Towards Empathetic Care Robots

Stefania Costantini, Pierangelo Dell'Acqua, Abeer Dyoub and Andrea Monaldini

Department of Information Engineering, Computer Science and Mathematics
University of L'Aquila, Italy
abeer.dyoub@univaq.it

29-31 May 2023

Outlines

Introduction

Introduction

- Healthcare is shifting into patient-centered healthcare with the objective to empower patients to become active participants in their care and to ensure better health outcomes.

Introduction

- Healthcare is shifting into patient-centered healthcare with the objective to empower patients to become active participants in their care and to ensure better health outcomes.
- However, the nonclinical needs of patients mental health and well-being are frequently overlooked by contemporary patient-centered healthcare models.

Introduction

- Healthcare is shifting into patient-centered healthcare with the objective to empower patients to become active participants in their care and to ensure better health outcomes.
- However, the nonclinical needs of patients mental health and well-being are frequently overlooked by contemporary patient-centered healthcare models.
- The COVID19 pandemic has accelerated the development of robots and virtual assisted living that can help care for persons with disabilities and aging adults both physically and emotionally.

Introduction

- Healthcare is shifting into patient-centered healthcare with the objective to empower patients to become active participants in their care and to ensure better health outcomes.
- However, the nonclinical needs of patients mental health and well-being are frequently overlooked by contemporary patient-centered healthcare models.
- The COVID19 pandemic has accelerated the development of robots and virtual assisted living that can help care for persons with disabilities and aging adults both physically and emotionally.
- Given the intimate human-machine interaction in the case of care robots, it has become fundamental for these robots to demonstrate an empathetic behavior.

Introduction

- Healthcare is shifting into patient-centered healthcare with the objective to empower patients to become active participants in their care and to ensure better health outcomes.
- However, the nonclinical needs of patients mental health and well-being are frequently overlooked by contemporary patient-centered healthcare models.
- The COVID19 pandemic has accelerated the development of robots and virtual assisted living that can help care for persons with disabilities and aging adults both physically and emotionally.
- Given the intimate human-machine interaction in the case of care robots, it has become fundamental for these robots to demonstrate an empathetic behavior.
- This would result in more productive and delightful interaction that contribute to the patient well-being and mental health and to the **trust** relation between the patient and the machine.

Introduction

Artificial Empathy

Artificial Empathy (AE) refer to the development of AI systems, such as care robots or virtual agents, that are able to detect and respond to human emotions in an empathetic way.

Introduction

Virtual Reality

Virtual Reality (VR) defines as a technology that creates simulated environments to mimic real-world situations.

Introduction

Virtual Reality

Virtual Reality (VR) defines as a technology that creates simulated environments to mimic real-world situations.

VR application in Healthcare

VR can turn threatening and tedious conditions into safe and enjoyable states. In recent years, employing this technology has been considered for the treatment of many mental illnesses especially for anxiety disorders. One approach that can be implemented in VR to treat anxiety is exposure therapy (VRET), it is client-centered and helps clients confront fear-inducing stimuli through guided exposures and is often paired with cognitive–behavioral therapy

Behavior Trees

- Behavior Trees (BTs) were invented as a tool to enable modular AI in computer games.

Behavior Trees

- Behavior Trees (BTs) were invented as a tool to enable modular AI in computer games.
- A behavior tree is essentially a mathematical model of plan execution, where each element (task, action, etc.) of a plan is associated to a node in the tree.

Behavior Trees

- Behavior Trees (BTs) were invented as a tool to enable modular AI in computer games.
- A behavior tree is essentially a mathematical model of plan execution, where each element (task, action, etc.) of a plan is associated to a node in the tree.
- Their strength comes from their ability to create complex tasks composed of simple tasks without worrying how the simple tasks are implemented.

Behavior Trees

- In our case, a BT is composed of the following types of nodes, where the type denotes the kind of task related to node execution:

Behavior Trees

Leaf Nodes:

- Action: An action represents a behavior that the character can perform. The action returns success, failure, or running state. An action is depicted as a white, rounded rectangle.

