PETEEI: A PET with Evolving Emotional Intelligence

Magy Seif El-Nasr
Texas A&M University
Computer Science Department
College Station, TX 77844-3112
magys@cs.tamu.edu

Thomas R. Ioerger
Texas A&M University
Computer Science Department
College Station, TX 77844-3112
ioerger@cs.tamu.edu

John Yen
Texas A&M University
Computer Science Department
College Station, TX 77844-3112
yen@cs.tamu.edu

ABSTRACT

The emergence of what is now called 'emotional intelligence' has revealed yet another aspect of human intelligence. Emotions have been shown to have a major impact on many of our everyday functions, including decision-making, planning, communication, and behavior. AI researchers have recently acknowledged this major role that emotions play, and thus have began to incorporate models for simulating emotions into agents. However, the emotional process is not a simple process; it is often linked with many other processes, one of which is learning. It has long been emphasized in psychology that memory and experience help shape the dynamic nature of the emotional process. In this paper, we introduce PETEEI (a PET with Evolving Emotional Intelligence). PETEEI is based on a fuzzy logic model for simulating emotions in agents, with a particular emphasis on incorporating various learning mechanisms so that an agent can adapt its emotions according to its own experience. Additionally, PETEEI is designed to recognize and cope with the various moods and emotional responses of its owner. A user evaluation experiment indicated that simulating the dynamic emotional process through learning provides a significantly more believable agent.

1. INTRODUCTION

Artificial Intelligence (AI) and psychology researchers have long been concerned with defining intelligence and finding a way to simulate it. Many theories of intelligence have been formulated, however, very few included emotions. In Howard Gardner's book, *Frames of the Mind*, he describes the concept of Multiple Intelligence [9]. He divided intelligence into six types: linguistic, musical, logical-mathematical, spatial, bodily kinesthetic and personal intelligence. Accordingly, a person's intelligence can vary along these dimensions. Gardner's theory is important because, by including personal intelligence, it incorporated the social and emotional capabilities that people possess, which then led to the rise of what is called the 'emotional intelligence' theory [10]. The importance of emotions in the theory of human intelligence has recently been strengthened through neurological evidence presented by Damasio [4].

As a result, many researchers within the agents and AI field have began to develop computational models of emotions. Simulating emotional intelligence in certain types of computer programs is important. Computational models of emotions are very useful to many applications, including personal assistance applications, training simulations, intelligent interfaces, and entertainment applications.

Recognizing the importance of the emotional process in the simulation of life-like characters [6], a number of computational models of emotions have been proposed within the agent's community [33, 23]. An important effort in this area is the OZ project [2, 25] at CMU. The OZ project simulates believable emotional and social agents; each agent initially has preset attitudes towards certain objects in the environment. Furthermore, each agent has some initial goals and a set of strategies that it can follow to achieve each goal. The agent perceives an event in the environment. This event is then evaluated according to the agent's goals, standards and attitudes. After an event is evaluated, special rules are used to produce an emotion with a specific intensity. These rules are based on Ortony et al.'s event-appraisal model [21]. The emotions triggered are then mapped, according to their intensity, to a specific behavior. This behavior is expressed in the form of text or animation [25].

Even though OZ's model of emotions has many interesting features, it has some limitations too. An emotion is triggered only when an event affects the agent's standards, attitudes or the success of its goals. The notion of partial success and failure is missing, since typically an event does not cause a goal to fully succeed or fully fail. Additionally, some emotions, such as hope and relief, may never be simulated since they rely on expectations.

In order for an agent to simulate a believable emotional experience, it will have to dynamically change its emotions through its own experience. We have developed a new model for an agent that can simulate this dynamic nature of emotions through learning. In this paper, we will present our model and discuss how different learning mechanisms are used to simulate the dynamic aspect of the emotional process. Additionally, we will report the results of some user evaluation experiments of a system we implemented with this model, called PETEEI.

2. COMPUTATIONAL EMOTION

In introducing our model, we will first describe the different learning mechanisms that were implemented in the model. In light of the learning mechanisms involved, we will describe how they influence or facilitate the emotional process. In this way the reader will better understand the specifics of the model.

2.1 Learning

We simulated four different learning mechanisms, which include (1) learning about event sequences and possible rewards, (2) learning about the user's actions and moods, (3) learning about which actions are pleasing to the user and which actions are not, and (4) the Pavlovian S-R conditioning. These learning mechanisms will be detailed in the next paragraphs, giving examples of their effects on the emotional process when appropriate.

