Graphical User Interface for Adaptive Human-Robot Interaction Design in Educational Activities Creation

Daniel Tozadore¹ and Roseli Romero¹

¹Instituto de Ciências Matemáticas e de Computação Universidade de São Paulo (USP) Av. Trabalhador Sancarlense, 400 – São Carlos – SP – Brasil

tozadore@usp.br, rafrance@icmc.usp.br

Abstract. Robots and interactive devices are being successfully applied in education to increase the students cognitive experience and, consequently, their learning rate. However, there is a lack of social robots with adaptive skills that can be easily programmed. In this paper, a Graphical User Interface to create, perform and evaluate educational activities with autonomous robots is presented. The interface runs in a system that provides adaptation and personalization through open source techniques for autonomous analysis and recognition of speech, focus deviation and facial emotion. During the activity execution, the adaptive algorithm detects student's body signals and verbal responses to adapt the addressed content to harder or easier questions. After creating and running an activity, all sessions' evaluation and information can be accessed for visual analysis, as well as students' preferences throughout the interaction.

1. Introduction

Smartphones and tablets are more common in electronic learning due to their low cost and users familiarization. Conversely, interactive robots are far from achieving popularization due to the high costs and lack of people knowledge about design and programming robots. Especially regarding its application in the educational field, the lack of training of teachers and their inclusion in the robot's design are pointed as one important concern in a worldwide scenario [Johal et al. 2018].

Researches involving social robots to achieve success in tasks with humans are placed in the Human-Robot Interaction (HRI) field. Social robots increase people's curiosity and motivation in social and intellectual activities because machines are commonly associated with factories or physical tasks. They are being used in the educational field as an option to perform practical exercises by telling stories or playing quizzes. However, after the robot loses its novelty and becomes a regular tool for teaching, a decrease in the students motivation and attention span is observed. A robot behavior with personalized conversations and content difficult adaptation may hold the students interest in the pedagogical interactions for a longer period of time. It was noted a lack in the literature about social robotic systems that allow non-programming people to design and execute HRI activities.

In this paper, a Graphical User Interface (GUI) to create, perform and evaluate educational activities is proposed. This solution allows any person to design HRI educational activities in an intuitive way. The GUI works handling pedagogical contents and setting up a robotic architecture to autonomously interact with students, approaching

these contents in individual sessions. The whole application is projected to be used in several matters during the school year. Thus, it also stimulates the teachers' creativity in planning new activities.

The robotic architecture is a cognitive adaptive system [Tozadore et al. 2017] that encapsulates a module based implementation to run over the robot's sensors and actuators. The interaction flow overview provided by this architecture is shown in Figure 1. The person who designs the interactions (the designer) can program the activities in the system's GUI. The designer is represented as the teacher in this illustration since it is expected that the teachers mainly perform this role. During the activities execution, the system controls the robot to autonomously interact an adapting to the student.

The system interface with the student is a NAO robot, from Softbank robotics ¹. NAO is a 60 cm tall humanoid robot that has visual and sound resources (among others) designed to interact with humans in general purposes. Its application in educational tasks has been explored in several studies and shown well accepted.



Figure 1. Architecture scheme overview.

Finally, this paper is focused on discussing the GUI solutions for computational modeling of the activities. The next sections are organized as it follows. Related works are presented in Section 2. Pedagogical methodology is described in Section 3. GUI solutions are presented in Section 4. Final considerations and future works are summarized in Section 5.

¹www.softbankrobotics.com

2. Related Work

Mountable kits as Lego Mindstorms² and Pete³ offer intuitive graphical interfaces to the users and they are the most application used as educational robotics [Benitti 2012]. However, their usage is limited to the STEM field⁴ domain. After programming, the robots execute the student's code and there is no interaction between them. Hence, they are perceived more than a learning tool than an agent that can play an active role in the students cognitive process. On the other hand, social robots are not programmed by the students, but they are more supportive during the activities and capable to cover topics for more areas [Belpaeme et al. 2017].