Behavior Trees

Leaf Nodes:

- Action: An action represents a behavior that the character can perform. The action returns success, failure, or running state. An action is depicted as a white, rounded rectangle.
- Condition: A condition checks an internal or external state. It returns either success or failure. A condition is represented as a gray, rounded rectangle

Behavior Trees

Inner Nodes:

Behavior Trees

Inner Nodes:

- Sequence Selector: A sequence selector is a node that typically has several child nodes that are executed sequentially. If every child node returns success, then this selector returns success. Should any child fail, the selector immediately returns failure. If a child returns running, the selector also returns running. A sequence selector is depicted as a gray square with an arrow across the links to its child nodes.

Behavior Trees

Inner Nodes:

- Sequence Selector: A sequence selector is a node that typically has several child nodes that are executed sequentially. If every child node returns success, then this selector returns success. Should any child fail, the selector immediately returns failure. If a child returns running, the selector also returns running. A sequence selector is depicted as a gray square with an arrow across the links to its child nodes.
- Priority Selector: A priority selector has a list of child nodes which it tries to execute one at a time, with respect to the specified order, until one of them returns success. If none of the children succeeds, then the this selector returns failure. If a child is running, it returns running. A priority selector is represented with a gray circle with a question mark in it.

Behavior Trees

Inner Nodes (continued):

Behavior Trees

Inner Nodes (continued):

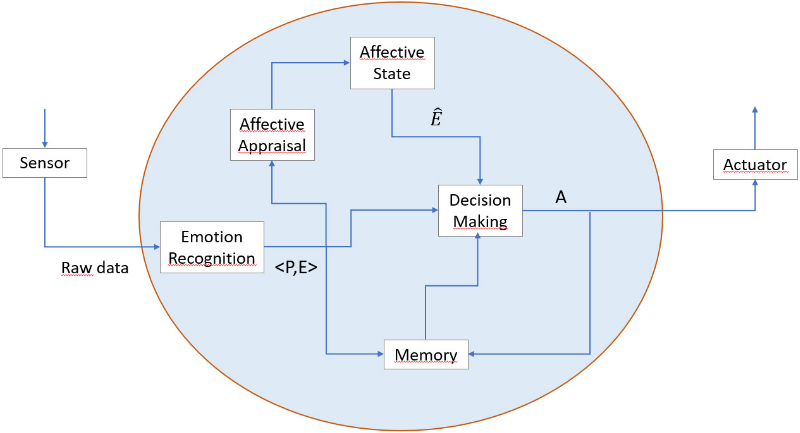
- Parallel Node: A parallel node executes all of its child nodes in parallel. A parallel node can stop executing its child nodes. One may specify the number of child nodes that must execute successfully for the parallel node to succeed, and those that must fail in order for the parallel node to fail. A parallel node is depicted as a gray circle with a P in it.

Behavior Trees

Inner Nodes (continued):

- Parallel Node: A parallel node executes all of its child nodes in parallel. A parallel node can stop executing its child nodes. One may specify the number of child nodes that must execute successfully for the parallel node to succeed, and those that must fail in order for the parallel node to fail. A parallel node is depicted as a gray circle with a P in it.
- Decorator: A decorator is a node that acts as a filter that places certain constraints on the execution of its single child node without affecting the child node itself. Decorators are represented as diamonds with descriptive text inside

A Framework for Empathetic Agents



A Framework for Empathetic Agents

- **Emotion Recognition:** This module receives the raw data from the sensory input and processes it to synthesize the user's affective/emotional state E.

A Framework for Empathetic Agents

- **Emotion Recognition:** This module receives the raw data from the sensory input and processes it to synthesize the user's affective/emotional state E.
- **Affective Appraisal** refers to the process in which events from the environment are evaluated in terms of their emotional significance.

A Framework for Empathetic Agents

- **Emotion Recognition:** This module receives the raw data from the sensory input and processes it to synthesize the user's affective/emotional state E.
- **Affective Appraisal** refers to the process in which events from the environment are evaluated in terms of their emotional significance.
- **Affective State** refers to how an entity is currently feeling, that is the product of its emotions at a certain moment in time. Emotions can be represented as signals coming from the Affective Appraisal module. The set of all signals of the same type forms the corresponding emotional state.

