LEARNING FROM WIZARD-OF-OZ USING

DYNAMIC USER MODELING

by

TASNIM INAYAT MAKADA

Presented to the Faculty of the Graduate School of

The University of Texas at Arlington in Partial Fulfillment

of the Requirements

for the Degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE

THE UNIVERSITY OF TEXAS AT ARLINGTON

May 2017

Acknowledgements

I would like to thank Dr. Filia Makedon for being a constant source of motivation and support and for providing me the opportunity to work with her and for supervising my research and thesis. I also want to thank all members and researchers at the Heracleia Human-Centered Laboratory for being supportive throughout the process.

I would also like to thank my committee members, Dr. Vassilis Athitsos for his guidance and Dr. Farhad Kamangar for his feedback which always helped me improve.

Additionally I would like to mention Konstantinos Tsiakas, he is a Ph.D. candidate at the Heracleia Human-Centered Laboratory, and his input in this research and thesis has been invaluable. It was a learning experience working with him and without him this work would not have been possible and I would like to thank him for it.

I would also like to thank Heracleia Human-Centered Laboratory for the infrastructure provided to me. All the instruments used for this thesis were provided by the laboratory.

And finally, I would like to thank my parents, to whom I will be forever indebted for all their support, motivation and trust in me.

.

April 20, 2017

Abstract

LEARNING FROM WIZARD-OF-OZ USING

DYNAMIC USER MODELING

Tasnim Inayat Makada, MS

The University of Texas at Arlington, 2017

Filia Makedon

Socially assistive robotics (SAR) is a field of study that combines assistive robotics with socially interactive robotics where the goal of the robot is to provide assistance to human users through social interaction [1]. The effectiveness of a SAR system basically depends on the user's engagement in the interaction and the level of autonomy obtained by the system such that it requires no human intervention. The focus of this thesis is to build a SAR system that progressively learns to make autonomous decisions in an online manner, based on human input. An expert/therapist provides guidance to the system during the interaction and learns progressively the therapist's training strategy. This approach is also known as Learning from the Wizard.

In the field of human–computer interaction, a Wizard of Oz experiment is a research experiment in which subjects interact with a computer system that subjects believe to be autonomous, but which is actually being operated or partially operated by an unseen human being. The user in this case, is interacting with a robot and performing a training task, while having no knowledge of the expert/therapist's involvement in it. We developed a Wizard Interface, which provides the therapist with a visualization of the learning system and information about the training session, based on which they can

modify the action selection mechanism. A main module of the system is the user modeling module. A user model is the collection and categorization of personal data associated with a specific user. Dynamic user models allow a more up to date representation of users. Changes in their learning progress or interactions with the system are noticed and influence the user models. The models can thus be updated and take the current needs and goals of the users into account. Dynamic user modeling allows the system to learn from updated models of the user based on their performance in the current task. In our case, the tasks performed by the user are memory retention tasks, in which the user is given a sequence of characters to remember and repeat in the same order. The difficulty level of the task is dependent on the length of the sequence that the user is asked to remember. To obtain maximum user engagement the task difficulty has to be increased/decreased appropriately with time. Using the user's performance in each task and the dynamic use model created, a neural network is trained until the system learns to make autonomous decisions, and would require minimal intervention from the expert/therapist. This system intends to greatly reduce the therapist/experts workload from therapy sessions and also create a SAR interaction that the user feels engaged in.

Table of Contents

List of Illustrations

List of Tables

Chapter 1

Introduction

Socially assistive robotics (SAR) is a field of study that combines assistive robotics with socially interactive robotics where the goal of the robot is to provide assistance to human users through social interaction [1]. Socially assistive robotics can be applied in a multitude of tasks like physical therapy, speech, gestures, memory-related, as well as tutoring. Its application in varied tasks also ensures a varied audience which can range from elderly, individuals with physical impairments, individuals with cognitive disorders as well as students. The effectiveness of a SAR system basically depends on the user's engagement in the interaction and the level of autonomy obtained by the system such that it requires no human intervention. The focus of this thesis is to build a SAR system that progressively learns to make autonomous decisions in an online manner, based on human input. An expert/therapist provides guidance to the system during the interaction and learns progressively from the therapist's training strategy.

One of the ways to provide a robot-based system the ability to learn is by using machine learning. Machine learning has already been used in assistive systems for stroke rehabilitation and autism, and thus holds promise in robot assisted training systems as well. Our motivation is to combine methodologies and approaches of Machine Learning and Human-Robot Interaction (HRI) to make an interface that can be used as a SAR system for any interactive exercise between the human and the robot. The exercise here is a working memory task in which an individual has to remember and repeat a given sequence of items (e.g., letters, names, colors, etc.). In our case, the robot announces sequences of letters of the alphabet. The difficulty of the task can be increased by increasing the length of the sequence. The decision of whether to increase, decrease or maintain the difficulty to extract maximum user engagement in the process is the challenge which we try to overcome using machine learning during the interaction. Machine learning algorithms come with their own practical issues, like neural networks require large datasets to train the network, whereas, Reinforcement Learning uses extensive and costly exploration to gather data points.

The aim of this research is to enable a therapist to train the system to provide a personalized training session. And for this purpose we propose to use a neural network as a learning mechanism that learns online through therapist guidance, and also employ dynamic user modeling to aid in the learning task, so that the learning task can be less expensive in terms of data requirement. All users ('therapists') will go through a training round of the tasks they are supposed to perform, this achieves two goals, firstly, the user is trained on how to interact with the robot and what actions to perform, and secondly the data collected for the user can be used for modeling and clustering which will help the neural network predict better actions based on the user's engagement and concentration. At this point, if the therapist feels that the actions suggested by the neural network are not accurate, they can change the actual action performed and the neural network now learns from this feedback about creating a more personalized interaction for the current user. We describe our case study and we evaluate our system using HCI evaluation metrics, as task performance, user engagement and therapist 'workload', by combining both subjective (user survey) and objective (interaction) data.

Chapter 2

Related Work

Thorough research was done before the taking on the task of designing a socially interactive robotics [1] task. And preliminary work was performed in order to figure out which sequence learning task works best for the users. The inspiration for the framework to build a socially assistive robotics' task was from [8], where the framework was designed to engage children; a robot was used to communicate with the children with a neural network to help it make decisions while the children performed the task of classifying emojis displayed on a screen in front of them. But for testing the system the authors designed a child model that would behave like a child would. We wanted to gain the advantage of using real users and hence, build upon their framework to work with real users. Also, the task was simplistic due to interaction with children, while interaction with adults gives us a bigger scope to the type of task we can design.

Since our interaction involves real users, it is not possible to have a deep learning curve for the action selection network, as its earlier interactions with the user in such a case might actually harm the user rather than help them. Hence, we used [4] for inspiration in creating a human-guided system using the Wizard-of-Oz technique. In this research, a human 'behind the veil' guided the actions of the robot while the user interacting with the robot has no clue that such a 'wizard' or expert exists. This would require greater involvement in the earlier stages of the interaction where the system is in a learning phase and the 'wizard' can decide when the system is ready to work autonomously. This ensures that the user does not, at any point, be harmed by the system's decisions. And also the expert would only required to be involved in the initial phases of the task, but will experience a reduced workload with time.

Now that a decision making ‘wizard’ is involved in the task, it is important to provide this expert with all the information that might be required to make a proper choice about the next action. [5] provided a solution as to how one can use a dynamic user model, for the purpose of updating a user’s knowledge after each interaction. Their research focused on classifying a user as a novice or expert based on the words used by them in the interaction. Since the known words have to be appended with each interaction, it was important that a dynamic user model be created to update this information. In our case we need to update the user’s performance at the end of each round of the sequence learning task in order to decide which difficulty level of the task should be provided to them in the next round, hence we create a dynamic model for the user based on their performance in all the rounds for the current session. This information can help the ‘wizard’ or expert to determine based on the performance throughout about which action can be taken next round. The work done in [10] also worked as inspiration to determine how supervised learning in an autonomous system can be designed. We designed a system that learns online, allows the ‘expert’ to intervene and change the decision, but without intervention will still continue to work autonomously.

After having most components of the architecture in place, it was important to determine the sequence learning task that would be performed by the user such that it can be useful to the user in cognitive tasks that they might perform on a daily basis. Work was already done in the Heracleia Human-Centered Laboratory for such a cognitive task for people with dementia. [7] worked as an inspiration in designing three cognitive tasks for the preliminary work we did for the research. These three tasks involved remembering a sequence of characters and repeating them either on buttons or using speech, and to get the middle element of the sequence. During all the three tasks, brain waves and EEG for the users was recorded for each session. All the information rich data was published in [3] along with some analysis performed on it.

The preliminary data analysis helped us determine that users were most concentrated on the task which involved buttons, and least concentrated in the one which involved recollecting the middle element. This streamlined the task we used for the current system, and only the task with buttons was used. All these research work greatly helped us design the architecture in the most user-friendly way possible and the publishing [3] helped us streamline the experimentation process.

Chapter 3

Thesis Motivation

Robots are becoming increasingly prevalent and their use in therapy and social interaction has already been proven to be effective and important. Interaction with a robot provides the user with a sense of interest as well as can reduce the workload of the therapist/expert who would have to perform a same set of actions for many users, and also many times for a single user in order to achieve positive results. The therapist/expert's role cannot be played by anyone else without knowledge about the patient or the domain.
For such cases, machine learning turns out to be the best possible solution, since on training it can learn to behave and interpret data well enough to understand as well as predict the next action the therapist/expert might take.

The motivation is to create a system that achieves the goal of an autonomous system, that would require minimal or no involvement of the therapist/expert after sufficient training has been provided.

Chapter 4

System Architecture

The system consists of two modules, the user module and the wizard module. The user, the robot and the sequence learning task performed by the user, are all part of the user module. And the wizard module consists of an interface with which the therapist/expert interacts, a dynamic user modeling module, and an action selection network. Output from the user module is sent to the dynamic user module, where it is processed by the action selection network and a predicted next action is displayed on the wizard interface, along with user performance data. Figure 4-1 displays the architecture of the system.