2.1.1 Learning about Events

The agent needs to know what events to expect, how likely they are to occur and how bad or good they are. This information is crucial for creating emotions. Consider the following: You are a student in a classroom. Your professor is giving out exam grades. You know that you did really bad, so you are expecting a bad grade. Your professor gives you your exam back, and surprisingly, you got a very good grade. Now, let's look at your emotional state. Before you got the exam, you felt some fear, because your goal of passing the class was threatened by the expectation of a bad grade. After you got the exam back, you realized that you got a good grade, and therefore your fear gradually changes to relief. As you can see, expectation played a very important role here in creating and further revising your emotions with the events that are occurring around you.

The example discussed above was described in a very simplistic way. As a matter of fact, many factors are influencing your emotional states. When an event occurs or when you think an event is about to occur, a *desirability measure* is calculated relative to a specific event [21]. This desirability measure will typically depend on the event's impact on a set of goals. It is often the case that a given event does not have any impact on any specific goal directly, but some sequence of events may eventually have an impact on some goals. Thus, in order to measure the desirability of a specific event, we will need to identify the link between an event or a sequence of events and the corresponding goals, which has been noted to be a very complex task [24]. The agent can potentially learn this link or sequence of links by using a reinforcement learning algorithm [11, 19]; we will briefly outline one reinforcement algorithm, namely the Q-learning algorithm. The reader is referred back to [11, 19] for more details.

It is often the case that an agent does not know the consequences of a given action until a complete sequence of actions ends. The agent, therefore, faces the problem of temporal credit assignment, which is defined as determining which of the actions in its sequence are responsible for producing the eventual rewards. To illustrate the solution that reinforcement learning offers to this problem, we will look at reinforcement learning in more detail. The agent represents the problem space using a table of Q-values in which each entry corresponds to a state-action pair. Optimal Q-values are supposed to represent the expected payoff in the long run if a specific action is taken in a specific state. The table can be initially filled with random values. The agent will begin from a state s. It will take an action, a, which takes it to a new state s'. The agent may obtain a reward, r, for its action [19]. If it receives a reward, it updates the table above using the following formula:

$$Q(s,a) \leftarrow r + \gamma \max_{a'} Q(s',a') \quad [19]$$

where r is the immediate reward, γ is a discount factor, s' is the new state, and a' is an action from the new state s'. Thus, the Q-value of the previous state-action pair depends on the Q-value of the new state-action pair.

This algorithm is guaranteed to converge only for deterministic Markov Decision Processes (MDP)[1]. However, in our application there is a one-to-one alternation between the agent and the user. It is almost impossible to predict the reward given a state and an action, because it may change according to the user's actions and environment. For example, at a state s_0 the agent may do an action x and as a consequence the user gives it a positive reward. At a later time step, the agent is in the same state s_0 so it takes action x again but this time the user rewards it negatively. Therefore, the user introduces a non-deterministic response to the agent's actions. In this case, we treated the reward as a probability distribution over outcomes based on the state-action pair. Figure 1 illustrates this process in more detail. The agent starts off in state s_0 and takes an action a_0 then the user can take either action a_1 that puts the agent in state s_1, or action a_2 that puts the agent in state s_2. The dotted lines show the nondeterminism induced by user's actions, while the straight black lines show the state-action transition as represented in the table. We use another learning model, which will be detailed in the next section, to determine the probability of user's actions.

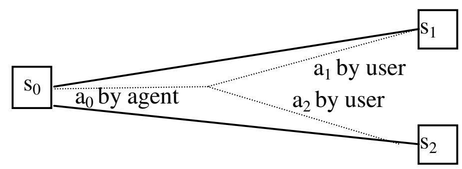

Figure 1. Nondeterministic Rewards

We used a new formula, which was stated in [19] to calculate the Q-value for a non-markov model. The formula is as follows:

$$Q_n(s,a) \leftarrow E[r(s,a)] + \gamma \sum_{s'} P(s' \mid s,a) \max_{a'} Q(s',a')$$

where $P(s' \mid s,a)$ is the probability of reaching a new state s' if action a is taken in state s. $E[r(s,a)]$ is the expected amount of reward [19].

At any given time, the agent will be faced with different actions to take with the possibility of different outcomes and different rewards. The formula described above gives the maximum expected reward given that the agent is at a particular state.