A Framework for Empathetic Agents

- **Decision Making** This module is responsible for selecting the next action to execute. It receives inputs from the Emotion Recognition module as well as from the Affective State and the Agent Memory module.

A Framework for Empathetic Agents

- **Decision Making** This module is responsible for selecting the next action to execute. It receives inputs from the Emotion Recognition module as well as from the Affective State and the Agent Memory module.
- **Agent Memory** The knowledge base represents the memory of the agent. Here, all information from sensory inputs P from the environment and user's emotional state E as well the actions A selected to be executed are stored with a time stamp.

A Framework for Empathetic Agents

- **Decision Making** This module is responsible for selecting the next action to execute. It receives inputs from the Emotion Recognition module as well as from the Affective State and the Agent Memory module.
- **Agent Memory** The knowledge base represents the memory of the agent. Here, all information from sensory inputs P from the environment and user's emotional state E as well the actions A selected to be executed are stored with a time stamp.
- **Actuators** The actions A selected by the Decision Making module are passed to actuators, whose role is to execute them on the environment. Actions come together with an emotion encoding to display agents emotions via verbal or visual communication. Each actuator, depending on the type of application, must be equipped with the ability to render the emotional aspect of actions. One example could be a verbal communication to a human while having a smiling face.

Emotional Behavior Trees

- It is not straightforward to couple behavior trees with emotions to mimic human emotional decision making. If we wish to have a natural and interesting agent's behavior, it is important that characters behave in an emotional way.
- It could be claimed that it is possible to incorporate emotions into behavior trees by merely using emotions in the conditions.
- However, doing so may create large cumbersome behavior trees that are difficult to manage.

Emotional Selector

- To take emotions into consideration, Johansson and Dell'Acqua extended the definition of behavior trees and introduced a new type of selector, called the emotional selector. They called the resulting model the emotional behavior tree (EmoBT).

Emotional Selector

- To take emotions into consideration, Johansson and Dell'Acqua extended the definition of behavior trees and introduced a new type of selector, called the emotional selector. They called the resulting model the emotional behavior tree (EmoBT).
- **Emotional Selector:** reorders its child nodes according to a number of identified relevant factors and the affective state of the agent. Once the ordering has been established based upon the probabilities of nodes, the emotional selector behaves as a priority selector. When it completes its execution, and re-execute, the ordering of the nodes must be re-calculated. An emotional selector is represented with a gray circle with the character 'E' in it.

VRET Companions

- VRET-Companion is a virtual character that play the role of user companion in a virtual reality settings. This character try to approach the user (represented as another character) and interact with her/him.

Modelling VRET-Companion Behavior via EmoBTs

- We will consider three relevant aspects for our application; Risk, Time and Planning. In the next slides, we show how to incorporate these three aspects into EmoBTs by following methodological steps :

Modelling VRET-Companion Behavior via EmoBTs

- Objectives: To model VRET-Companion character.

Modelling VRET-Companion Behavior via EmoBTs

- Objectives: To model VRET-Companion character.
- Relevant aspects R = Risk, Time, Planning. Risk perception: The perceived risk of an action is greatly influenced by emotions.

Modelling VRET-Companion Behavior via EmoBTs

- Objectives: To model VRET-Companion character.
- Relevant aspects R = Risk, Time, Planning. Risk perception: The perceived risk of an action is greatly influenced by emotions.
- E = fear, fatigue, sadness. Only three emotions were identified for simplicitly of exposition.

Modelling VRET-Companion Behavior via EmoBTs

- Objectives: To model VRET-Companion character.
- Relevant aspects R = Risk, Time, Planning. Risk perception: The perceived risk of an action is greatly influenced by emotions.
- E = fear, fatigue, sadness. Only three emotions were identified for simplicity of exposition.
- Definition of Risk-value (Risk Assessment): Risk has to do with how dangerous the character believes a situation is. A risk value is between 0 and 1; 0 being no risk at all, and 1 being extremely dangerous. The risk value measures the probability of risk. EmoBTs cannot reason about the risk of performing an action, but we allow the designer to add a risk value to each leaf node in the tree, and derive the associated risk for the inner nodes. Action: An action has a risk value that is set by the designer (0 by default). Condition: A condition has a risk value that is set by the designer (0 by default).