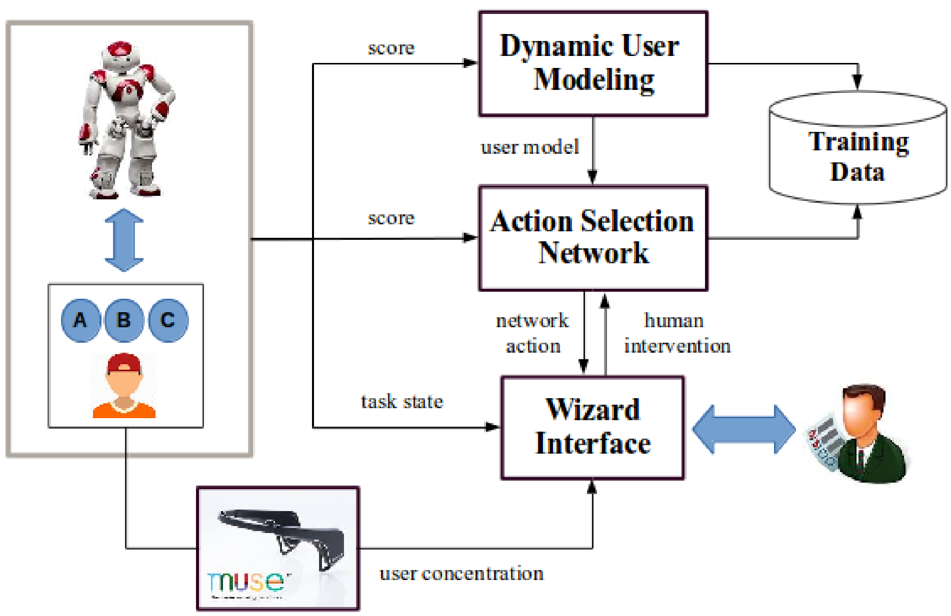

Figure 4-1 System Architecture

4.1 Sequence Learning Task

The sequence learning task is performed by the user, based on which they get a score. The task is a working memory retention task where the user has to remember a sequence. Performing the task involves the user listening to a sequence of characters spoken by the robot, remembering it, and pressing buttons in front of them to repeat the same sequence. The sequence varies in length but consists of only three characters ‘A’, ‘B’ and ‘C’. Buttons with these characters are placed in front of the user, so they can repeat the sequence by pressing these buttons. The length of the sequence is determined by the difficulty level selected in that particular round. There are 6 actions ranging from 0 to 5. Action 0, the easiest, corresponds to sequence length 3, action 1 for a sequence of length 5, action 2 is for a sequence of length 7, and action 3 is a sequence of length 9. Actions 4 and 5 are more concerned with feedback rather than increasing the difficulty of the task. Action 4 gives a positive or motivational feedback based on your performance in the previous round, i.e. if the user succeeded, it will motivate the user to keep doing well, but if they did not succeed, it will motivate the user to try harder and perform better. Action 5 is when a user receives negative or challenging feedback based on their performance in the previous round, i.e. if the user succeeded in the previous round, it will challenge them by asking them to repeat similar performance in the next round, while, if the user did not succeed, they will be asked to pay more attention and focus on the task. The length of the sequence for actions 4 and 5 remain the same as that of the previous round. Table 4-1 tabulates all the actions and difficulty for each action. The score for difficulties 0 to 3 are 1, 2, 3, and 4 respectively if the user succeeds and -1, -2, -3, and -4 if the user fails. Success and failure are determined on whether the user is able to remember the exact sequence and reproduce it by pressing buttons. For actions 4 and 5, the user is scored based on the length of the sequence in the previous round. The user is wearing a Muse headband while performing the task which gives EEG, brainwaves and concentration values for the user.

All data regarding the user's performance and EEG activity are sent to the expert interface for visualizing and to be used by the action selection network.

Table 4-1 Actions and their respective difficulty level

Difficulty (Length L)	A0	Easy	L = 3	
	A1	Medium	L = 5	
	A2	Normal	L = 7	
	A3	Hard	L = 9	
Feedback	A4	Positive	"Very good!! Keep going" or "Oh, you missed but don't worry!!"	Same difficulty
	A5	Negative	"Maybe that was too easy" or "Aren't you paying any attention?"	

4.2 Wizard Interface

The wizard interface is displayed to the therapist/expert, who helps in making a decision about the level of difficulty the user should face in the next round, based on their performance and EEG values in the previous round. Figure 4-2 displays the wizard interface.

Figure 4-2 Wizard Interface

As can be seen in Figure 4-2, the left portion of the wizard interface displays all information about the user, which are, the sequence of characters the user had to remember in the previous round; their average reaction time, which is the time a user takes to complete pressing the sequence on the buttons after the robot finished saying it out; the robot feedback, this can take 3 values, ‘None’ if no feedback was provided, and ‘Positive’ or ‘Negative’ if positive or negative feedback was provided respectively; the total score of the user in a particular session; the user’s score in the previous round, which will be a positive or negative value; the performance graph, which shows the current model of the user based on their performance in the entire session. Performance in each level of difficulty is displayed in a bar graph; and a plot for the concentration values during while they were performing the previous task. The right portion of the interface contains buttons which the ‘wizard’ can use to make decisions about the difficulty of the next round for the user. The ‘Play !’ button is used to start a new session for a user. Buttons ‘L = 3’, ‘L = 5’, ‘L = 7’, ‘L = 9’, ‘Positive Feedback’ and ‘Negative Feedback’ can used to determine the value of the next difficulty the user will face, where are intuitively labeled for sequence length 3, i.e. action 0, sequence length 5, i.e. action 1, sequence length 7, i.e. action 2, sequence level 9, i.e. action 3, positive feedback, i.e. action 4 and negative feedback, i.e. action 5, respectively. The ‘Visualizations’ button is used to visualize user clustering via a graph based on the data that has already been collected. Clustering is performed based on the user’s performance over all sessions they have had with the system. When the action selection network suggests the action for the next round, it is highlighted, and the wizard and click on any other button to change the action to be taken. If the expert does not change the suggested action in 3 seconds, it is performed. A timer in the bottom indicates the amount of time the wizard has to change the action proposed by the action selection network.

4.3 Dynamic User Modeling

In order to make autonomous and appropriate decisions, the SAR system needs to keep track of a user’s abilities and preferences. The representation of user patterns is a user model, based on which the system learns to perform the appropriate actions. There are many different user modeling approaches, depending on the application. A motivation of this work was to find an appropriate user modeling approach in a dynamic manner - dynamic user modeling.

Based on our approach, the user model depicts the abilities of the user in the specific game, considering user's performance history. The system keeps track of user performance and task context and updates dynamically the model, based on which it makes decisions. The user model describes user performance at each difficulty level as a vector {P1, P2, P3, P4}, where $P_i = P(success|difficulty = i)$. The initialized model is {-1, -1, -1, -1} to describe unobserved performance data at a specific difficulty level. The system updates the probabilities after each round based on the session history data and it uses the latest version of the model to select the next action [5].

The Figure 4-3 shows a dynamic user model for a user after a few rounds. 1, 2, 3, and 4 represent the four levels of difficulty. We can see from the figure that the user has always succeeded in first difficulty, i.e. a sequence of length 3; has failed in half the rounds played for second and fourth difficulty levels, i.e. sequence of length 5 and 9 respectively. For the third difficulty, i.e. a sequence of length 7, the user has failed 2 rounds and succeeded in one.

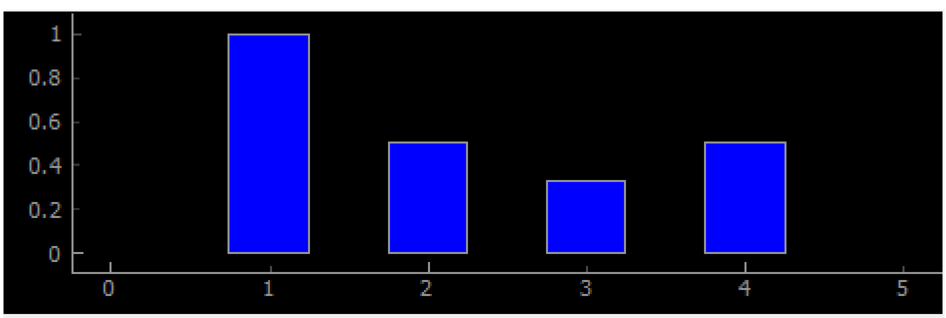

Figure 4-3 Dynamic user model for a user

Figure 4-4 is the dynamic performance model for the same user after 2 more rounds, of difficulty level 2 and 3, i.e. length of the sequence being 7 and 9. The user successfully completes both these rounds hence the user model is updated as in the figure to show the increase in performance for both the rounds. Figure 4-5 displays the updated model after the next round where the user succeeds in remembering yet another sequence of length 5, and Figure 4-6 displays the updated user model, when in the next round the user fails at remembering a sequence of length 9, resulting in a lower bar for the respective difficulty.