Emotions and moods have a great impact on human decisions and event evaluation [4, 10]. The mood is thus used in the model to guide the expectation values of the next state, s', given that the agent is in a state s. Instead of calculating the Q-value of a state by maximizing the reward, the mood is used as an averaging factor for the new states. As noted in [3], when the agent is in a positive mood it will tend to expect positive events to occur, and otherwise it will expect negative events to occur. Thus, we refined the expectation mechanism to capture this phenomenon. If the agent's mood is positive then the agent will expect desirable events with a β degree more than the undesirable events and vise versa. We modified the equation above to include the β value. Thus, the revised formula is as follows:

$$Q_n(s,a) \leftarrow E[r(s,a)] + \gamma \times \sum_{s'} \begin{cases} match & (1+\beta) P(s' \mid s,a) \max_{a'} Q(s',a') \\ otherwise & P(s' \mid s,a) \max_{a'} Q(s',a') \end{cases}$$

For all the actions that match the mood, the value will be incremented by a β degree, otherwise the previous equation holds.

---

[1] A Markov Decision Process refers to an environment in which rewards and transition probabilities for each action in each state depend only on that state, and are independent of the previous actions and states.

We will take an example to illustrate this idea. For example, the agent is trying to decide between two actions. These actions are illustrated in Figure 2. If he plays with the ball then there is a 70% chance that it will end up in state, s_2, which has a max Q-value of 2.5, but there is also a 30% chance of ending up in state, s_1, which has a max Q-value of -1.5. While if it ignores the ball then there is an equal chance of getting to state, s_3, which has a max Q-value of -3, or state, s_4, which has a max Q-value of 5. Thus, if we use regular probability calculation, action *Ignore(Ball)* will have a value of 1 and *playWith(Ball)* has a value of 1.3, so the agent will play with the ball. However, if we take the mood into account then the calculation will be different, because the agent will no longer expect all outcomes equally. So if for example, the agent is in a good mood, he will expect positive rewards with a degree β more than the negative rewards. If we set β to 50%, then *PlayWith(Ball)* will have a value of 2.7 and *Ignore(ball)* will have a value of 3. By considering the mood, the agent will tend to favor *IgnoreBall(Ball)* over *PlayWith(ball)*, which was considered more desirable using probability calculation.

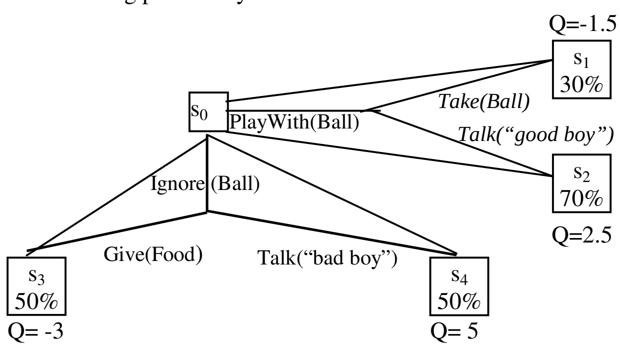

Figure 2. An Example of Reinforcement Learning

2.1.2 Learning about User

Since the agent is interacting with the user, the agent will have to learn about the user's patterns of actions. In our model, we develop a probabilistic model of the user by keeping track of sequences of actions that the user takes, which we call patterns. We focused on patterns of length three, i.e. the user did action a_1, then action, a_2, and finally action, a_3. This concept can be further illustrated using the pet and the owner example. Suppose the owner goes into the kitchen *(a_1)*, takes out the pet's food from the cupboard *(a_2)* and feeds the pet *(a_3)*. These three consecutive actions led to the pet being fed. Thus, if the owner goes to the kitchen again, the pet would probably expect to be fed to a certain degree. As it can be seen from the example, learning sequences of actions can be very useful in predicting other people's actions. Considering the simple behavior of a pet we are trying to model, we felt that conditioning probabilities on the prior two actions was sufficient.

To keep track of the different patterns of the user actions, we used a three dimensional table of action counts. For example, if action, a_0, is followed by action, a_1, followed by action, a_2, then the entry *pattern*[a_0, a_1, a_2] represents the count of the action, a_0, followed by the action, a_1, followed by action, a_2. Using this table we can calculate a probability of a given action to occur according to the history of actions that had already occurred. For example, suppose we know that actions, x and y, had occurred at times t_1 and t_2, respectively. If *pattern*[x, y, z]>0, then we calculate the probability of z to occur using the following formula:

$$P(z \mid x, y) = \frac{pattern[x, y, z]}{\sum_i pattern[x, y, i]}$$

Otherwise, if the number of observations is too scarce, then we use the following formula, conditioned on only one prior action:

$$P(z \mid y) = \frac{\sum_i pattern[i, y, z]}{\sum_j \sum_k pattern[j, y, k]}$$

Otherwise, we use the following formula, which is just the prior probability of z:

$$P(z) = \frac{\sum_j \sum_k pattern[j, k, z]}{\sum_j \sum_k \sum_l pattern[j, l, k]}$$

Using this probability, the agent can calculate what actions to expect from the user, and it can further calculate a desirability measure using the Q-table discussed in Section 2.1. Using the desirability and expectation measures, the agent will be able to create emotions, as we will discuss later.