Modelling VRET-Companion Behavior via EmoBTs

- **Sequence Selector**: Since a sequence selector performs every child node of the sequence, the risks of every child nodes must be combined. The overall risk value is calculated as:

$$Risk_j = 1 - \prod_{i=1}^{N} (1 - Risk_i)$$

where N is the number of child nodes of j.

Modelling VRET-Companion Behavior via EmoBTs

- **Priority Selector**: A priority selector j only executes one of its child nodes. Since we cannot determine in advance which node will be executed, we define the risk value of j as the average of the risk of every child node i:

$$Risk_j = \frac{\sum_{i=1}^{N}(1 - Risk_i)}{N}$$

Modelling VRET-Companion Behavior via EmoBTs

- **Parallel Node**: Since all of the child nodes of a parallel node j are executed, the risk is defined as:

$$Risk_j = 1 - \prod_{i=1}^{N} (1 - Risk_i)$$

where N is the number of child nodes of j.

Modelling VRET-Companion Behavior via EmoBTs

- **Parallel Node**: Since all of the child nodes of a parallel node j are executed, the risk is defined as:

$$Risk_j = 1 - \prod_{i=1}^{N} (1 - Risk_i)$$

where N is the number of child nodes of j.

- **Decorator**: The risk value of a decorator is the same as the one of its child node.

Modelling VRET-Companion Behavior via EmoBTs

- To mimic how affective states influence decision making, **emotional weights** was introduced for every relevant factor. Below we show the Risk factor. Let $e_1^+, \ldots, e_M^+$ (resp. $e_1^-, \ldots, e_N^-$) be the values of the emotions that positively (resp., negatively) affect the perception of risk. We define the emotional weight for risk as:

$$E_{Risk} = \frac{\sum_{i=1}^{M} e_i^+}{M} - \frac{\sum_{j=1}^{N} e_j^-}{N}$$

Modelling VRET-Companion Behavior via EmoBTs

- For every aspect $r \in R$ we define the weight $W_{r,i}$ of every child node i of any emotional selector. We consider the *Risk* aspect. The weight for risk for a child node i is calculated as:

$$W_{Risk,i} = (1 - E_{Risk} \times \delta) \times Risk_i$$

where $Risk_i$ is the risk value for the child node i. Note that $W_{Risk,i}$ should be clamped to the interval [0; 1] since it represents a probability. The variable δ determines how much emotions affect the weights. Its value must be between 0 and 1, where 0 signifies no emotional impact and 1 corresponds to full emotional impact. The weight for time is calculated in the following way:

$$W_{time,i} = (1 - \frac{1}{1 + \mu \times time} \times max((1 - \lambda + \lambda \times E_{time}), 0)$$

where μ is a variable that is set to a value that fits the time span used in the simulation.

Modelling VRET-Companion Behavior via EmoBTs

- time is the emotional effect delay; time calculated as:

$$time = L_i + \frac{U_i - L_i}{2} \times ((1 - \sigma \times E_{opt})$$

where E_{opt} is the emotional impact on optimism. The weight for the planning is calculated as:

$$W_{plan,i} = (1 - \frac{1}{1 + \omega \times plan_i} \times max((1 - \phi + \phi \times E_{plan}), 0)$$

where ω is to fit the planning amount of the simulation.

Modelling VRET-Companion Behavior via EmoBTs

- time is the emotional effect delay; time calculated as:

$$time = L_i + \frac{U_i - L_i}{2} \times ((1 - \sigma \times E_{opt})$$

where E_{opt} is the emotional impact on optimism. The weight for the planning is calculated as:

$$W_{plan,i} = (1 - \frac{1}{1 + \omega \times plan_i} \times max((1 - \phi + \phi \times E_{plan}), 0)$$

where ω is to fit the planning amount of the simulation.

- The overall weight of a child node i is calculated as:

$$W_i = \alpha \times W_{Risk,i} + \beta \times W_{Time,i} + \gamma \times W_{Plan,i}$$

The constants α, β and γ give importance to their respective factors. To select which child node to execute, we list them in ascending order according to the weight value W_i. Hence, the lower value of W_i, the more desirable is the node.