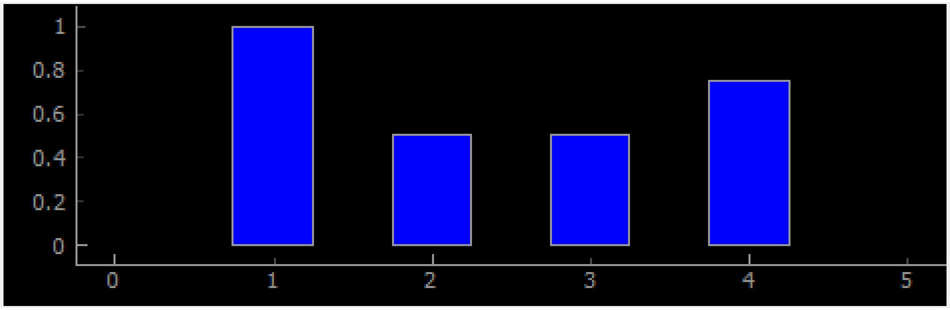

Figure 4-4 Dynamic user model for the same user after succeeding in the task for sequence length 7 and 9

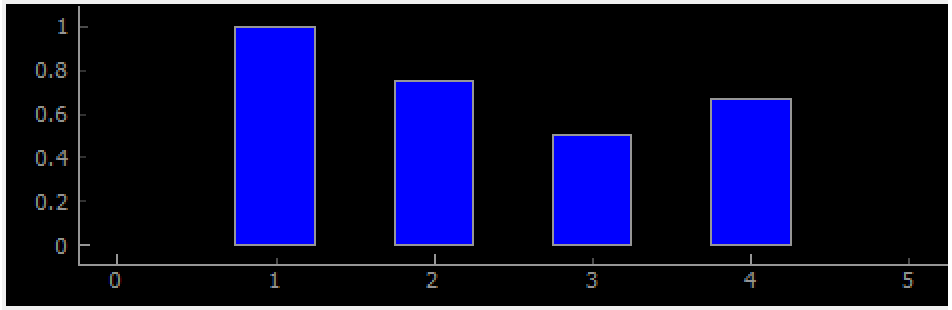

Figure 4-5 Dynamic user model for the same user after succeeding in the task for sequence length 5

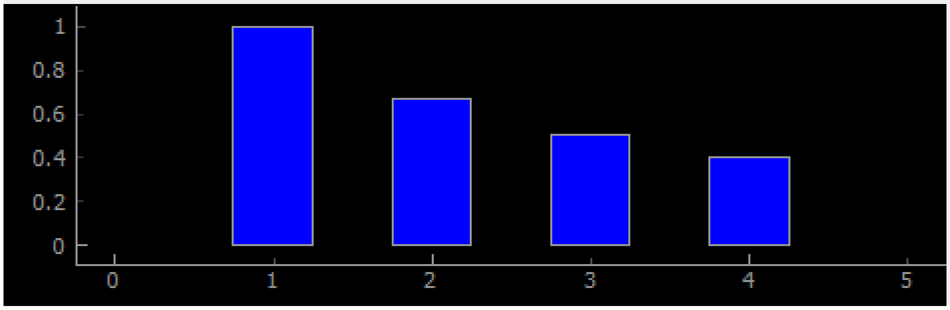

Figure 4-6 Dynamic user model for the same user after failing in the task for sequence length 9

The advantage of such a representation is that the user model encapsulates the history of a user's performance and can adapt to changes as well, i.e., when a user gets disengaged and fails where they should succeed, the model changes which will affect the future actions taken for him. We argue that such a dynamic user modeling approach combined with the current task state (task difficulty, robot feedback) are appropriate features to describe the current user and context, based on which the robot makes decisions online.

4.4 Action Selection Network

The action selection network helps select the next 'action', which defines the difficulty level the user should face in the round, as well as the robot verbal feedback. A neural network learns from the wizard's input and predicts the next action that should be attempted. The network takes 6 input parameters, 4 parameters from the dynamic user model, the score in the previous round and a normalized value action taken in the previous round. The neural network has three layers; the input layer takes the above mentioned 6 parameters as input, 1 hidden layer, and the output layer that provides a normalized value for the action to be taken in the next round. Figure 4-7 shows how the dynamic user model and the action selection network work together.

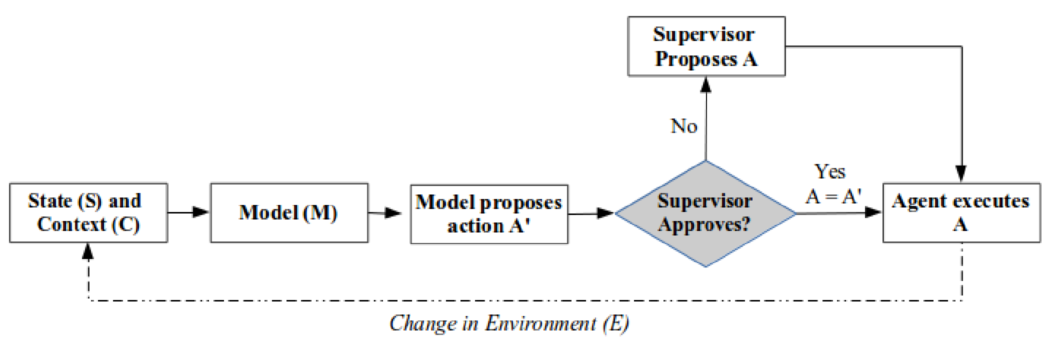

Figure 4-7 Action Selection Network Flowchart

As mentioned in [5] during a human-robot interaction based on online machine learning, it is important that initial errors be mitigated, which can be obtained with expert intervention to avoid unwanted effects of an incorrect action on the user. The above Figure 4-7 explains how the action selection network with dynamic user model as input, proposes an action A, and a supervisor/wizard can change it for an action that they deem to be better suited for the user. The state S and context C represent all the input received from the user module, the model M represents the dynamic user model that is used as input to the action selection network, A' is the action proposed by the model and A is the final action performed.

4.5 Task Concentration using MUSE EEG

Concentration is one of the parameters used by the wizard to determine the action to be performed in the next round. The Figure 4-8 and Figure 4-9 are examples of the concentration graph that are displayed on the interface for one of the users' during a session. Figure 4-8 displays a graph of the user when the concentration values are low, while Figure 4-9 displays the concentration graph for a user when they are really engaged in the task.

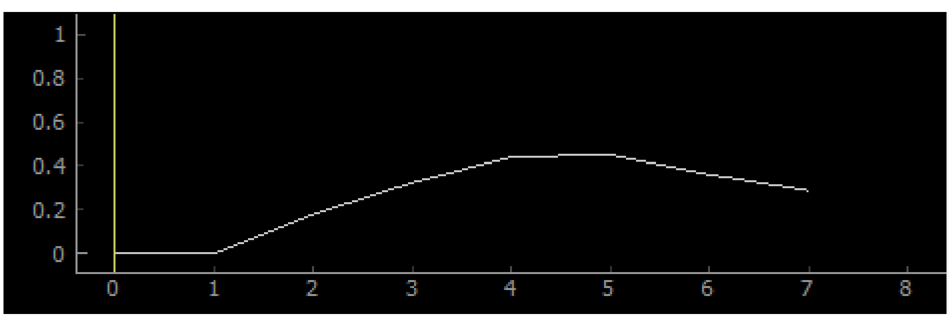

Figure 4-8 Concentration graph when a user is less concentrated on the task

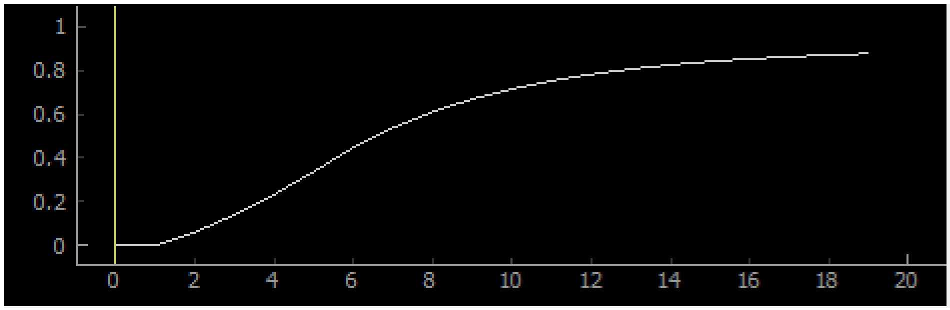

Figure 4-9 Concentration graph when a user is very concentrated on the task

The aim for the wizard is to maximize the user's performance and engagement in the task. Concentration is used as a parameter to determine user engagement. Muse head band gives values for concentration along with EEG values. The wizard is displayed a mean of concentration values over time, this enables the wizard to visualize how engaged the user was in the previous task and can be of guidance in making a decision about the next action to perform, so as to engage the enough, and not increase the difficulty when a user loses engagement on not being able to perform, and not decrease the difficulty where the user does not find it challenging enough to be engaged.

Chapter 5

Preliminary Data Collection

5.1 Experimental Procedure and Analysis

A socially assistive robotics experiment requires the interaction between the user and the robot to be engaging. The assistive robot helps improve the quality of the interaction and can be helpful for educational as well as health-care purposes. Preliminary work for this system involved engaging the user in different tasks during their interaction with the robot. These tasks require a user to use different capabilities and skills. The user's EEG, brain waves and concentration values were recorded and saved. The purpose of this work was to collect preliminary data and analyze the same.

5.1.1. Sequence Learning Task

The sequence learning task for this work involved three different types of tasks that the user was supposed for perform for each session. These task modes are Buttons, Speech and Flanker. The 'Buttons' task mode is the one used for current experiments as well, where the instructing robot says a sequence of characters, and the user is supposed to repeat these characters by pressing the respective buttons in front of them. The Speech mode required the user speak and repeat the sequence of characters said by the robot. And lastly, the Flanker mode, in which, given a sequence of characters, the user has to identify the middle character of the sequence, and then press its respective button. The difficulty levels and interaction between the user and the robot was similar to the task described in Table 4-1.