2.1.3 Learning about pleasing & displeasing actions

External feedback was used to learn what actions are pleasing or displeasing to the user. In some situations, the user can give the agent an assessment of his actions by saying some remarks, such as 'good boy' or 'shame on you.' Through this learning scheme, the agent will identify the most recent action performed by it as the action that was evaluated by the user. Temporal credit assignment is not a problem here; we assumed that standards or learning about pleasing and displeasing actions may in fact rely on one action rather than a sequence of actions that can be generalized to a one global action. Therefore, we used this learning scheme to build up the values of the agent using the user feedback to refer to most recent action done by the agent.

The agent keeps the user's remarks with the action and situation that preceded them. It can then evaluate its own actions in terms of a function: *Quality(x, s)=v*, where x is an action that happened in a particular situation s, and v is the value given to the action-situation pair. When given feedback, the agent searches its knowledge base for the action-situation pair. If they were not found, a value, v_a, is assigned to the new action-situation pair. However, if the action-situation pair was found and the Quality matches the Quality that was documented then the pair is reinforced (incremented), otherwise the agent becomes confused and decrements the pair.

The utility of this learning mechanism is two folded. Firstly, it creates the baseline by which the agent can set its own standards. These standards are then used to determine emotions like pride or shame. In addition, these standards are used to evaluate the user's actions and trigger emotions, including admiration and reproach. The intensities of these emotions are calculated as a function of the value, v, of the learnt rule. Secondly, at times the agent may feel guilty or may want to please the user to get something. This technique is widely used in relationships, whenever a partner feels guilty, he/she goes under a submissive role; he/she will go out of his/her way to do something for the other partner to make up for what he/she did wrong [8].

2.1.4 Pavlovian Conditioning

Associating an object directly with an emotion forms yet another type of learning. For example, if the agent experienced pain when an object, *g*, touches it, then an emotion of pain will be associated with the object, *g*. This kind of learning does not depend on the situation per se, but it depends on the object-emotion association. This is especially true in the taste aversion paradigms of Pavlovian conditioning, where a rat would associate sweet with poison after some trials where the rats were injected with poison subsequent to eating sweets [5].

Each of these associations will have an accumulator, which is incremented by the repetition and intensity of the object-emotion occurrence. This type of learning will provide the agent with another type of expectation triggered by the object rather than the event. Using the count and the intensity of the emotion triggered, the agent can calculate the expected intensity of the emotion. We used the formula shown below:

$$\langle I(e)|o\rangle = \frac{\sum_{events(i)} I_i(e)}{n_o}$$

where *events(i)* are events that involve the object *o*, $I_i(e)$ is the intensity of emotion *e* in event *i*, and n_o is the total number of events involving the object *o*. In essence, the formula is averaging the intensity of the emotion in the events where the object, *o*, was introduced.

2.2 Simulation of Emotion

So far we have described how learning occurs in the agent's model. Although, we have given some examples on how the different learning mechanisms could potentially produce emotions, we did not describe the process of emotion simulation in detail. We will now attempt to detail the emotional process and link it back to the learning process when applicable.

Figure 3 depicts the emotional process and it outlines where the four learning mechanisms are used within the process. The four learning mechanisms are shown in the figure through shaded ovals. The figure also depicts the information passed through the learning and the emotional model through boxes. As the figure shows, learning about events sequences and rewards/costs associated with each sequence is used for two purposes. Firstly, it determines the desirability measure of the sequence of events given their rewards and costs. Secondly, it gives the agent the ability to predict the likelihood of certain events to occur given that a sequence of events has already occurred. Learning user's actions also enables the agent to predict, with a certain probability, the next user action. Thus, if the agent is in situation *s* at time t_0 and it is expecting *n* events to occur and *y* events have already occurred at time t_0. According to the two learning mechanisms mentioned above, the agent will produce a desirability measure of the *y* events and the *n* expected events. These events along with their desirability measures are passed to an emotion generation process, as shown in figure 3. Additionally, the expectation measure (or likelihood of occurrence) of the *n* expected events will have to be passed to the emotion generation process.

Emotion is generated according to the expectation of certain events occurrence and the desirability of those events. We used some rules to simulate the emotion generation process. These rules were taken from [21]. For example, joy was described as the occurrence of a desirable event, and hope was described as the occurrence of an unconfirmed desirable event. Therefore, as it can be seen, the emotion of hope will use the expectation and desirability measure. Hope's intensity is then calculated as a function of these measures. On the other hand, joy is calculated using the desirability measure alone, but its intensity is calculated as a function of the desirability and the expectation of this event at the earlier time step.