VRET-Companion Behavior

- There are different ways in which the the VRET-character might interact with the user character.

VRET-Companion Behavior

- There are different ways in which the the VRET-character might interact with the user character.
- Here we would like to show how the emotional selector can be used to let the VRET-Companion choose the behavior which is most suitable under the current emotional circumstances.

VRET-Companion Behavior

- There are different ways in which the the VRET-character might interact with the user character.
- Here we would like to show how the emotional selector can be used to let the VRET-Companion choose the behavior which is most suitable under the current emotional circumstances.
- We design a simple interaction scenario where the character has the following simple interaction choices: it can simply greet the user 'say hi', then it can go away; it can check the weather outside, if there is sun, comments the weather; it can decide to start a conversation, or even play music and start to dance encouraging the user to mimic the dance movements. The character should lay down to rest when its energy is low; and it should maybe go around looking for users.

VRET-Companion Behavior

- There are different ways in which the the VRET-character might interact with the user character.
- Here we would like to show how the emotional selector can be used to let the VRET-Companion choose the behavior which is most suitable under the current emotional circumstances.
- We design a simple interaction scenario where the character has the following simple interaction choices: it can simply greet the user 'say hi', then it can go away; it can check the weather outside, if there is sun, comments the weather; it can decide to start a conversation, or even play music and start to dance encouraging the user to mimic the dance movements. The character should lay down to rest when its energy is low; and it should maybe go around looking for users.
- The emotional behavior tree used for the example is depicted in the following slide

VRET-Companion Behavior

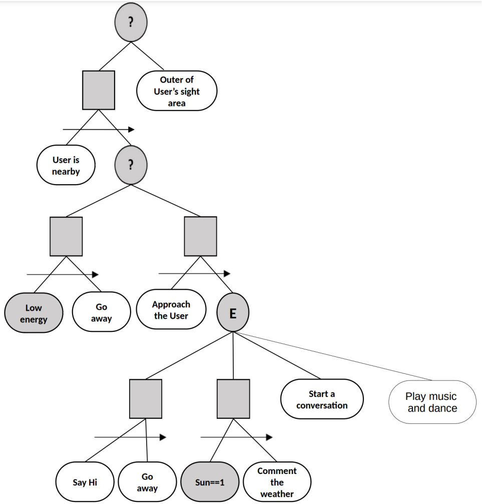


Figure: The behavior tree for the VRET Companion

VRET-Companion Behavior

- In the tree there is on emotional selector with four child nodes (two sequence selectors s1, s2, and two simple action nodes n3, and n4). This emotional selector contains the set of interaction options the character has when it decides to approach the user. The first child node contains a sequence of two actions: to say hi then go away.

VRET-Companion Behavior

	Say hi	Go away	sun==1	Comment weather	Start a conversation	Play music and dance
Time	[0,0]	[0.3,1.0]	[0,0]	[1.5,2.5]	[7,10]	[2,5]
Risk	0.033	0.0	0.0	0.0	0.7	0.6
Plan	1	1	0	1	1.7	1.9

Figure: Time, risk and planning values of leaf nodes

VRET-Companion Behavior

- The figure in the previous slide shows the amount of risk, planning, and time intervals for every child node of the emotional selector.

VRET-Companion Behavior

- The figure in the previous slide shows the amount of risk, planning, and time intervals for every child node of the emotional selector.
- For this scenario we consider the emotions: fear, sadness, and fatigue. Constants $\alpha, \beta, and \gamma$ are set to 1.0. And constants $\mu, \lambda, \sigma, \delta, \omega, and \phi$ are set to 0.8, 0.9, 0.5, 0.6, and 0.6 respectively. We use fear as negative emotional impact for risk with a static value of 1.0 as balance since we do not include any positive emotional impact. For planning, we use fatigue as a positive emotional impact. For time, we use sadness as positive emotional impact. For this example we let E_{opt} be zero.

VRET-Companion Behavior

- We let the specified emotions take different values to illustrate the effect this has on the action selection. In Figures in the next slides, we list the overall weights for different factors given different emotional states. It can be easily seen that emotions greatly affect the factor weights in different ways, resulting in different overall weight for the child nodes.