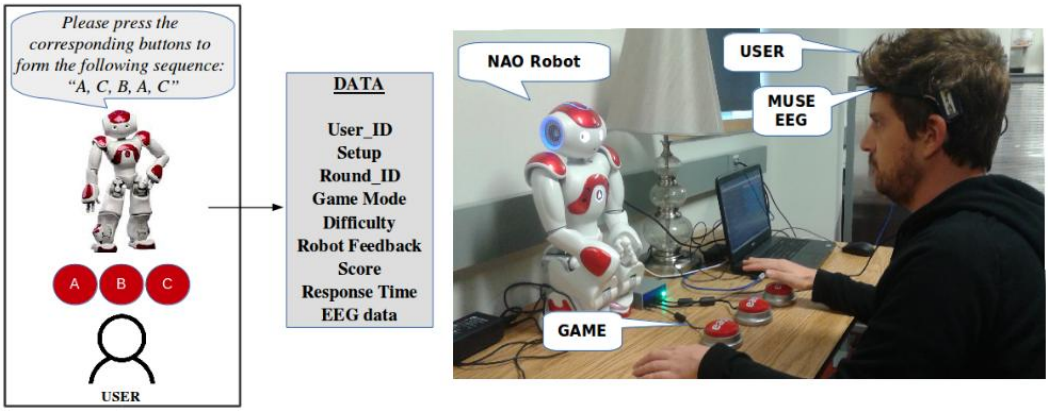

Figure 5-1 Experiment setup

5.1.2 Experimental Procedure

In order to collect interaction data a user study was conducted where participants performed the sequence learning tasks mentioned above. During the experiment, the NAO robot instructs and monitors and evaluates the user during the training session, collecting interaction data which includes EEG, brain waves and concentration values. At the beginning of the experiment, each user is asked to take place in front of the robot and wear the Muse EEG sensor. After the task administrator ensures the correct placement of the Muse sensor, the NAO robot greets the user and describes to them the sequence learning task and the different modes. After the introduction, the robot asks the user if the process was clear to them. During the task, the robot performs an action (A0-A5) that defines the difficulty level or the feedback type of the next turn. The task difficulties match the specification mentioned in Table 4-1, and the experiment setup is as displayed in Figure 5-1.

We defined two different experimental designs, in terms of how the task difficulty changes in each round. Each user performs the task for both designs: blocked and mixed. We followed these two different designs in order to capture user data under different variations of difficulty. The robot action sequence (including feedback actions) is predefined and same for all users.

1. Blocked Design: In this design, the difficulty levels are gradually increasing from the lowest (L = 3) to the highest (L = 9) difficulty, for each task mode. Each user has to perform the task for 9 rounds for each task mode, resulting in 9 x 3 = 27 rounds.

2. Mixed Design: In this design, the difficulty levels are mixed and change during the task. Each user has to perform the task for 12 rounds for each task mode, resulting in 12 x 3 = 36 rounds.

5.1.3 Analysis and Results

Initial statistical analysis of the data with respect to parameters difficulty, mode and concentration was made. It was found that the user's performance was highest and concentration was lowest during the 'Flanker' mode and the easiest difficulty level of the 'Buttons' and 'Speech' modes. This shows that since the user needs to remember only a single character in the 'Flanker' mode it could have been one of the easiest modes. It was also found that the reaction time of the user was faster without feedback and no significant differences were found between the positive and negative feedback. Machine Learning analysis was also performed on the data, and the analysis as well the entire dataset was published.

This work acted as a preliminary example of the interaction between the user and the robot and helped to visualize the requirements for a system that will be used by the therapist. It also helped identify parameters like reaction time, overall performance and the performance in the previous round that would help the therapist in making an informed decision about the user's next action, as well using in training the action selection network. This work is published in Robots for Learning workshop in HRI conference on March 2017, and is called 'Towards Designing a Socially Assistive Robot for Adaptive and Personalized Cognitive Training'.

Below, we show two examples of the analysis. Figure 5-2, shows the percentage of correct answers at the different difficulty levels. We observe that as the difficulty increases, the number of correct answers decreases. However, there are some users who could benefit from higher difficulty levels. Figure 5-3 shows the results of the user survey on how robot feedback affected users in their performance, as a self-report. Such results indicate that there is a need for different training strategies, based on user performance and preferences. The goal of this thesis is to learn such training strategies through the interaction with an expert.

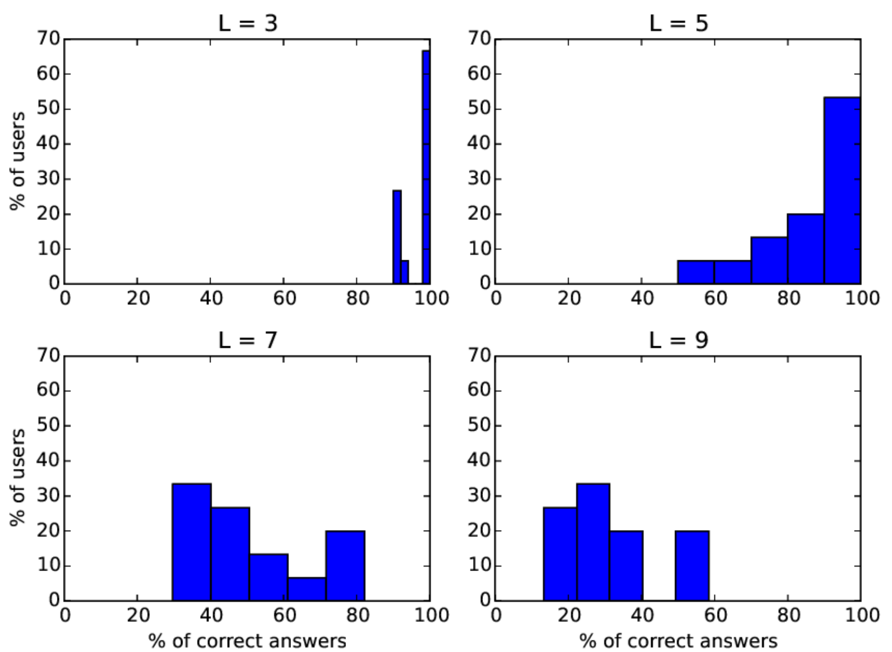

Figure 5-2 Analysis and Results

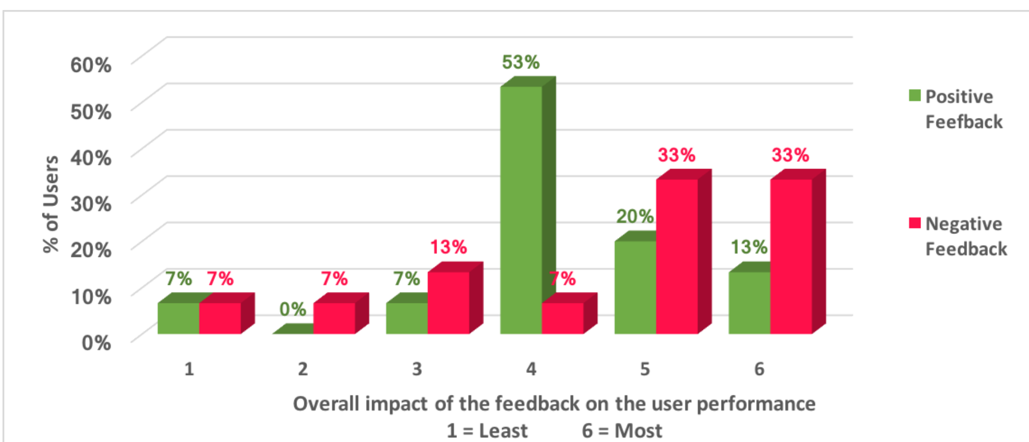

Figure 5-3 Feedback Analysis

The concentration values for all users for the buttons task in each difficulty level was plotted and Figure 5-4 shows that the users were more concentrated during a task with higher difficulty than on one with a lower difficulty.

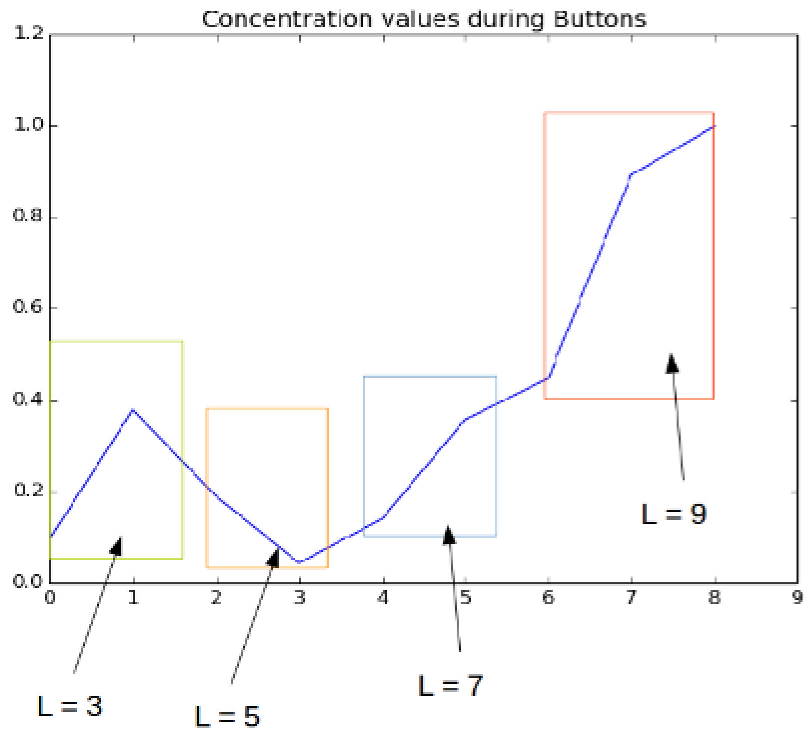

Figure 5-4 Concentration values for one user during task mode buttons

This shows that with increasing difficulty in the buttons task, the users' concentration in the task increased. Since dataset for all users was published with the paper, Figure 5-5 shows the database schema that was used to create the database to store the data. This schema can be used to query and get data for each any user during any task.