Additionally, emotions can also be generated from the Pavlovian conditioning, where a single event or an object is linked in memory with a single emotion, and thus, if the object or event is presented to the agent, the agent will tend to automatically trigger the emotion associated with that object or event.

2.3 Agent's Behavior

The agent's behavior depends on its emotional, motivational states and the current situation or the event perceived. We used fuzzy rules to relate sets of emotions to corresponding behaviors. For example, consider the following rule:

if	Anger is High
	AND Taken-away(dish)
THEN	behavior is Bark-At(user)

The behavior, Bark-At(user), depends on what the user did and the emotional intensity of the agent. If the user did not take the dish away and the agent was angry for some other

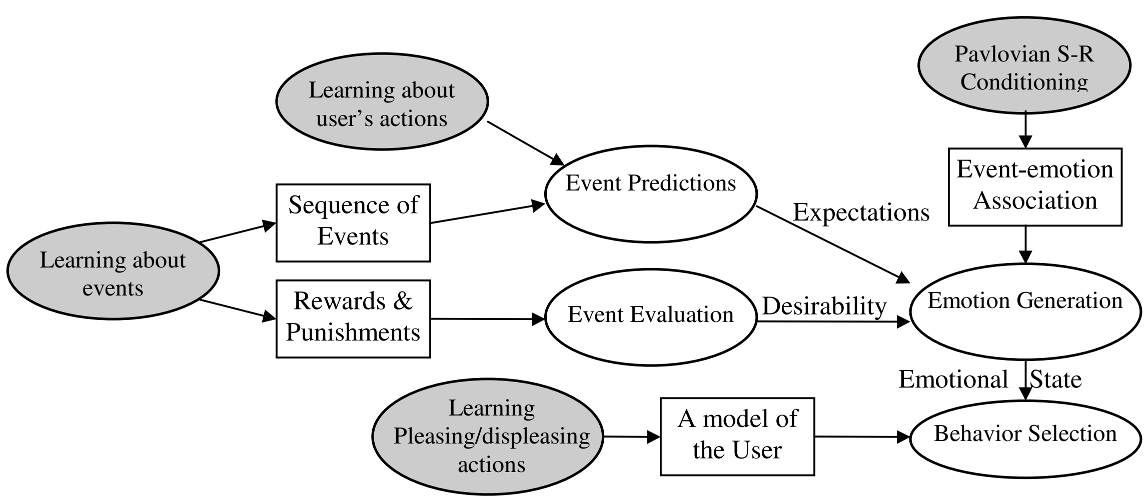

Figure 3. The Emotional/Learning Processes

reason, it would not be inclined to bark at the user (although it might in some situations), because the user might not be the cause of its anger. Thus, it is important to identify both the event and the emotion. It is equally important to identify the cause of the event. In this case, we are assuming that non-environmental events such as taken-away(dish), throw(ball), etc. are all caused by the user.

Furthermore, the agent's behavior is not solely determined by its emotional state, but it also depends on some predefined social norms that defines and limits the expressiveness of its behavior [1, 27]. For example, if the user is in a very bad mood and the user is considered a good friend, the agent will not tend to be very aggressive in its behavior. Hence, if its emotional state is dominated by anger, and the rules described above suggests an aggressive behavior, this anger and aggressive behavior will be suppressed. Moreover, if the user is perceived as happy and the agent needs food or drink, then the agent will go into a submissive role and will act on actions that it learned to be pleasing to the user, as previously mentioned.

These rules of behavioral or social norms are preset. Typically, these preset behavioral rules will tend to shape the personality of the agent. Therefore, by incorporating personality and self-presentation, the agent will tend to change these preset rules. Nevertheless, we kept this issue as a future enhancement.

3. SIMULATION AND RESULTS

In this section, we will describe how the model, presented in the previous section, was actually implemented. We chose to model emotions in a pet, because pets are simpler than humans. They do not require sophisticated planning and their goal structure is much simpler than humans'. Furthermore, to evaluate the believability of an agent's behavior, one will have to appeal to users' judgments, and thus it would be better to model a pet rather than some other creature, because most people have some expectations about what a pet can or cannot do.

PETEEI was designed to have the look and feel of a typical Role Playing Game interface. It has five major scenes: a garden, a bedroom, a kitchen, a wardrobe and a living room. The garden scene is illustrated in Figure 4. The user can interact with the pet through various actions, including (1) introducing objects to the pet's environment, (2) taking objects away from the pet's environment, (3) hitting objects, including the pet, and (4) talking aloud.