Intelligent tutoring systems (ITS) are systems which adapt to the student necessities. They are commonly (and since a long time ago) used in electronic learning [Murray 1999]. Their application increases the students learning experience and provides better consequences in the content fixation. In general, the ITS also provide to the teachers an easy approach to plan the activity to be performed through GUI. Nonetheless, embodied systems provide a more complete experience than virtual learning environment [Platz et al. 2018]. These contributions goes from pronunciation skills [Spaulding et al. 2018] to Mathematics [Clabaugh et al. 2017].

Another advantage in social robots is to aggregate human relation characteristics to the process. For instance, personalization in social interactions have shown themselves an alternative in keeping the engagement of the users (the persons which are using the application). In the same way, by simulating the feeling of rapport building between robot and user, is possible to explore social techniques to enhance the results in the performed activity [Lucas et al. 2018]. Researches with interactive robotics architectures suggested significant improvement in multimodal interaction, achieved with a simple file management solution [Cortellessa et al. 2018]. Multimodal emotional robots are playing an essential role when interacting with children. Results shown that the more human communication resources are demonstrated by the robot, the more the children's confidence in those systems increases [Kessous et al. 2010].

However, little is known about how much these works with robots collaborate to place the teacher in comfortable and active role in planning the activity. This is pointed as an issue to be enhanced in the area [Johal et al. 2018].

3. Pedagogical Model

The pedagogical model is based on constructivism, as the educational robotic in general [Kafai et al. 2017]. The tutor (in this case the NAO robot) plays the main role in the interaction and measure by questions how much the student is rightly constructing its knowledge. The questions can consider objects to be handled by the robot and stories of daily problems to be addressed, characterizing the constructivism.

The ideal scenario is to use this system as practical exercise fixation after regular classes about the topics' concepts. Each meeting between robot and student is called a session. During a session, the robot presents a concept to the student and evaluate if he/she

²www.lego.com/en-us/mindstorms

³www.pete.com.br

⁴Science, Technology, Engineer and Mathematics

had understood this explanation by asking questions. For that, every topic that aims to be addressed needs to be registered in the system. The topics have concepts - which is the topic explanation - and as many questions regarding this concept as the designer want to be approached. It is mandatory to divide the questions into five levels of difficulty and at least one question per level in order to guarantee content adaptation. Each session follows the same scheme divided in three phases: *Welcome Dialogue, Content Approaching* and *Closure Dialogue*.

In the first meet, the *Welcome Dialogue* phase will recognize the student's face and insert it into the users database. If the student is already registered, all his/her information is recovered to be used in the following conversations. Content adaptation is mapped in the *Content Approaching* phase. The topics' concept is discussed followed by a random number of questions defined by the designer. The questions are chosen in the difficulty level set by the adaptive function in an instant t of the interaction. The t instant is considered the time to realize a question and it may vary from one question to another. Finally, in the *Closure Dialogue*, the robot makes a content summary to the student about what was approached in the session. As a feedback, some student's skills, such as average time to respond and correct answers rate, are reported and discussed by the robot. Additionally, some tips about how to improve or keep these skills for the next sessions are presented.

The computational mapping of this methodology is achieved by coding, which is a very unclear process for those whose do not have programming knowledge. To work around this problem, a graphical interface is presented in the next sections, aiming to approximate regular teachers to robotic solutions for education. The challenge in modeling a content following the presented guideline can be seen as a secondary contribution since it also stimulates the teachers' creativity and pedagogical skills.

4. The Interface

The proposed GUI was implemented to operate over a cognitive adaptive system. Its code can be accessed in the project github website⁵. Its goal is to allow regular teachers, or any person who does not have programming skills, to easily manage the activities and the system's settings. The adaptive function's goal is to make the interactions as attractive as possible to the student, based on the indicators read along the session. The designer only needs to set up some variables in the GUI (detailed in Subsection 4.4).

The framework PyQt4⁶ was used to the GUI's development. It facilitates the integration with the architecture that are also implemented in Python language. The software is preferably configured for 14 inches monitors and runs in the same window all the time. The system functionalities are handle in the bottom section of this window by changing the tabs, as detailed in the next subsections. For detailed technical development of each module, please check [Tozadore et al. 2017].