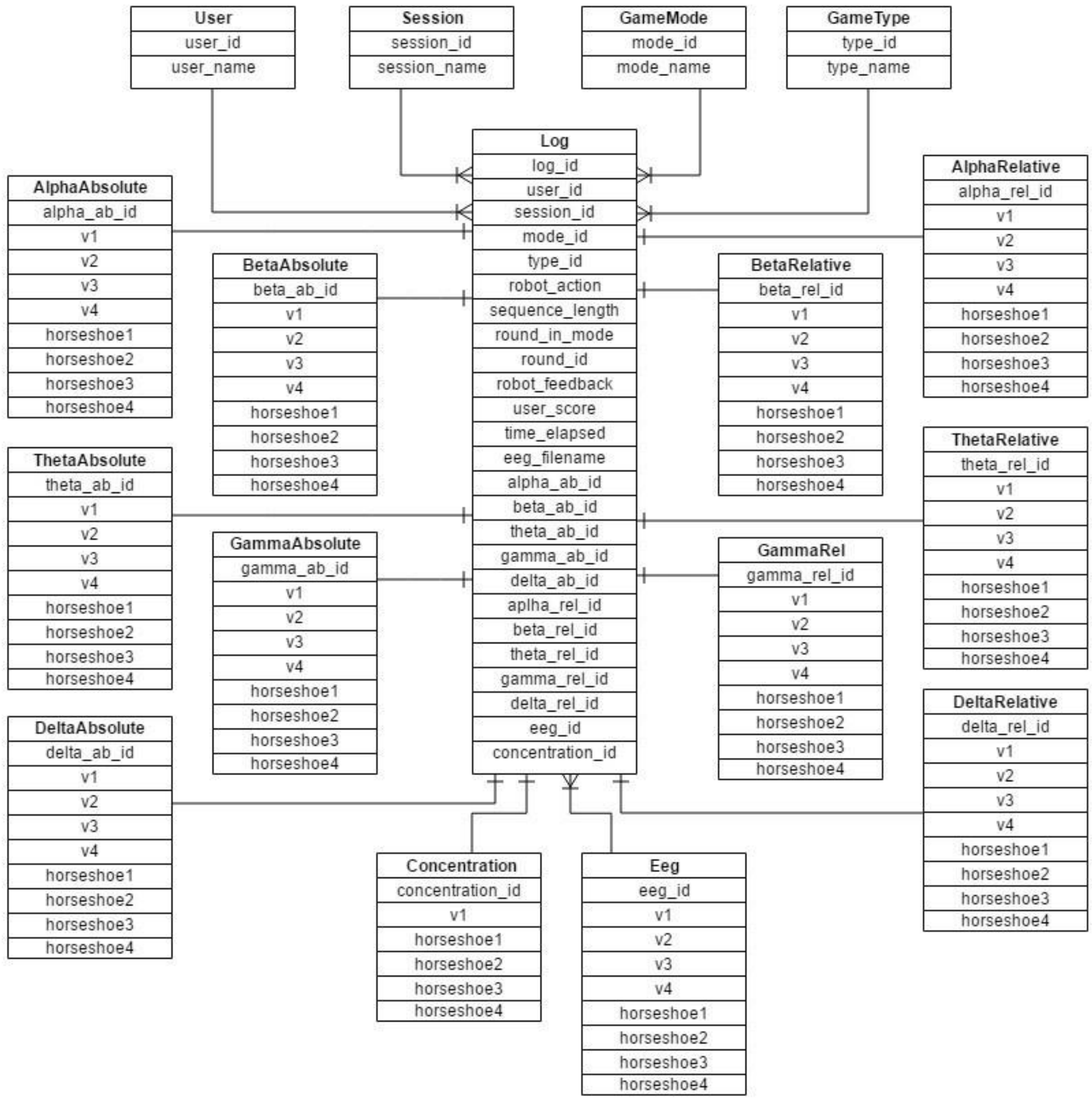

Figure 5-5 Database schema

5.2 User Clustering

Clustering was performed on the previous data to partition the observation into groups. K-means was used as the clustering technique which determines the observations into clusters in which each observation belongs to the cluster with the nearest mean, serving as the prototype of the cluster. For the purpose of visualizing the cluster to which a user belongs, data for each user was parsed to create their user models based on performance in each round, and this final user model was used for clustering. Users with similar performance models are clustered together and may be of some information to the therapist/wizard.

Chapter 6

System Integration

The user module and the wizard module are separate systems that are required to communicate with each other from frequently in order to share the user's data. Since the wizard module's presence should not be known to user, it was preferred if both the modules can be implemented such that they exist on separate systems and the 'wizard' does not have to be in proximity of the robot and user interaction. This required client-server architecture so the data updates to the interface would be seamless and do not hamper the human robot interaction.

6.1 The Client

The client for this system would be the user module, since it has data after every round that needs to be sent to the wizard interface. But apart from connecting with the wizard interface, it has multiple interactions with the NAO robot, the buttons using which the user performs the sequence learning task and the muse server, which continuously gives EEG, brainwaves and concentration data for the user. The client connects to the Nao robot using its application programming interfaces (APIs). The input from the buttons is processed as keystrokes in the client and the entered sequence is verified against the test sequence in the client itself. The client is responsible for generating a random sequence of the difficulty specified and then verifying if the sequence entered by the user is correct or not. The client is also connected to the muse headband which continuously sends data. Muse uses Open Sound Control (OSC) to pass data around. OSC is a simple protocol for sending data over a network. To save the EEG, brainwaves and concentration data for each round, the client continuously stores data from the Muse OSC server into data files. From these data files, at the end of each round, we obtain values for concentration, and then minimize the data to mean of every 10 points, and send it to the wizard module for visualization along with the sequence, total score, score in previous round, action in previous round and the average reaction time.

6.2 The Server

The server this case is the ‘wizard’ module. It acts as a server by always listening for data from the client connection. The server is responsible for updating the interface with the information provided by the client, using the data to create a dynamic user model, and send it to the action selection network, which then outputs a proposed action. These proposed action, dynamic user model and concentration values are also visualized on the interface. The interface acts as an agent using which the expert/therapist can provide their input, and change the proposed action. Once the next action is determined, by either the action selection network or the wizard, it is then returned to the client so the user can continue in the next round of the session with the difficulty determined by the server.

The client at the end of each round sends data to the server, who performs the process again to determine the new action for each round. This will happen for all rounds of the session.

Chapter 7

Experimental Procedure

The focus of this thesis is to make a system that minimizes the workload of the therapist/expert, a ‘wizard’ in this case. Hence, the experimental procedure defined is focused on the wizard and not the user interacting with the system. Each subject in our case therefore acts as a wizard, who uses the visualizations on the interface and decides the next action for the user. The user in this case is kept as a control factor and not varied during the experiment i.e. the same user is present for all subjects.

Each subject performs 2 sessions with the user, and each session consists of 20 rounds. A round consists of the robot saying a single sequence of characters and the user repeating that sequence using the buttons in front of him. The system has two phases, the non-learning phase (NL) phase and the learning phase. In the NL phase, the wizard interface does not use dynamic user modeling and action selection network, but outputs a random action as the proposed action to the subject. In the learning phase, the dynamic user modeling and action selection network are used to predict the proposed action for the subject. The aim of these phases is to validate the reduction in the therapist’s workload as a result of using user modeling and action selection network. Two protocols were designed for the subjects, in protocol ‘A’ the subject interacts with the system in non-learning phase for the first session, and with the system in learning phase for the second session. In protocol ‘B’, it is the opposite; the subject interacts with the system in learning phase for the first session, and with the system in a non-learning phase for the second session. This eliminates any bias created by the subjects (acting as the wizard) or the user.

7.1 Participants

A total of 4 participants acted as the subjects for the experiment, all of which were within the age range of 25-35. 2 participants were randomly chosen to work on Protocol ‘A’ and the other 2 for protocol ‘B’. One additional participant acted as a user, which performed the sequence learning task and interacted with the robot, for subjects during both A and B protocols.

7.2 Hypothesis

The hypothesis of the experiment is that the subject acting as ‘wizard’ should have to intervene more in the decision making process for next action during the non-learning phase of the session than the learning phase. The system should provide better action selection with the dynamic user model and the learning algorithm. By using two protocols, A and B, which alternate between the learning and non-learning phase, it is assured that there is no bias from the subject in terms of learning how to use the system.

7.3 Interaction Protocol

There are two interactions going on through the experiment, the first interaction is between the user and the robot, which should work seamlessly, and the user should not be able to notice any difference in the interaction based on the protocols. The second interaction is between the subject and the interface. This is what concerns us more, since the system aims at reducing the wizard’s workload. The subject/wizard should be able to visualize all data and make decisions as necessary.

For the experiments, each subject was explained all the parameters that are displayed on the interface and was asked to maximize the performance of the user through the overall session based on their dynamic user model, also ensuring that the user is always engaged in the task, based on their concentration values. The interface highlighted the proposed action, which the subject/wizard can change based on their decision. The interface waits for 3 seconds to for the wizard to change the proposed action, after which the action is finalized as the next action to be taken and sent to the user module to perform. Each subject was asked to perform one session and take a user survey, then move on the second session and take another user survey. The number of interventions the subject had to perform to change the decision of the system was measured for each session of the user.

It took around 15-20 minutes for each participant to get through both learning and non-learning sessions.

7.4 User Survey

The user survey consisted of questions that can help determine the subjects ease of using the system and their experience while using it. On the survey, a subject could rate their experience with the system on a level of 1 to 7 where 1 means they totally disagree with the statement, while 7 mean that they totally agree. All intermediate levels range in increasing levels of agreement from 1 to 7. The questions asked after each session to the user are; the overall experience with the system was enjoying and easy; the robot could operate unsupervised correctly; the robot provided correct suggestions; my workload got lighter as interaction was progressing. At the end of both the sessions, the subject was also asked questions about the system which they were supposed to rate on a similar scale. These questions were, visualization of textual information enhanced my decision making; visualization of user performance model enhanced my decision making; and visualization of user concentration enhanced my decision making.

Chapter 8

Experimental Results

8.1 Interaction Data

Concentration values are used to determine the engagement of a user in the sequence learning task. The user should be more engaged during the learning phase of the experiment than the non-learning phase in order to move towards a completely autonomous system. The Figure 8-1 below is the concentration graph for a user's session during the non-learning phase.

