/research/animatas/justhink/

[2] Utku Norman. *JUSThink_World (currently private).* Github repo: https://github.com/utku-norman/justhink_world

The scenario consists of individual (e.g. as in a test for assessment) and collaborative (with a robot) activities. This project is implemented on the collaborative activity, in which the human and the robot as (same-status) peers collaboratively construct a solution to this problem by deciding together which tracks to build, and submit it as their solution to the system. They take turns in suggesting to select/pick a specific connection, where the other either agrees or disagrees with this suggestion. A track will be built only if it is suggested by one and accepted by the other. In the activity, screens tend to be used as primary input devices, on which the instruction from the robot will be shown at the top middle. We want to complement it with verbal interaction: by what the user says at least partially understood by the robot for richer interaction. A sample dialogue happens between a user and the robot while playing the activity is shown as below:

Player: Let's build a bridge between **Bern** and **Luzern**!

(`Condition 1:Both location keywords are detected`)
(`A connection is made between Mount Bern and Mount Luzern`)
Robot: You have successfully connected **Bern** and **Luzern**. Do you agree with that?
Player: Yes, I agree!
(`Intention agree`)
Robot: Thanks for your agree. What do you want to do now?
Player: I'm done.
(`Intention submit`)
Robot: Are you sure that you want to submit?
Player: Nope.
(`Intention disagree`)
Robot: Your submission has been canceled.

(`Condition 2:Only one location keyword is detected`)
Robot: Do you want to connect **Mount Bern** and **Mount Zurich**?
Player: No. I want to go to **Luzern**
Robot: Do you want to connect **Mount Bern** and **Mount Luzern**?
Player: Yes, please connect them.
(`A connection is made between Mount Bern and Mount Zurich`)
Robot: You have successfully connected **Bern** and **Luzern**. Do you agree with that?
Player: Okay!
(`Intention agree`)
Robot: Thanks for your agree. What do you want to do now?
Player: Remove everything.
(`Intention clear`)
Robot: You clear all. What do you want to do now?

(`Condition 3:Neither of location keywords is detected`)
Robot: Invalid locations. I heard ... Please repeat.

Basically, users will have five kinds of intentions in this process:

- `connect:Build a route between two mountains`
- `agree:Agree with the other's suggestion`
- `disagree:Disagree with the other's` suggestion
- `clear:Erase all the routes on the map`
- `submit:Submit and check if the current plan is the optimal solution`

Among the five intentions, we can further divide them into simple ones such as `agree, disagree, clear, submit`, as well as the complex intention `connect` which usually includes at least two keywords - name of `location 1` and `location 2`.

3 System Architecture

There are two ROS nodes implemented for the project, namely `Speech_recogniser` and `Itention_interpreter` [3].

Figure 2: ROS computation graph for intention detector

3.1 Speech_recogniser

The aim of this node is to do automatic speech recognition with Google Speech Recognition API(v2)[4], then publishing the recognized utterances at a ROS topic in the format of `String` [1]. In addition, it contains a feedback system indicating if the robot is listening to the user or not. The feedback system is developed in two versions: LED color feedback on Reachy, and sound effect indicator. A comparison of testing results without and with two different feedback systems will be illustrated in the later sections.

The `Speech_recogniser` node consists of the following methods:

- __init__(): Initialization of speech recognizer and microphone.
- intention_detection(): Main method to print prompt, start speech recognition, and report error.
- recognize_speech_from_mic(): Transcribe speech from recorded from 'microphone', give feedback before and after each listening period.

[3]Yiwen Ma. *ros2_intention_detector.* Github repo: `https://github.com/yma97/ros2_intention_detector`

[4]Google Speech Recognition API (v2), with parameters "url"=`http://www.google.com/speech-api/v2/recognize?`, "key"=None, "lang"="en-US", "client"="chromium"

3.2 Intention_interpreter

The node `Intention_interpreter` subscribes the node `Speech_recogniser`, receiving the transcript, detecting user's intention through keywords analysis, providing instructions or followup messages, and executing the corresponding actions in the JUSThink activity. In this script, a `Boolean` value `followup` is used as a flag to help with the smooth interaction.

The `Intention_interpreter` node consists of the following methods:

- __init__(): Initialization of the activity.
- listener_callback(msg): Be invoked when a new message is received.
- action_detection(intention_dic, wordList): Be called when followup is False. Check the five basic intentions and execute the action in the world accordingly.
  Available intentions: [connect], [clear], [submit], [agree], [disagree]
- followup_detection(intention_dic): Be called when followup is True. Intention is checked for the followup actions to confirm, re-guide user to right direction, and execute the action in the world accordingly.
  Available intentions: [submit-confirmation], [connect-suggestion]
- keyword_detection(wordList): Detect user's intention when speech recognition succeeds. Compare the words in the transcript with the keywords set.
- connect_location(wordList): Connect two locations in the activity.