In Figure 2, it is shown the main window discriminated by sections. The activity summary (1) - highlighted in green - is fixed and the designer can use the bottom section (2) - highlighted in green - to configure the activity by each functionality. The activity

⁵https://github.com/LAR-Educational/Architecture_v2_0/tree/master/ Arch_2_1/GUI

⁶https://www.riverbankcomputing.com/software/pyqt/download

©⊜ © Menu Users Help						
New Activity	Activity	Spatial Geometry				
Load Activity	Description	Activity addressing spatial geometric content.				
	Path	/Activities/Spatial_Geo	Activities/Spatial_Geo			
Edit	Created in	30/06/18 - Created by admi	30/06/18 💌 Created by admin			
Exit	Last Edit	30/06/18 🕽 Edited by admin				
	Lar Engli	nguage sh t 95 t %	eech Recognizer pogle : 90 : %	Distance Threshold	Leventchein Distance Longest 🛟	
Default Question	Default Questions Dialogue Files Answers Keywords Analysis Insert Ouestion Clear Table Save Load File					
Questions			Answers			6
Lets finish the session How are you? What day is today? What is your's creator names? Hi		Do you want to finish the session I'm very happy to talk to you. Today is friday. Daniel Hello there.	1?			

Figure 2. Main Window divided in: 1 - the activity summary (fixed); and 2 - the functionalities tab.

summary (1) is composed by the main menu buttons in the left, the activity properties in the middle and a activity picture in the right.

Designers need to be registered in the system and sign in into the software for issues tracking. All their action are registered and can be accessed lately. Along the software usage, all information is stored in files that can be reused in other activities or shared through storage devices and networks.

The system is divided into the following functionalities, coded in the corresponding tabs. Some tabs are briefly or not commented in this paper due to size constraint.

4.1. Content

The Content tab allows to create and manage the topics according to the adopted Pedagogical Model (Section 3). The activities can have as many topics as the designer wants and it is also possible to import topics from other activities. The content is defined by topics the designer aims to address during the content approaching phase. As can be seeing in Figure 3, topics are easily inserted by clicking on the "New Subject" button. Already registered topics are handled in the corresponding combo box.

Topics' concepts are inserted or displayed in the concept field. The concept is the topic definition and it is exactly what the robot will explain about this topic to the student. Due to resources limitations, the robot exclusively counts on verbal explanation to address the contents. However, future works include adding a visual display, such as tables or screens, to increase the explanation experience. In the bottom of the content tab, the designer registers the questions of every difficulty level. It is mandatory to register one question of each level and desirable to have as many as possible.

4.2. Dialogue

In the Dialogue tab is possible to configure the system variables that will control the verbal interaction. In this project, all dialogues and expected answers must be registered

stateget soliterite th	sion Adaption Knowledge System Reports Students Eva	luations PLAN and RUN!		
Subject: Face, Edges and Vertices				
	Co	ncept		
A vertice is a corner. An edge is a line segment that joins two vertices. It is also the faces' encounter. A face is an individual surface delimited by the edges. Dificulty Level 4 - Hard : Insert Question Delete Question				
Dificulty Level 4 - Har	d : Insert Question Delete Question		Clear Table	
Dificulty Level 4 - Har Dificulty	d :) [Insert Question] Delete Question]	Expected Answer	Clear Table	
Dificulty Level 4 - Har Dificulty 1 1 - Very Easy	d : Insert Question Delete Question V Question Whait is a face?	Expected Answer A face is an individual surface	Clear Table	
Dificulty Level 4 - Har Dificulty 1 1 - Very Easy 2 1 - Very Easy	rd : Insert Question Delete Question • Question Whait is a face? What is a vertice?	Expected Answer A face is an individual surface A vertice is a corner	Clear Table	
Dificulty Level 4 - Har Dificulty 1 1 - Very Easy 2 1 - Very Easy 3 2 - Easy	d : Insert Question Delete Question Vestion Question Whait is a face? What is a vertice? How many faces does have a cube?	Expected Answer A face is an individual surface A vertice is a corner 6	Clear Table	
Dificulty Level 4 - Har Dificulty 1 - Very Easy 2 - Very Easy 3 2 - Easy 4 3 - Medium	rd : Insert Question Delete Question V Question Whait is a face? What is a vertice? How many faces does have a cube? How many vertices does have a pentagon?	Expected Answer A face is an individual surface A vertice is a corner 6 5	Clear Table	
Dificulty Level 4 - Har Dificulty 1 1 - Very Easy 2 - Very Easy 3 2 - Easy 4 3 - Medium 5 4 - Hard	d : Insert Question Delete Question V Question Whait is a face? What is a vertice? How many Yaces does have a cube? How many vertices does have a pentagon? What is the vertice and edge relation?	Expected Answer A face is an individual surface A vertice is a corner 6 5 An edge is a line segment that joins two vertices	Clear Table	
Dificulty Level 4 - Har Dificulty 1 1 - Very Easy 2 1 - Very Easy 3 2 - Easy 4 3 - Medium 5 4 - Hard 6 5 - Very Hard	d : Insert Question Delete Question V Question Whait is a face? What is a vertice? How many faces does have a cube? How many vertices does have a jentagon? What is the vertice and edge relation? How many faces does have a figure with 4 vertices?	Expected Answer A face is an individual surface A vertice is a corner 6 5 An edge is a line segment that joins two vertices 4	Clear Table	