Section 2 emphasized the role of learning in building the dynamic nature of emotional responses. To evaluate this claim, we built three models. In the first model, PETEEI was configured to produce a random set of emotions and random behaviors. This experiment was essential to provide a baseline for the other experiments. Sometimes users could be impressed by the graphics and sound; and the fact that the pet in the pictures reacted to the user's actions might prompt some users to answer some questions with positive feedback. A second model was implemented where PETEEI simulated emotions with no learning. This model was implemented by using constant expectations. Thus, emotions were triggered using the desirability degree which depends on the preset goals and standards of the agent. Even though this model is sufficient to invoke some emotions, such as anger, sadness or joy, the model cannot generate emotions such as fear or hope. Additionally, the agent's emotional state in this model will vary only according to the situation and time spectrum. On the other hand, by adding the learning model to the agent's emotional process, the emotional states of the agents will be changing according to the situation, time and experience. We thus implemented a third version where PETEEI simulated emotional responses using the learning mechanisms discussed in Section 2. Model 2 and 3 were used to assess the importance of learning in the believability of the emotional responses.

Twenty-one users were recruited to evaluate the model and the hypothesis involved. The users were chosen from the first year undergraduate class. This choice was made to decrease the bias inflicted by specialized background knowledge. Users met with the principle investigator for a period of two and a half-hours. During this time, they ran the different simulations using ten scripted scenarios and answered a questionnaire.

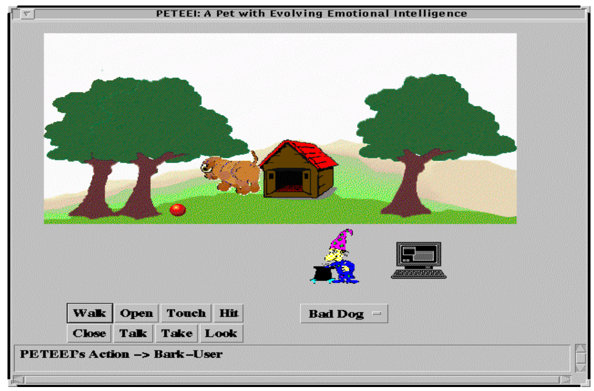

Figure 4. Model's Interface

To explore how the users perceived the learning component in each model, four questions were presented. Question A asked the user to rate PETEEI's learning about the user, question B asked the user to rate PETEEI's learning about the environment, question C asked the user to rate PETEEI's learning about good or bad actions, and finally question D asked the user to rate PETEEI's overall learning ability. The users were asked to answer the questions and rate PETEEI's abilities in a scale from 1 to 10, where 1 is worst and 10 is best. The questions were as follows:

A. A subject *x* can learn about the other people's behavior to know what to expect and from whom. Do you think PETEEI learns about you?

B. A subject could learn more about the environment to plan for his/her actions. Do you think that PETEEI learns about its environment?

C. A subject could learn how to evaluate some actions as good or bad according to what people say or according to his own beliefs. Do you think that PETEEI learns about good and bad actions?

D. Overall how would you rate PETEEI's learning using the four criteria listed above?

Answers were collected for the questions above and an average was calculated for each question, shown in the Table 1 as the mean. We also calculated the confidence interval for a 95% confidence with a sample size of 21. This measure was calculated using the following formula:

$$\bar{x} \pm 1.96(\frac{\sigma}{\sqrt{n}})$$

where $\bar{x}$ is the mean, the σ is the variance and *n* is the size of the sample which is 21 in this case. The confidence interval is shown in the table. As it can be seen, the numbers for all the models without the learning component, namely models 1 and 2, were between zero and two. There were some differences that can be noted from the table among these two models. However, the difference is small and the confidence levels overlap greatly, leading us to believe that users did not really perceive learning in these models. In contrast, the learning model (model 3) was observed to significantly increase the learning perceived in all questions.

Table 1. Learning Ratings

Model	Question A		Question B	
	mean	**interval**	**mean**	**interval**
1	0	0-0	0	0-0
2	0.5	-0.1-1.1	0.3	-0.2-0.9
3	7.7	7.3-8.1	5.95	4.7-7.1
Model	**Question C**		**Overall**	
	mean	**interval**	**mean**	**interval**
1	0.1	-0.1-0.3	0.238	0.051-0.425
2	0.4	-0.2-1	0.91	0.44-1.37
3	7.67	7.25-8.08	7.952	7.6-8.3

The users were asked to rate the behavior of the pet in terms of how convincing it is. The users were told not to rate it in terms of animation or facial expression, but to concentrate on the behavioral and emotional aspect. We analyzed users' answers. The results are shown in Table 2.