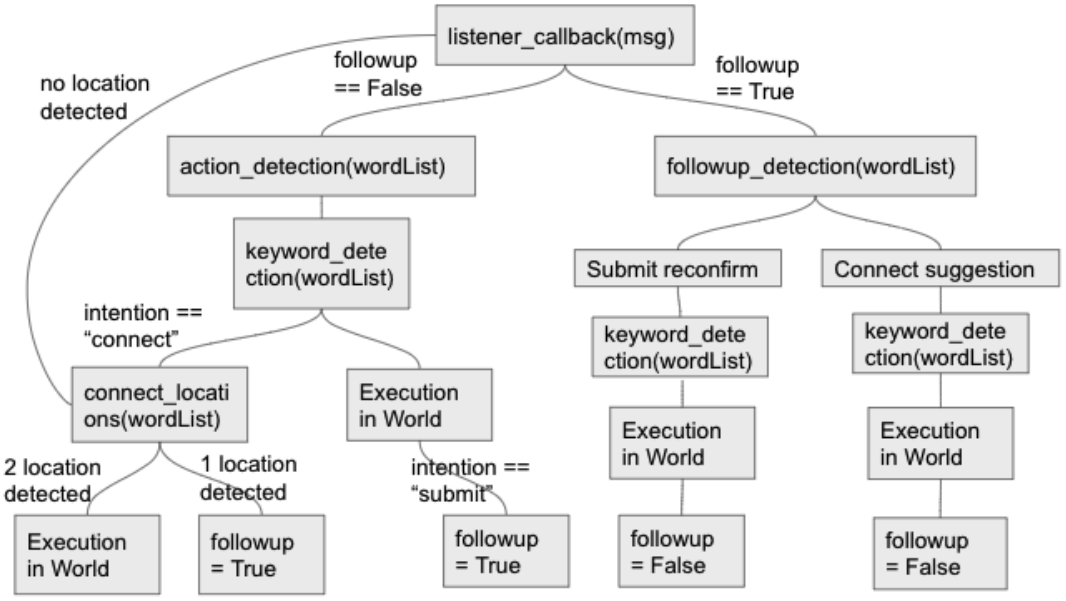

Figure 3: Flowchart of Intention Interpreter

The keyword list for each intention is shown as below:
keywords_connect: [‘connect’,‘kinect’,‘go’,‘from’,‘build’,‘bridge’,‘add’,‘another’,‘walk’, ‘building’,‘going’,‘put’,‘route’,‘train’,‘bridges’,‘connected’]
keywords_clearall: [‘clear’,‘delete’,‘remove’,‘clean’,‘erase’,‘empty’,‘cancel’,‘disconnect’]
keywords_submit: [‘submit’,‘done’,‘end’,‘finish’,‘terminate’,‘finished’]
keywords_agree: [‘yes’,‘yea’,‘okay’,‘agree’,‘ya’,‘like’,‘do’,‘good’,‘great’,‘okay’,‘ok’,‘fine’, ‘sure’,‘nevermind’,‘accept’,‘acceptable’,‘except’,‘smart’,‘accepted’,‘correct’]
keywords_disagree: [‘no’,‘nope’,‘not’,‘don’t’,‘disagree’,‘waste’,‘wasting’]
keywords_location: [‘montreux’,‘neuchatel’,‘basel’,‘interlaken’,‘bern’,‘zurich’,‘luzern’,‘lucerne’,‘zermatt’,‘st.gallen’,‘davos’]

4 Iterative Design

In this project, an iterative design has been used to continuously improve the smooth interaction between the user and the robot(JUSThink activity). After each round of validation, some modifications were made according to the issues observed during the test. In general, in addition to the survey, each validation round had four participants, two of whom had prior knowledge of the activity and two of whom were completely new. In this case, prior knowledge means that the user has participated in one or more previous rounds of experiments. All the participants are EPFL Master students, with an age range 22-25. All the rounds of validation were recorded and analyzed, with the first and third round of validation taking place on a laptop somewhere on campus with varying levels of background noise, and the second and fourth round of validation taking place on Reachy in a CHILI Lab meeting room with very low noise levels.