Figure 3. Content tab with 3D geometry example.

in the system. This is mandatory because the system uses these sentences to compare their distance to the student answers and, based on a threshold set by the designer, judge if the answer is right or wrong. The chosen comparison algorithm is the Levenshtein distance, which is very used in Natural Language Processing techniques and DNA comparison.

In the Dialogue functionality tab (Figure 2) is possible to set the components: Language (English or Portuguese), Volume(0 to 100 %), Speech Recognition Method (Google Recognition or NAO's Default), Robot's Speech Speed (0 to 100 %), Levenshtein Distance Threshold (0 to 1) and Levenshtein Distance Method (Longest or Shortest). The bottom section is a frame with tabs responsible for control the tables: "Default Questions" to register possible questions that can be made to the robot in any part of the interaction; "Conversation Set Up" to write Welcome and Good Bye dialogues; and "Answers Keyword Analysis" to set students vocabulary for affirmation, negation and doubt.

4.3. Vision

The Vision module is optional in the activities. It is responsible for recognize and classify the objects using Machine Learning methods. The implemented methods are Multilayer Perceptron (MLP) Networks, K-Nearest Neighbor (KNN), Support Vector Machines (SVM) and Convolutional Neural Networks (CNN). Objects database creation or reuse is required for every activity that uses the vision module.

4.4. Adaption

The Adaptive module aims to change the robot's behavior according to the observed student's indicators, expressed by body language and verbal answers. These indicators were divided into three main groups regarding the measures of Attention, Communication and Learning.

The objective measures of each group are summarized in Table 1. The groups are shown in the first line (with the corresponding denotation function in parentheses) follow by the respective indicators in their columns. They are: Face gaze for the Attention; users' Emotions for Communication; and Right/Wrong answer and Time to answer the proposed exercises for the Learning group. The average of the objective measures of each group result in a final major value of the class, named as α to Attention, β to Communication and γ to Learning.

VII Congresso Brasileiro de Informática na Educação (CBIE 2018) Anais do XXIX Simpósio Brasileiro de Informática na Educação (SBIE 2018)

Table 1. Objective measures by group.				
Attention (α)	Communication (β)	Task (γ)		
Face gaze	Number of Words	Right/Wrong answer		
	Emotions	Time to answer		

Each major value is calculated by their respective measures tolerance normalization between 0 and 1. The vectors correspond to the activation value for each weight in the F_{Adp} calculation in Equation 1. The maximum limit for the major values is 1 and when applied to the F_{Adp} equation, they will fully activate their corresponding weight from the class. 0 means that there was no detection of this class activity in the instant t. Thus, this value corresponding weight will not contribute to the F_{Adp} in the instant t + 1. In other words, the classes major values α , β and γ mean how much their respective class is being critical (from 0 for none to 1 for maximum) in the specific instant t, whereas the weights potentialize how much their respective class is contributing to the F_{Adp} calculation and their values are the same all along the session. It is worth to notice that the sum of the weights should not be greater than one in order to fit the interval from 0 to 1 of the adaptive function.