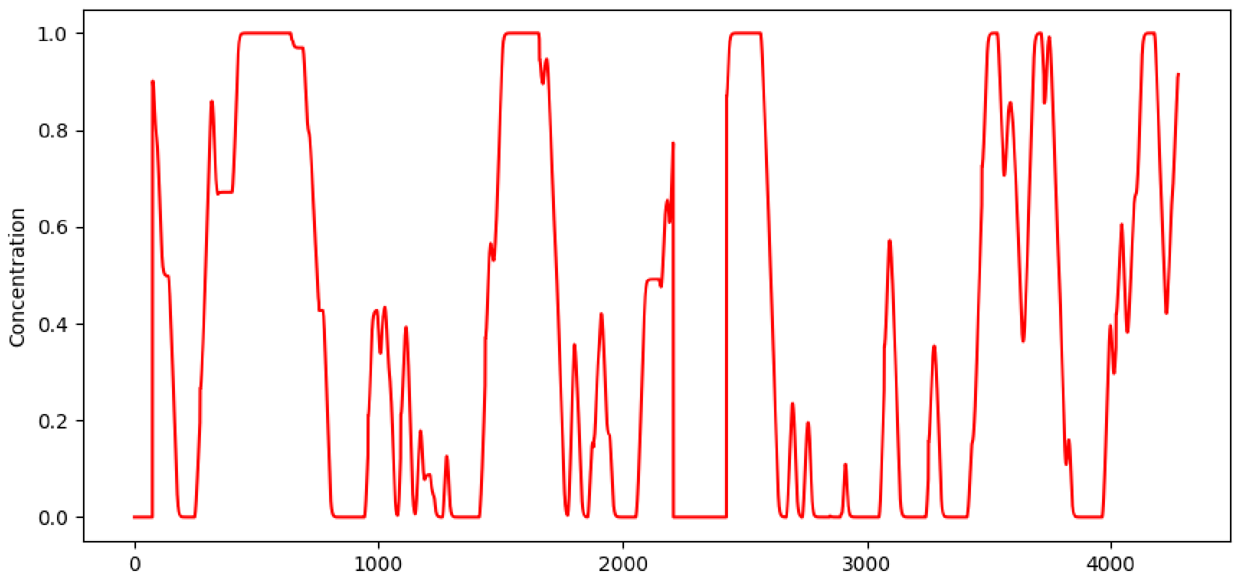

Figure 8-1 Concentration graph for a user in the non-learning phase

Concentration values provided by the muse headband range between 0 and 1, with 0 being the lowest and 1 being the highest. As we can observe in the Figure 8-1, concentration increases and decreases various times. This is because of the nature of the sequence learning task, which requires the user to remember the sequence of characters in each round, hence the user's concentration decreases at the end of each round until they get the next sequence. We can also see from the figure that this particular user has maximum concentration, i.e. peaks reaching value 1, 6 times during a non-learning session. Figure 8-2 displays the concentration graph for the same user during the learning phase.

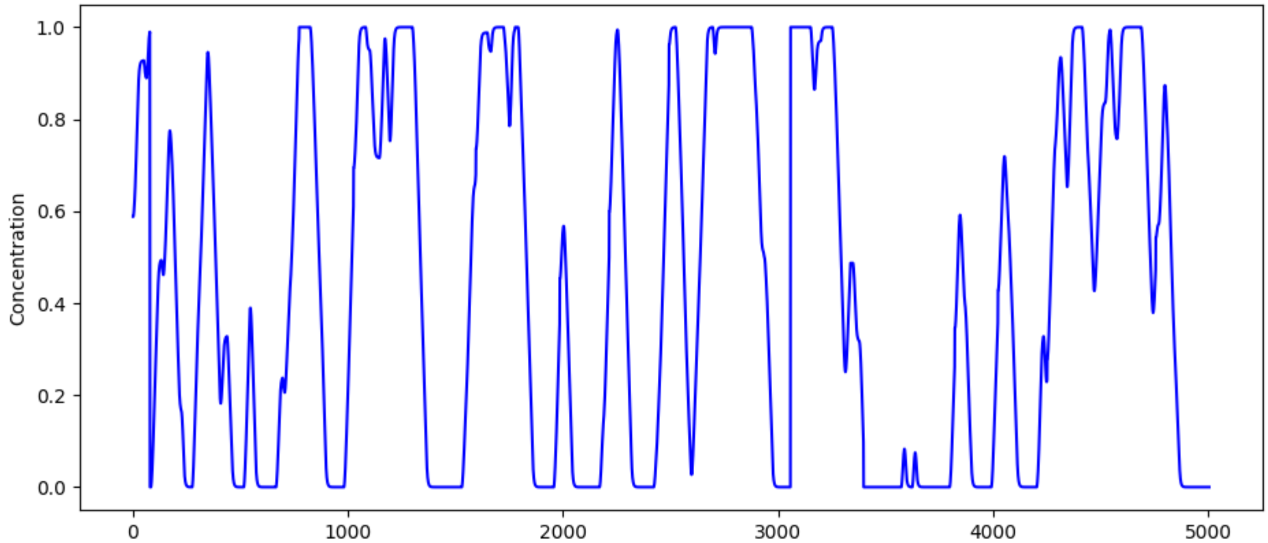

Figure 8-2 Concentration graph for a user in the learning phase

As we can see in Figure 8-2, the pattern is similar to the non-learning phase, with the difference being in the number of times the user reaches highest concentration, i.e. peaks reaching value 1, which is 11 times. Figure 8-3 shows both the graphs together for better comparison.

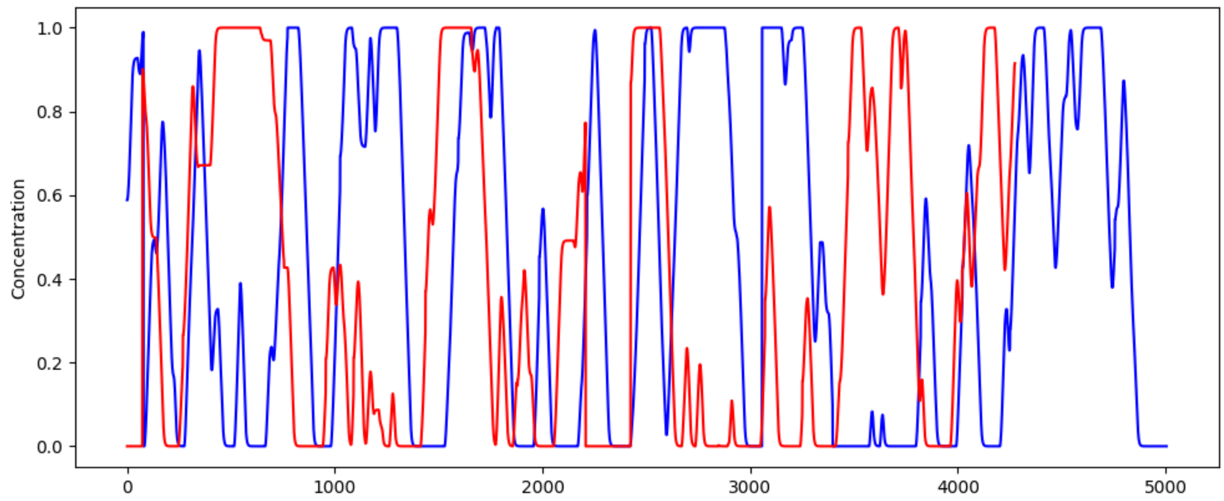

Figure 8-3 Concentration graph for a user in the learning phase vs. non-learning phase

As mentioned above, Figure 8-3 shows the concentration graph for a particular user in the learning phase vs. the non-learning phase, where the red line indicates the non-learning phase and the blue line indicates the learning phase. As we can see, time taken for the learning phase is longer but the user is more concentrated during the tasks, i.e. the number of peaks reaching value 1 are more during the learning phase than the non-learning phase. This indicates that the user was more concentrated during the sequence learning tasks, and thus, is more engaged in the task.

8.2 Intervention Data

An intervention is when the subject changes the decision proposed by the action selection network using the wizard interface. The number of interventions by the subject are compared from the learning and non-learning phase. To reduce the work load of the therapist, according to the hypothesis, the number of interventions by the subject should be lesser in the learning phase than the non-learning phase. This trend was observed in the 6 subjects that acted as wizards for the system. Figure 8-4 shows the number of interventions for each user in the non-learning phase and the learning phase.

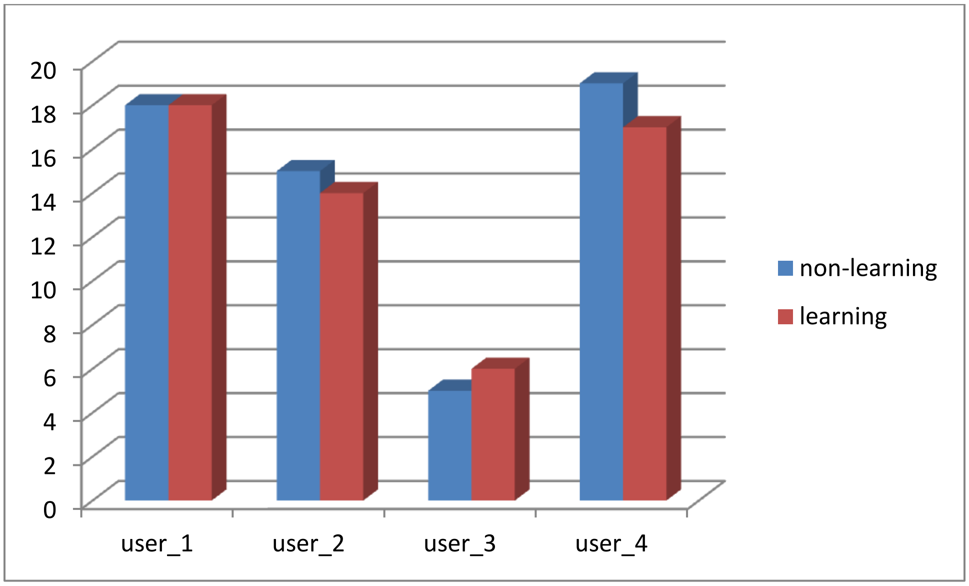

Figure 8-4 Intervention for each user in the learning phase and non-learning phase

As we can observe from the Figure 8-4, the number of interventions by the subjects in non-learning phase versus the learning phase is more 2 out of 4 times, while it is same for 1 user and lesser for 1 user. Since the number of rounds in each session was 20, we can say that the subjects had to intervene in 71.25% of the proposed actions to change it in the non-learning phase, while for the learning phase the subjects had to intervene only 68.75% of the times. This reduction in interventions can be further decreased as the action selection network is trained more, and can lead to an autonomous system, as hypothesized, hence reducing the workload of the therapist/expert. We can also observe that the last user, when the action selection network is most trained, the intervention ratios over time have reduced.