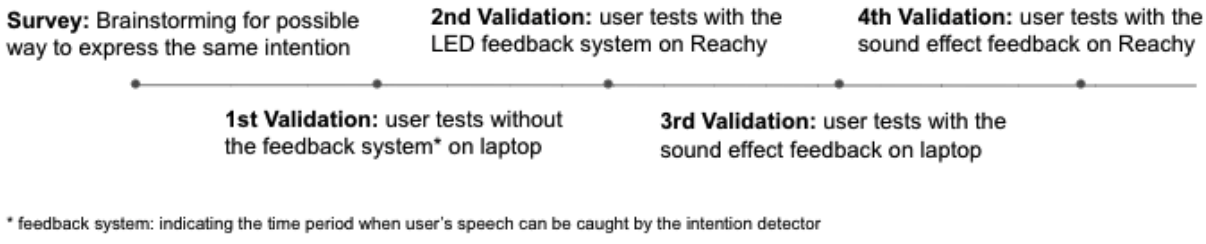

Figure 4: Scheme of the iterative design process

4.1 Survey

The initial interview aims to collect users’ thoughts and their possible way to express the same intention. Only the keyword detection without connecting to the activity had been implemented till the survey. With a microphone and a laptop running the speech recognition and intention interpreter nodes, six participants were given the description of the activity, pretended they were playing the activity, and tried as many times as they can to express the five basic intentions in different ways to the detector. The keyword lists were expanded according to the new discovery in each interview that were not caught

by the current system. In the table, the red color indicates the most frequent expressions for each intention.

connect	clear	submit	agree	disagree
"Connect xxx to xxx"	"Clear"	"I am done"	"I agree"	"I don't agree"
"I want to go from xxx to xxx"	"Please delete everything"	"Can you please submit"	"I want to do what you just said	"No"
"I want to connect xxx to xxx"	"Please remove everything"	"I want to submit"	"I like that suggestion"	"No, I disagree"
"Build the bridge from xxx to xxx."	"I want to clear all the path"	"I would like to end the map"	"Yes"	"I disagree with what you said. "
"From xxx to xxx."	"I want to delete all the path"	"I finish creating all the bridges"	"Let's do this"	"I don't agree about ur selection"
"I want to walk from ...."	"Clear everything"	"I want to terminate the construction	"It's a good idea"	"I think what you suggest is not good"
"building a bridge "	"Clean everything"	"Submit the route"	"That's a great idea"	"No, I don't want to do this action"
"Going from ...."	"Erase all the bridges"	"Submit"	"I like your option"	"I don't think it's a good idea"
"Go to"	"Empty the map"	"Stop it" (new for round 2)	"okay, it's fine"	"It's not a good idea"
"I want to put another bridge"	"I want to cancel my choice"		"ok"	"you are so stupid"
"I want to go right from Montreux"	"Clear all"		"sure"	"we are wasting money"
"Go from xxx to xxx"			"accept"	"Nope"
"add another route from xxx to xxx"				"Disagree"
"I want to take a train from xxx to xxx"				

Figure 5: Brainstorming result of the possible expressions for five basic intentions

4.2 Validation Setup & Changes

4.2.1 First Round Validation

In the first round of validation, participants played the JUSThink activity by working through a microphone plugged to a laptop at a distance of less than 5 centimeters. Participants could speak whenever they wanted without a feedback system indicating when the detector was listening.

Changes: Connect subscriber to the world activity; Add the log system printing detailed steps user go through; Expand keywords list according to the brainstorming result.

4.2.2 Second Round Validation

In a second round of validation, participants played the activity by working through a microphone on Reachy, about 30 to 40 centimeters away from the user. Reachy uses an LED feedback system, with green indicating that Reachy is listening and red indicating that it has stopped listening. Only when the LED lights are green can the participant's voice commands be captured for recognition.

Changes: Refactor code and change `connect` detection method from using connect keywords to location keywords; Set speech recognition listening period from automatic stop to max 10 seconds to solve stuck in listening issue; Add instruction for ambitious movement between locations without direct route; Implement the LED feedback system in detector.

4.2.3 Third Round Validation

In the third round of validation, participants played the activity by working through a microphone plugged to a laptop as in the first round, but with a audio feedback system.

When the "di-" sound effect appears, the detector starts listening. When a "di-di-di" sound effect appears, the detector stops listening. Only during the time period between the start and the end of the voice can the participant's voice commands be captured and recognized.

Changes: Add three keywords for `connect` and `locations` to solve the recognition issue; Set speech recognition listening period to max 8 seconds since usually a user's voice command is less than 8 seconds; Implement the sound feedback system in detector.

4.2.4 Fourth Round Validation

In the fourth round of validation, participants played the activity by working through a microphone on the Reachy as in the second round, but using the same sound feedback system as in the third round. The purpose of the validation is to test whether the presence of the robot would add any additional stress that would affect the test results.