Table 2. Behavior Ratings

Model	Behavior	
	mean	**Interval**
1	1	0.67 - 1.33
2	4.33	3.60 - 5.07
3	8.095	7.65 - 8.542

As the table shows, the random model (model 1) did not convey a realistic or convincing behavior to the user since the rating was on average one. In contrast, the introduction of the nonlearning model (model 2) improved this measure by about 3.33 units on average. The learning model (model 3) further improved this measure by 2.6 to get an average of 8.1 units.

For a complete review of the protocol and questions asked, the reader is referred to [28]. The results discussed above confirmed our earlier hypothesis that simulating the dynamic emotional process provides a significantly more believable agent.

5. DISCUSSION

A computational model of emotions, such as the one presented here, can be of great benefit to certain types of computer programs. Such a model could potentially improve many training applications, especially when team training is an issue. Training applications have traditionally been concerned with simple tasks, such as maintenance, which normally involve a sequence of simple lessons that are given to a trainee [15]. Clearly, group training is quite different since it relies on many other factors, such as group coherence, group interactions, moral, and dominance within the group [20]. It has been noted that most of these different concepts heavily depend on emotions and experience [20]. Moreover, these training systems can use human-like agents (i.e., synthetic characters) to replace some of the team members. Clark Elliot summarized current efforts in human-like agent simulations, some of which simulated characters with different personalities and different emotions [6]. Therefore, modeling emotions and personality may provide a good starting point for team training applications.

Through many studies in social psychology, emotions, shared knowledge and self-presentation have been acknowledged to be the main facilitators of human-human communication [31, 27, 1]. These studies have inspired many researchers to look for alternative ways to solve the human-computer interaction problem using human-human interaction as a model. Some researchers tried to solve this problem by developing personalized interface agents - agents that learn about the user and try to assist him/her in any way possible [14]. Some others tried to make the interface more appealing to the user and to invoke emotional responses from the user in the process [12]. Research is still open for new ideas on how to use emotions and learning to enhance the interface, and break the barrier between human and computer.

A model of emotions can also be used to simulate emotions in character animation. Role Playing Games (RPG) is one class of character animation in which it is important to simulate emotions and emotional behaviors if characters are to be simulated as agents. An RPG is a game where the user assumes a specific role. The user is normally given a mission according to the story plot of the game. During the game, the user will meet and interact with different characters. Most of the commercially available RPGs use scenario driven characters. To build a scenario-driven character, the character builder will have to anticipate all the possible scenarios for all the possible interactions within the game. It is almost impossible to build a prefect character using this method since character builders might miss a few combinations or users might want to talk to the character when he/she is not supposed to, and therefore, these characters sometimes appear to be unconvincing. On the other hand, if a character is a synthetic agent, the character builder's mission is much simplified. He/she will need to identify the character's goals, behaviors, and personality. Thus, an agent model that can simulate different personalities and emotions may provide a very useful tool in this type of application.

The applications described above are but a few of the many computer simulated-applications that may use emotions in one way or the other. These applications demonstrate the importance of emotional/social agents as computational models or tools.

6. FUTURE WORK & CONCLUSION

The model presented in this paper is mainly an expansion on the many ideas that have been presented in previous research (psychological and computational models). Our evaluation indicated that incorporating various learning mechanisms, such as social learning, Pavolvian conditioning, learning about the companion's patterns of actions, improved the believability of the agent and produced a more dynamic emotional response. Nevertheless, the model still falls short in several aspects.

Psychologists recognized personality to be a factor that influences and is being influenced by the learning and the emotional processes [10]. Furthermore, personality has been recognized by many artists to be one of the most important factors in character animation [32]. Some researchers have added personality to their emotional agent models [13]. However, personality is a very challenging concept to simulate, since there are many traits and individual variations [26]. We believe personality is an important component to be added to our model in the near future.

Human intelligence includes many other phenomenon that have challenged and intrigued psychologists for decades. These concepts include self-perception, self-awareness, self-expansion, self-presentation, and self-esteem [16, 1, 10]. They have been recognized by psychologists to have a major impact on the subject's stability, emotional and social responses [10, 1].

In conclusion, learning helps in capturing the dynamic nature of the emotional process. Our model is just another building block in our attempt to understand real human intelligence, and to make use of its simulation in various applications.

7. ACKNOWLEDGMENTS

This work was partially supported by NSF Young Investigator Award IRI 92-57293.