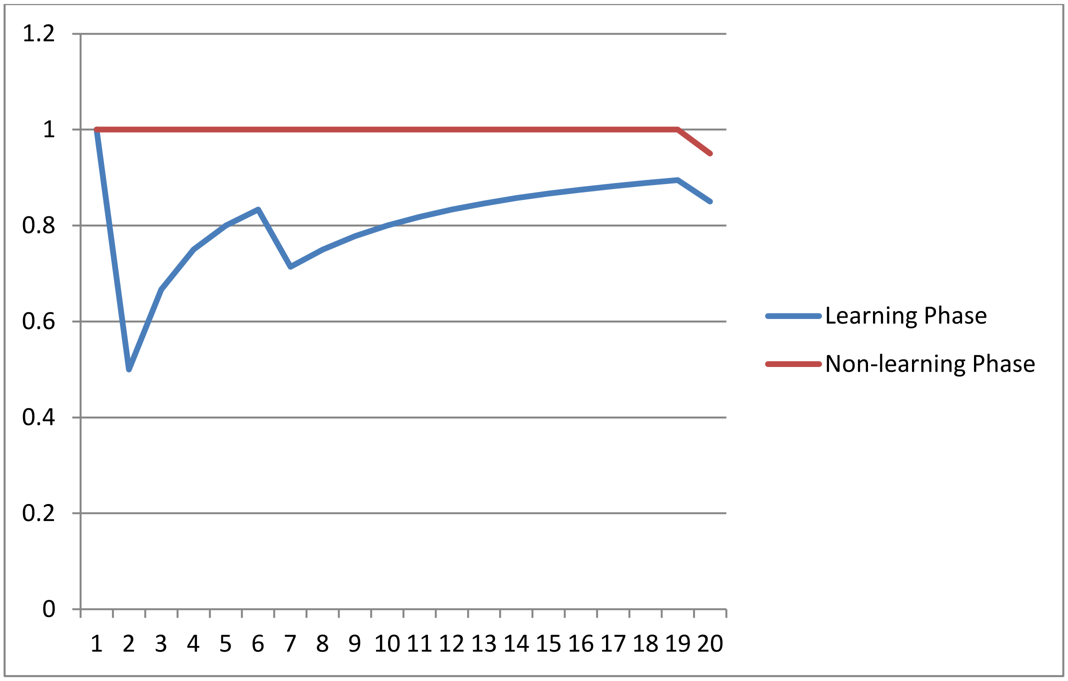

Figure 8-5 Intervention ratio for subject 4

As we can observe from the figure above, intervention ratio for the last user, when the action selection has access to maximum training data, the number of interventions required for the subject for each round of learning phase vs. each round of non-learning phase is always lesser. This shows that with time, the number of interventions required is reduced.

8.3 Questionnaire Data

The user was asked to fill out a questionnaire at the end of each session, and also after the end of the experiment. Following is the graph for statements about the visualizations on the interface, 3 statements were made which are, whether textual visualization helped in decision making (S9), whether user performance model visualization helped in decision making (S10) and whether user concentration visualization helped in decision making (S11). The user was asked to rate these statements from values 1 to 7, and Figure8-6 below shows the response of the subjects for these statements.

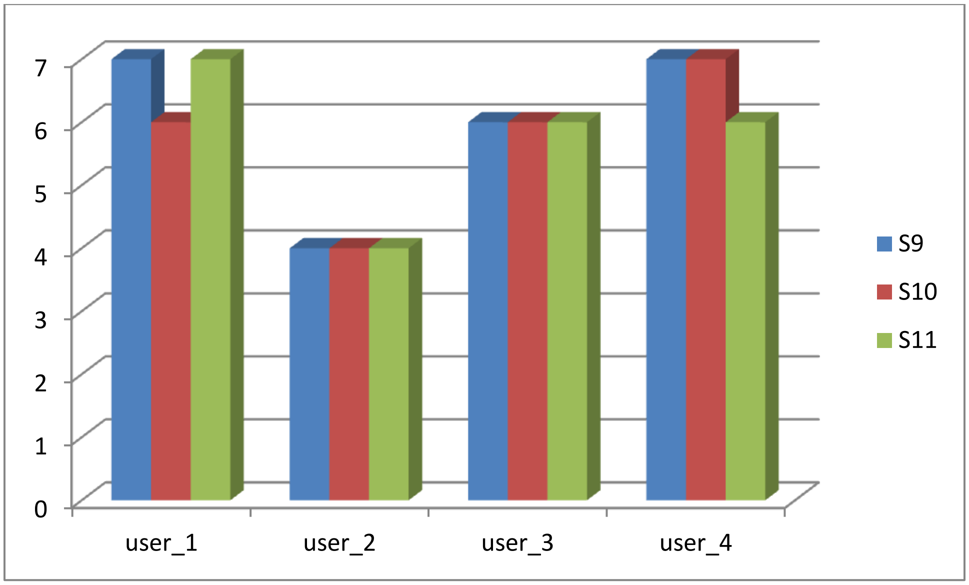

Figure 8-6 Questionnaire data for all users for system related statements

As we can see from the graph above, all subjects have rated 4, 6, and 7 for textual visualization being helpful in the decision making process, i.e. S9. All subjects have rated 4, 6, and 7 for user performance model helping them in making decisions i.e. S10. These high ratings indicate that subjects agree to them to quite an extent. For statement S11, that is, whether user concentration visualization helps in the decision making, all users again rate the system at 4, 6, and 7. The high rate of agreement indicates that the visualizations on the wizard interface are effective and do help in the decision making process.

Apart from questions about the interface, the subjects were also asked to rate their experience with the wizard module after each session. One of the sessions was non-learning while the other was learning. The statements that they rated are, the overall interaction with the system was enjoying and easy (S1), the robot could operate unsupervised correctly (S2), the robot provided correct suggestions (S3), and my workload got lighter as interaction was progressing (S4). Figure 8-7 displays the user responses to each of these statements after non-learning phase, and fig displays the user responses to each of these statements after the learning phase.

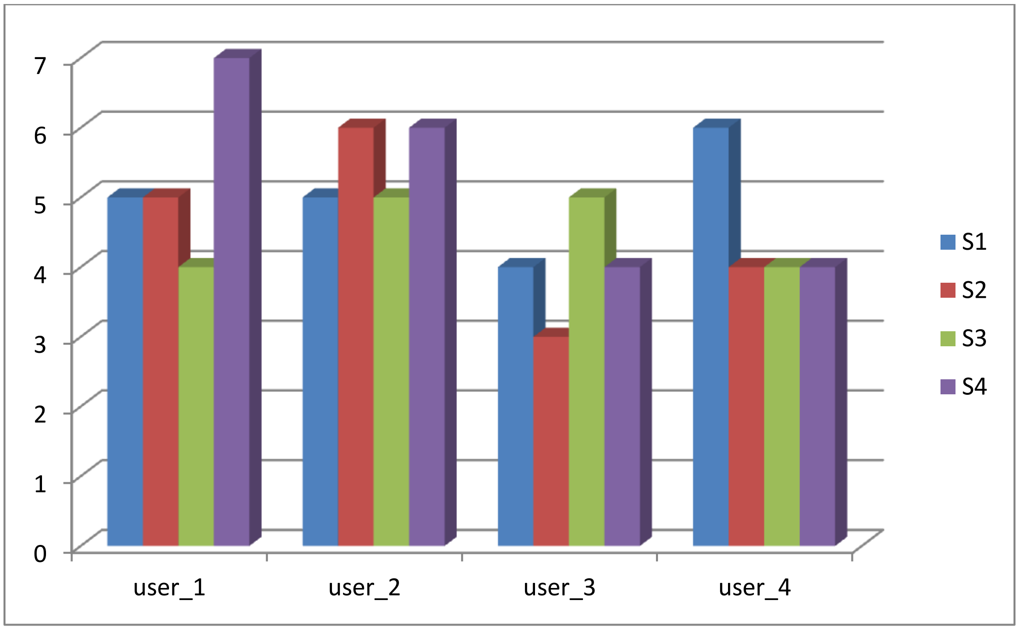

Figure 8-7 Questionnaire data for all users after the non-learning phase

Figure 8-7 displays the answer to the questionnaire after the non-learning task. 5 out of 6 subjects found the experience of interacting with the system enjoying and easy, and agreed to the statement S1. All users agree, by selecting ratings of 4 and more, that the robot could operate unsupervised correctly, i.e. S2. All users also agree, by selecting ratings of 4 or more, that the robot provides correct suggestions, i.e. S3. As for statement S4, we can observer that from user 1 to user 4 the rating decreases, this shows that the users felt that their workload decreased as time passed in comparison to the learning phase. Since user 1 had almost non-trained action selection network, for them the work load in learning phase and non-learning phase were not that different, while the last user felt that that their workload did not reduce much as they interacted with the system, hence rating 4, which is slightly agree. Hence, we can observe that a trained action selection network does perform better than a random system, and can with more training lead to an autonomous system.