VRET-Companion Behavior

Fear				
	W_{risk}	W_{time}	W_{plan}	W
s1	0.0627	0.034	0.22	0.3167
s2	0	0.0615	0.15	0.2115
n3	1.33	0.087	0.152	1.569
n4	1.14	0.0737	0.213	1.3

Figure: Weights For S1, S2, n3 and n4 when the value of fear is 1.0 and the value of the remaining emotions is 0.0.

VRET-Companion Behavior

Sadness				
	W_{risk}	W_{time}	W_{plan}	W
s1	0.033	0.34	0.22	0.593
s2	0	0.615	0.15	0.765
n3	0.7	0.87	0.152	1.722
n4	0.6	0.737	0.213	1.55

Figure: Weights For S1, S2, n3 and n4 when the value of sadness is 1.0 and The VALUE OF THE remaining emotions is 0.0.

VRET-Companion Behavior

Fatigue				
	W_{risk}	W_{time}	W_{plan}	W
s1	0.033	0.034	0.55	0.617
s2	0	0.0615	0.375	0.437
n3	0.7	0.087	0.38	1.167
n4	0.6	0.0737	0.533	1.2067

Figure: Weights For S1, S2, n3 and n4 when the value of fatigue is 1.0 and the value of the remaining emotions is 0.0.

VRET-Companion Behavior

- The VRET companion example above is simulated under different emotional states. In the figure in the next slide, the weight values for each action are shown under different emotional conditions. It can be seen that the weight values change widely due to emotional impact. For example, when the character is afraid, s2 is the most desirable choice because it is not risky. When the character is sad s1 is the most suitable one because it takes much shorter time to execute. Finally when the character is tired, then s2 is selected since it consists of one action that needs little planning.

VRET-Companion Behavior

- The VRET companion example above is simulated under different emotional states. In the figure in the next slide, the weight values for each action are shown under different emotional conditions. It can be seen that the weight values change widely due to emotional impact. For example, when the character is afraid, s2 is the most desirable choice because it is not risky. When the character is sad s1 is the most suitable one because it takes much shorter time to execute. Finally when the character is tired, then s2 is selected since it consists of one action that needs little planning.
- It is possible to manipulate the order of the execution of the child nodes to force the character to choose a less desirable node (action) be assigning probabilities to child nodes.

VRET-Companion Behavior

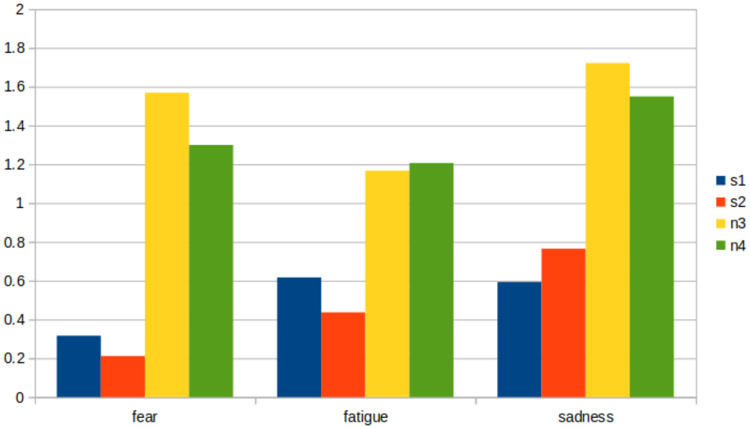


Figure: The weight values for the VRET companion example, given different emotional states. When listed, each emotion has the maximum value 1.0

Conclusion

- In this work we outlined our line of work on emotional human-automation interaction, with the intention of modeling realistic, believable characters.
- we proposed a module for managing emotions in human-AI interaction, to be potentially incorporated in any agent architecture.
- Research on computational modeling of empathy has shown that empathic capacity in interactive agents lead to more trust, help coping with stress and frustration and increase engagement.
- Equipping artificial social agents with empathic capabilities is, therefore, a crucial and yet challenging problem.

Contacts

Thanks

Thank you very much for listening!

Web Page

https://www.disim.univaq.it/AbeerDyoub
https://www.abeerdyoub.com/

Email:

abeer.dyoub@univaq.it