8. REFERENCES

[1] Aron, A. and Aron N. Self-expansion motivation and including other in the self. In S. Duck (ed.), *Handbook of personal relationships: Theory, research, and interventions (*2nd ed., pp. 251-270). Chichester, England:Wiley, 1997.

[2] Bates, J. The role of emotion in believable agents. *Communications of ACM,* 37:7, 122-125, 1992.

[3] Bower, G. H. and Cohen, P. R. Emotional influences in memory and thinking: Data and theory. In M. Clack and S. Fiske (ed.), *Affect and Cognition* (pp. 291-331). London, England:Lawrence Erlbaum Association Publishers, 1982.

[4] Damasio, A. Descartes' error: Emotion, reason, and the human brain. New York: G. P. Putnam, 1994.

[5] Domjan, M. *The principles of learning and behavior* (4th ed.). Boston: Brooks/Cole Publishing Co. 1998.

[6] Elliot C. and Brzezinki J. Autonomous agents as synthetic character. *AI Magazine,* AAAI press, 13-30, 1998.

[7] Eibl-Eibesfeldt, I. *Human ethology*. New York: Aldine de Gruyter, 1989.

[8] Fiske, A. P. The four elementary forms of sociality: Framework for a unified theory of social relations. *Psychology Review,* 99, 689-723, 1992.

[9] Gardner H. *Frames of Mind.* New York: Basic Books, 1973.

[10] Goleman D. *Emotional Intelligence.* New York: Bantam Books, 1995.

[11] Kaelbling, L. P., Littman, M. L., and A. W. Moore. Reinforcement Learning: A Survey. *Journal of Artificial Intelligence Research,* 4, 237-285, 1996.

[12] Kim, J. and Moon, J. Y. Designing towards emotional usability in customer interfaces - trustworthiness of cyber-banking system interfaces. *Interacting with computer: The interdisciplinary journal of human-computer interaction,* 10, 1998.

[13] Loyall, A. B. and Bates, J. Personality-based believable agents that use language. *Proc. of the first autonomous agents conference,* 1997.

[14] Maes, P. Agents that reduce work and information overload. *Communications of ACM,* 1:7, 37-45, 1997.

[15] Mark, M. A. The VCR tutor: Design and evaluation of an intelligent tutoring system. Master's thesis, University of Saskatchewan, Saskatoon, Saskatchewan, 1991.

[16] McKay, M. and Fanning P. *Self Esteem*. St. Martin's Paperbacks, USA, 1978.

[17] Miller, G. A. The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Review,* 63, 81-97, 1956.

[18] Minsky, M. The society of the mind. New York: Simon and Schuster, 1986.

[19] Mitchell, T. M. Machine Learning. New York: McGraw-Hill Co., 1996.

[20] Nye, J. L. and Brower, A. M. *What's social about social cognition?* London: Sage Publications, 1996.

[21] Ortony, A., Clore, G., and Collins, A. *The cognitive structure of emotions*. Cambridge: Cambridge University Press, 1988.

[22] Pfeifer, R. Artificial Intelligence Models of Emotions. *Cognitive perspective on emotion and motivation,* 287-320, 1988.

[23] Picard, R. W. *Affective Computing*. Cambridge: MIT press, 1997.

[24] Reilly, W. S. and Bates, J. Building emotional agents. Pittsburgh, PA: Carnegie Mellon University, Technical Report CMU-CS-92-143, 1992.

[25] Reilly, W. S. Believable social and emotional agents. School of Computer Science, Pittsburgh, PA: Carnegie Mellon University, Ph.D. thesis, 1996.

[26] Riso, D. R. and Hudson, R. *Personality Types*. USA: Houghton Mifflin, 1996.

[27] Rusbult, C. E. and Arriaga, X. B. Interdependence theory. S. Duck (ed.), *Handbook of personal relationshps: Theory, research, and interventions* (2nd ed., pp. 221-250). Chichester, England:Wiley, 1997.

[28] Seif El-Nasr, M. Modeling emotion dynamics in intelligent agents. Department of Computer Science, College Station, TX: Texas A&M University, Master's thesis, 1998.

[29] Simon, H. Motivational and emotional control of cognition. *Psychological review,* 74, 29-39, 1967.

[30] Simon, H. *The science of the artificial.* Cambridge, MA:MIT Press, 1996.

[31] Suchman, L. *Plans and situated actions.* Cambridge University Press, 1981.

[32] Thomas, F. and Johnston, O. *The illusion of life*. New York: Abbeville Press, 1981.

[33] Velasquez, J. Modeling emotions and other motivations in synthetic agents. *Proc. of the AAAI conference*, 10-15, 1997.