The adapted robot's behavior, denoted as Ψ , is an iterative function calculated by its last value added by the function F_{Adp} , as shown in Equation (1):

$$F_{Adp}(t) = w_{\alpha} * \alpha_t + w_{\beta} * \beta_t + w_{\gamma} * \gamma_t \tag{1}$$

where t is the instant in which the robot is approaching one question of a topic. The F_{Adp} is a function to adapt the resulting robot's behavior Ψ , trying to optimize the interaction engagement and learning rate. The parameters α , β and γ are the group activation function outputs and the w_{α} , w_{β} and w_{γ} are the corresponding weights, set by the designer before the session starts. In Equation (2), it is shown the final calculation of the robot's behavior state in the instant t. The $\Psi_o = 3$ state guarantees the system starts in neutral behavior.

$$\Psi_{t} = \begin{cases} 3, & \text{if } t = 0\\ \Psi_{t-1} + 1, & \text{if } F_{Adp}(t) \ge 0, 66\\ \Psi_{t-1} + 0, & \text{if } 0, 66 < F_{Adp}(t) < 0, 33\\ \Psi_{t-1} - 1, & \text{if } F_{Adp}(t) \le 0, 33 \end{cases} , t \in \mathbb{N}$$

$$(2)$$

The possible robot's behaviors Ψ is summarized in Table 2, which are in a discrete and incremental scale from 1 to 5, in which 1 means very easy content questions and 5 means very hard content questions, varying the level of gestural and personalized conversation interaction.

Regarding the adaptive tab in the GUI, the values α , β and γ are set in the corresponding section as can be seeing in Figure 4. By default, the weights are configured as $w_{\alpha} = 0.3, w_{\beta} = 0.2$ and $w_{\gamma} = 0.5$, which means that the adaptation function gives more importance to the learning rate vector.

4.5. Personalization and Knowledge Database

System databases are simply handled by files. There are two main databases: Students database and the system Knowledge database. Students database stores personal informa-

Table 2.	Robot behavior	states	and the	corresponding	question	difficulty	and
interactiv	ity level for eacl	۱ state.					

Ψ_t	Question Difficulty	Personal conversation and gestural interaction
1	Very Easy	None
2	Easy	Low
3	Medium	Medium
4	Hard	High
5	Very Hard	Low



Figure 4. Adaptive tab view.

tion, as students' name, last name, birthday date, school year, and 8 personal preferences as sport, dance, team, music, toy, hobby, game and food. The system searches in the Knowledge database about the student choices definition and uses it in sporadically talks, aiming to simulate long-term relations. These talks are triggered by the adaptive module if it detects signs of students low engagement. Similarly, in the system Knowledge database is recorded all the nouns definitions manually inserted or automatic searched by the system on the internet. It is possible to insert content into the system through the Knowledge tab or the system can search by them during the interaction. In case of new entries, the system searches in the Wikipedia website ⁷ through a python API. Small text analysis is employed to extract the noun abstract. Personal database stores information created about the robot "personal life" is also stored, such as how old he/she are, how many brothers and sisters he/she have, what is his/her name and so on. Previous studies showed significantly higher rapport building when humanizing the robot.

4.5.1. Evaluation

Once the sessions are individual, it can be searched by the student that realized the activity in the evaluation database and this activity evaluation summary will be shown in the Evaluation Tab. The evaluation summary shows the Student Name, Execution Date, su-

⁷www.wikipedia.com

pervisor (the designer that performed this activity), the time it was executed and a student picture at the section top. In the section's bottom, a multi-tab frame shows the sessions evaluation overview, a time line session evolution, the adaption system metrics and some pictures taken by the robot during the session. The most important to the teachers is the "Evaluation Overview" tab that displays the session duration time, the student correct and wrong answers, the time to answer rate, the time spent in each topic, the questions of each difficulty level chosen by the system, the adaptive measures and evolution, some interaction pictures and graphical analysis of the student performance.