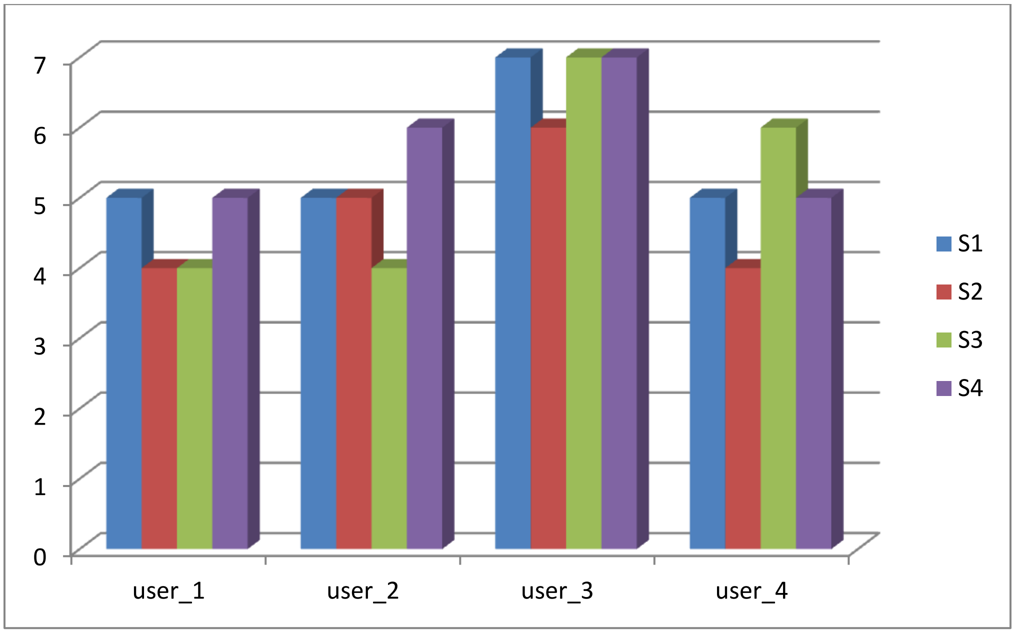

Figure 8-8 Questionnaire data for all users after the learning phase

Figure 8-8 displays the answer to the questionnaire after the learning task. All 4 subjects found the experience of interacting with the system enjoying and easy, and agreed to the statement S1. All users agree, by selecting ratings of 4 and more, that the robot could operate unsupervised correctly, i.e. S2. All users also agree, by selecting ratings of 4 or more, that the robot provides correct suggestions, i.e. S3, all users also agreed, by selecting rating of 4 or more, that their workload got lighter as the interaction was progressing. As for statement S4, user 1 felt that their work load did not reduce as the interaction proceeded, since their rating for non-learning phase was 7 and learning phase was 5, while user 4 agreed that their workload decreases since during the non-learning phase their rating was 4, while for the training phase, the rating provided is 5. This goes on to show that the action selection network on more training does and will provide better action selection. The statements S1, S2 and S3 help in determining that since the system and robot can both operate unsupervised; an autonomous system can be created.

8.4 Performance Data

The following is the performance for each user in both the learning and non-learning phase of the task. The performance here is for the user performing the sequence learning task and not the subject acting as the therapist/expert. Since maximizing the user's performance was the wizard's task this is analyzed.

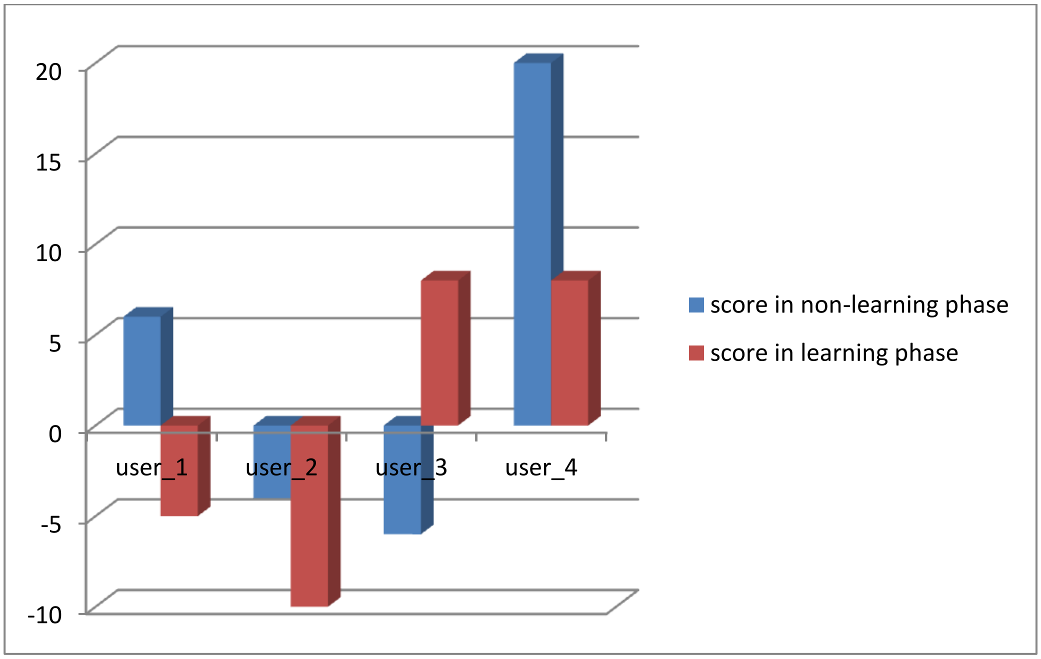

Figure 8-9 Performance of the users in the learning phase and non-learning phase

As we can observe from the Figure 8-9, no particular correlation is seen in the user's performance in the learning and non-learning phase. This can be attributed to the fact that each subject acting as a therapist/expert attempted to maximize the user's performance based on what they thought was the best action by increasing or decreasing the difficulty of the task, this involves a learning curve for how the system works, as well as understanding the user's performance, also considering that the subjects were not actual experts to be able to make the best decisions. All these factors contributed to the random nature of the performance from the user.

8.4 Conclusion

The intervention data shows that the no. of interventions subject made, are lesser during the learning phase, than the non-learning phase which is as expected. This helps to show that the dynamic user modeling and action selection network will, given more time and training, start to reduce the wizard's workload. The questionnaire on the other hand, showed that the subjects did not initially feel that the workload reduced as the interaction progresses in the learning phase as well, but as the number of interactions increase the action selection network has more been trained more, the last user does feel that the workload reduces more for the learning phase than the non-learning phase. We must also consider that for most first sessions, whether it is a learning phase or non-learning phase, the user will perceive the learning curve of using the system as part of the work apart from the interventions required. And since during the second session the user is better acquainted with the system, they might do not perceive that as workload and the decision making process takes priority. This helps us understand that initially the 'wizard' would require some training on using the system for him to better help in maximizing the user's performance.

Apart from the user survey data that helps us look at the system from an HCI perspective, the actual intervention data does show promise that with more training the system holds great potential to becoming an autonomous system.

References

[1] Defining Socially Assistive Robotics
David Feil-Seifer and Maja J Matari´c
Interaction Laboratory, University of Southern California
Proceedings of the 2005 IEEE, 9th International Conference on Rehabilitation Robotics

[2] Assessment of Task Engagement using Brain Computer Interface Technology
Paul Mccullagh, Gaye Lightbody, Leo Galway, Chris Brennan, David Trainor
School of Computing and Mathematics, Ulster University, UK
Ambient Intelligence and Smart Environments: Workshop Proceedings of the 11th International Conference on Intelligent Environments, At Prague, Volume: 19

[3] Towards Designing a Socially Assistive Robot for Adaptive and Personalized Cognitive Training
Konstantinos Tsiakas, Cheryl Abellanoza, Maher Abujelala, Michalis Papakostas, Tasnim Makada, Fillia Makedon
CSE Department, University of Texas, Arlington
International Conference on Human Robot Interaction 2017 - Robots for Learning Workshop, At Vienna, Austria

[4] Human-Guided Learning of Social Action Selection for Robot-Assisted Therapy
Emmanuel Senft, Paul Baxter, Tony Belpaeme
Centre for Robotics and Neural Systems,
Cognition Institute Plymouth University, U.K.
Proceedings of The 4th Workshop on Machine Learning for Interactive Systems at ICML 2015, PMLR 43:15-20, 2015

[5] Adaptive Generation in Dialogue Systems Using Dynamic User Modeling
Srinivasan Janarthanam, Oliver Lemon
Heriot-Watt University
Computational Linguistics, December 2014, Vol. 40, No. 4, Pages: 883-920

[6] Measuring Task Engagement as an Input to Physiological Computing
Stephen H Fairclough, Katie C Ewing, Jenna Roberts
Liverpool John Moores University
Affective Computing and Intelligent Interaction and Workshops, 2009. ACII 2009. 3rd International Conference

[7] Interactive Learning and Adaptation for Robot Assisted Therapy for People with Dementia
Konstantinos Tsiakas, Cheryl Abellanoza, Fillia Makedon
University of Texas at Arlington
PETRA '16 Proceedings of the 9th ACM International Conference on PErvasive Technologies Related to Assistive Environments

[8] SPARC: Supervised Progressively Autonomous Robot Competencies
Emmanuel Senft, Paul Baxter, James Kennedy, and Tony Belpaeme
Centre for Robotics and Neural Systems, Plymouth University, United Kingdom
Social Robotics: 7th International Conference, ICSR 2015, Paris, France

[9] Pay Attention! Designing Adaptive Agents that Monitor and Improve User Engagement
Dan Szafir, Bilge Mutlu
Department of Computer Sciences, University of Wisconsin–Madison
Proceedings of the 2012 ACM annual conference on Human Factors in Computing SystemsProceedings of the 2012 ACM annual conference on Human Factors in Computing Systems

[10] Supervised Autonomy for Online Learning in Human-Robot Interaction
Emmanuel Senft, Paul Baxter, James Kennedy, Severin Lemaignan, Tony Belpaeme
Playmouth University, Lincoln Centre for Autonomous Systems, Ghent University
Pattern Recognition Letters 2017

Biographical Information

Tasnim Inayat Makada received her Bachelor of Engineering degree in Computer Engineering from the Saurashtra University in India. She joined University of Texas at Arlington in Fall 2015 and worked with Heracleia Human-Centered Laboratory as a student researcher to work on her thesis under the guidance of Dr. Filia Makedon. She will receive her Master of Science degree in Computer Science in Spring 2017.

Tasnim has worked as a software engineer for 4.5 years in India, with 2 years for Infostrech Solutions Pvt. Ltd. and 2.5 years at Persistent Systems, and interned at Lucid LLC. during Summer 2016. Her research interests are Human-Computer Interaction, Machine Learning, and Socially Assistive Robotics.