Changes: Add two new keywords for `agree`; Implement the sound feedback system in detector on Reachy.

5 Results

5.1 Evaluation Metrics

Four main metrics are used to evaluate the performance of intent detection and interactions during the validation process:

Detection rate calculated by dividing **successfully detected intentions** by **total attempted intentions** reveals the accuracy of the detection result. Among them, total attempted intentions is calculated as the number of intentions that participants try to express through words, and successfully detected intentions refers to the number of intentions that are accurately detected and executed in world activities.

Average steps taken per intention calculates the average number of steps taken for all successfully detected intentions. The steps taken for each intention are counted as the number of utterances by the participant to express the same intention.

Average followups used per intention calculates the average number of followups used for all successfully detected intentions. The number of followups mainly came from intentions `connect` and `submit`.

Average time used per intention (seconds) records the average time it takes a participant to complete an action in the activity. The time used for completing an action is calculated from the end of the last execution in the activity until next execution.

5.2 Intention Detection Performance

5.2.1 First Round Validation

	Participant 1	**Participant 2**	**Participant 3**	**Participant 4**
Background noise level	Medium	Medium	High	Low
Mother tongue	French	French	French/English	Italian
Previous knowledge of the activity	Yes	No	No	Yes
Detection rate	66.67%	90.00%	100.00%	88.89%
Avg steps taken per intention	3.75	3.11	2.33	2.38
Avg followups used per intention	1.25	1.11	0.50	1.11
Avg time used per intention (seconds)	46.59	28.84	23.63	27.10

Table 1: Result of the First Round Validation

Observations: Participants received no feedback on when the system is listening or not. They could speak to the microphone any time, while the system may or may not be listening. Due to the interval between two listening cycles in which the Google API was called for speech to text recognition, this resulted in some voice commands not being captured by the detector. In addition, some participants changed their intentions after detecting some failed attempts, which was a major factor in reducing detection rates. Another factor was inaccurate speech recognition results, which could be due to high background noise level, the pronunciation variants from participants (especially for the mountain names that are partially due to accent but more than that), or issues in Google speech to text API.

5.2.2 Second Round Validation

	Participant 1	**Participant 2**	**Participant 3**	**Participant 4**
Background noise level	Low	Low	Low	Low
Mother tongue	English/Spanish	Arabic	Chinese	French
Previous knowledge of the activity	Yes	Yes	No	No
Detection rate	90.91%	87.50%	95.24%	80.00%
Avg steps taken per intention	2.70	2.29	2.25	3.13
Avg followups used per intention	0.90	0.86	0.95	1.13
Avg time used per intention (seconds)	35.61	41.26	30.81	48.28

Table 2: Result of the Second Round Validation

Observations: Firstly, participants felt more stressed when they were doing the activity with the LED feedback system, and they tended to give the shorter commands. For example, some of them used command “xxx to xxx” instead of “Go from xxx to xxx”. Secondly, participants had to look at both the main screen for activity state and the LED lights on Reachy. Sometimes they concentrated too much on the activity and forgot about the feedback system. Finally, there was no obvious improvement in speech recognition

accuracy even if the background noise level was low. The possible reason for that might come from the hardware difference (microphone and distance to user), as well as the different accent of the participants.

5.2.3 Third Round Validation

	Participant 1	**Participant 2**	**Participant 3**	**Participant 4**
Background noise level	Medium	Medium	Low	Low
Mother tongue	French	Italian	Chinese	Chinese
Previous knowledge of the activity	Yes	Yes	No	No
Detection rate	100.00%	87.50%	100.00%	77.78%
Avg steps taken per intention	2.11	2.00	2.09	3.14
Avg followups used per intention	1.00	0.43	1.00	0.57
Avg time used per intention (seconds)	19.62	25.21	28.41	36.80

Table 3: Result of the Third Round Validation

Observations: Participants in this round of validation knew exactly when their speech could be heard so they did not make useless attempts when the system was not listening. Plus, with the sound feedback system, they could keep their eyes on the screen all the time, but not as in the second round in which they need to look at both the robot and the screen.

5.2.4 Fourth Round Validation

	Participant 1	**Participant 2**	**Participant 3**	**Participant 4**
Background noise level	Medium	Medium	High	Low
Mother tongue	French	Chinese	Chinese	French
Previous knowledge of the activity	No	Yes	No	Yes
Detection rate	63.64%	100.00%	100.00%	100.00%
Avg steps taken per intention	2.50	2.69	2.50	2.11
Avg followups used per intention	1.33	0.89	1.17	0.89
Avg time used per intention (seconds)	36.76	30.84	32.79	22.60

Table 4: Result of the Fourth Round Validation

Observations: Even when the robot is present, participants could still focus their attention on the screen through a sound feedback system.