4.6. Plan and Run

Finally, after setting the described configurations, the Plan and Run tab (Figure 5) provides the session high-level scheme to the designer. In this tab, the robot IP address and the robot communication port is required to initiate the session. The "Choose" button in the "Subjects Order" frame opens a window to choose the order that the topics will be approached and the number of questions for each topic. After setting these mandatory settings, the "Start" button is enabled. By starting the session, some variables that change over time are tracked in the bottom left section of the window in corresponding frames. When the session is started, the "Start" button changes to "Stop" button. The middle section of the screen displays the robot's camera image and the screen in the right section shows a terminal for monitoring system logs in execution time.



Figure 5. Plan and Run tab: The monitored variables in the left, the robot view in the middle and the terminal in the right.

5. Conclusion

In this paper, it was proposed a Graphical User Interface for content management. It runs over a robotic architecture which approaches registered contents. By following the adopted methodology, it is possible to set up the robotic system to interact in educational activities. The proposed GUI is an alternative to work around the problem of lack of familiarization with social robots by non-programming people. Although the architecture is initially designed to work only with NAO robots, it is being extended to work with other robots and interactive devices as well.

Acknowledgment

This study was financed by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001 and the Conselho Nacional de Desen-

volvimento Científico e Tecnológico - Brasil (CNPq).

References

- Belpaeme, T., Vogt, P., van den Berghe, R., Bergmann, K., Göksun, T., de Haas, M., Kanero, J., Kennedy, J., Küntay, A. C., Oudgenoeg-Paz, O., et al. (2017). Guidelines for designing social robots as second language tutors. *International Journal of Social Robotics*, 3:325–341.
- Benitti, F. B. V. (2012). Exploring the educational potential of robotics in schools: A systematic review. *Computers & Education*, 58(3):978–988.
- Clabaugh, C., Tsiakas, K., and Mataric, M. (2017). Predicting preschool mathematics performance of children with a socially assistive robot tutor. In *Proceedings of the Synergies between Learning and Interaction Workshop*@ *IROS, Vancouver, BC, Canada*, pages 24–28.
- Cortellessa, G., Fracasso, F., Sorrentino, A., Orlandini, A., Bernardi, G., Coraci, L., De Benedictis, R., and Cesta, A. (2018). Robin, a telepresence robot to support older users monitoring and social inclusion: Development and evaluation. *Telemedicine and e-Health*, 24(2):145–154.
- Johal, W., Castellano, G., Tanaka, F., and Okita, S. (2018). Robots for learning. *International Journal of Social Robotics*, pages 293–294.
- Kafai, Y., Sawyer, C. I. R., Papert, S., Harel, S. C. I. I., Papert, S., Duval, E., et al. (2017). Technology and theories of learning. *Technology Enhanced Learning: Research Themes*, 17(1):169.
- Kessous, L., Castellano, G., and Caridakis, G. (2010). Multimodal emotion recognition in speech-based interaction using facial expression, body gesture and acoustic analysis. *Journal on Multimodal User Interfaces*, 3(1-2):33–48.
- Lucas, G. M., Boberg, J., Traum, D., Artstein, R., Gratch, J., Gainer, A., Johnson, E., Leuski, A., and Nakano, M. (2018). Getting to know each other: The role of social dialogue in recovery from errors in social robots. In *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, pages 344–351. ACM.
- Murray, T. (1999). Authoring intelligent tutoring systems: An analysis of the state of the art. *International Journal of Artificial Intelligence in Education (IJAIED)*, 10:98–129.
- Platz, M., Krieger, M., Niehaus, E., and Winter, K. (2018). Suggestion of an E-proof Environment in Mathematics Education. Springer International Publishing, Cham.
- Spaulding, S., Chen, H., Ali, S., Kulinski, M., and Breazeal, C. (2018). A social robot system for modeling children's word pronunciation: Socially interactive agents track. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, pages 1658–1666. International Foundation for Autonomous Agents and Multiagent Systems.
- Tozadore, D., Pinto, A., Valentini, M. C. J., Zavarizz, R., Rodrigues, V., Vedrameto, F., and Romero, R. (2017). Project r-castle: Robotic-cognitive adaptive system for teaching and learning. *Accepted in IEEE Trans. on Cognitive and Developmental Systems*.