5.3 Comparison

5.3.1 Impact of the Feedback System

An improvement of performance comes from less steps taken and less average time used for one intention. As can be seen from Table 5, the number of average steps taken gradually

	Validation round 1	**Validation round 2**	**Validation round 3**	**Validation round 4**
Avg detection rate	93.06%	88.00%	91.00%	91.00%
Avg of avg steps taken per intention	2.75	2.59	2.34	2.44
Avg of avg followups used per intention	0.93	0.96	0.75	1.07
Avg of avg time used per intention (seconds)	29.85	38.99	27.51	30.75

Table 5: Overall Comparison of Results for Four Rounds Validation Using Average

	Validation round 1	**Validation round 2**	**Validation round 3**	**Validation round 4**
Median of avg steps taken per intention	4.375	4	5	4.25
Median of avg followups used per intention	1.75	1.25	1.375	2.5
Median of avg time used per intention (seconds)	41.71	56.45	32.14	52.83
Avg of avg steps taken per intention	4.60	3.91	2.85	4.13
Avg of avg followups used per intention	2	1.93	1.44	2.50
Avg of avg time used per intention (seconds)	45.93	58.35	38.52	53.35

Table 6: Comparison of Results for Intention "connect" Using Median and Average

	Validation round 1	**Validation round 2**	**Validation round 3**	**Validation round 4**
Median of avg steps taken per intention	1	1.125	1.625	1
Median of avg followups used per intention	0	0	0	0
Median of avg time used per intention (seconds)	14.51	16.97	12.66	13.59
Avg of avg steps taken per intention	1.21	1.72	1.84	1.48
Avg of avg followups used per intention	0.11	0.19	0.15	0.225
Avg of avg time used per intention (seconds)	15.42	24.11	16.73	17.42

Table 7: Comparison of Results for Intention "agree/disagree/clear/submit" Using Median and Average

decreased from the first round of validation without feedback system to the second round of validation with LED feedback system, and finally the performance was the best in the third round of validation with audio feedback system, since the user had the indicator to tell them when to speak so that they gave up useless attempts. According to Table 6 and Table 7, the improvement mainly came from intention `connect`, while there was no obvious improvement in the performance of simple intentions. Same for the average time used per intention, the third round validation used least time among all rounds. Thus, we can conclude that the feedback system can improve the intention detection performance, especially for the more complex intentions such as `connect`.

5.3.2 Difference between Visual and Audio Feedback System

By comparing the results of LED feedback system validation (round 2 validation) with sound feedback system validation (round 3 and 4 validation), the average steps taken and the averages time used per intention has largely decreased. It is worth noting that the second round of verification took the most time on average among the four rounds, as participants had to focus on both the robot's screen and the LED, which consumed a lot of time. Thus, we can conclude that the audio feedback system performed better than the visual one in this case.

5.3.3 Impact of the Presence of Robot

By comparing the results of the third and the forth round of validation, in which both rounds used the audio feedback system but one without the robot and one with the robot, it can be seen that the performance was slightly decreased due to hardware differences, but still better than the results with the visual feedback system. By interviewing participants, especially two users who participated in the third round of validation, they did not feel any difference or significant stress because they were not actually interacting with the robot. We can conclude that the presence of robots is not a source of stress. However, some participants said they still enjoyed owning the robot because it made the activity more engaging. They expected the robot to do some movements to add more fun, so the user can get more involved in the activity.

Acknowledgement

I would first like to acknowledge the host lab for the project Computer-Human Interaction in Learning and Instruction (CHILI) and the project supervisors Barbara Bruno and Utku Norman. who provided me the advice and guidance through the whole project.

In addition, I would like to thank my friends, Jan, Marie, Roberto, Sarah, Nelson, Jouanna, Catalina, Eileen, Raphael, Weiyi, Jingran, Alexis, and Jade, who were willing to take the time to participate in the experiment and gave me a lot of meaningful feedback and suggestions.

References

[1] David Amos. *The Ultimate Guide To Speech Recognition With Python*. URL: `https://realpython.com/python-speech-recognition/`.

[2] Utku Norman, Babara Bruno, and Pierre Dillenbourg. *Mutual Modelling Ability for a Humanoid Robot: How can it improve my learning as we solve a problem together?* in Robots for Learning Workshop in 16th annual IEEE/ACM Conference on Human-Robot Interaction (HRI 2021).