Modeling Self-Efficacy in Intelligent Tutoring Systems: An Inductive Approach

Scott W. McQuiggan[1], Bradford W. Mott[2], and James C. Lester[1]

[1]*Department of Computer Science, North Carolina State University, Raleigh, NC, USA*
swmcquig@ncsu.edu and lester@ncsu.edu[1]

[2]*Emergent Game Technologies, Chapel Hill, NC, USA*
bradford.mott@emergent.net

Abstract. Self-efficacy is an individual's belief about her ability to perform well in a given situation. Because self-efficacious students are effective learners, endowing intelligent tutoring systems with the ability to diagnose self-efficacy could lead to improved pedagogy. Self-efficacy is influenced by (and influences) affective state. Thus, physiological data might be used to predict a student's level of self-efficacy. This article investigates an inductive approach to automatically constructing models of self-efficacy that can be used at runtime to inform pedagogical decisions. It reports on two complementary empirical studies. In the first study, two families of self-efficacy models were induced: a static self-efficacy model, learned solely from pre-test (non-intrusively collected) data, and a dynamic self-efficacy model, learned from both pre-test data as well as runtime physiological data collected with a biofeedback apparatus. In the second empirical study, a similar experimental design was applied to an interactive narrative-centered learning environment. Self-efficacy models were induced from combinations of static and dynamic information including pre-test data, physiological data, and observations of student behavior in the learning environment. The highest performing induced naïve Bayes models correctly classified 85.2% of instances in the first empirical study and 82.1% of instances in the second empirical study. The highest performing decision tree models correctly classified 86.9% of instances in the first study and 87.3% of instances in the second study.

Keywords: *Affective user modeling, affective student modeling, self-efficacy, intelligent tutoring systems, inductive learning, human-computer interaction*

1. Introduction

Affect has begun to play an increasingly important role in intelligent tutoring systems. Recent years have seen the emergence of work on affective student modeling (Conati and Maclaren, 2005), detecting frustration and stress (Burleson and Picard, 2004; Prendinger and Ishizuka, 2005), modeling agents' affective states (André and Mueller, 2003; Gratch and Marsella, 2004; Lester et al., 1999), devising affectively informed models of social interaction (Johnson and Rizzo, 2004; Paiva et al., 2005; Porayska-Pomsta and Pain, 2004), and detecting student motivation (de Vicente and Pain, 2002). All of this work seeks to increase the fidelity with which affective and motivational processes are modeled in intelligent tutoring systems in an effort to increase the effectiveness of tutorial interactions and, ultimately, learning.

Self-efficacy is an affective construct that has been found to be a highly accurate predictor of students' motivational state and their learning effectiveness (Zimmerman, 2000). Defined as "the belief in one's capabilities to organize and execute the courses of action required to manage prospective situations" (Bandura, 1995), self-efficacy has been repeatedly demonstrated to

[1] This paper (or a similar version) is not currently under review by a journal or conference, nor will it be submitted to such within the next three months.

directly influence students' affective, cognitive, and motivational processes (Bandura, 1997). Self-efficacy holds much promise for intelligent tutoring systems (ITSs). Foundational work has begun on using models of self-efficacy for tutorial action selection (Beal and Lee, 2005) and investigating the impact of pedagogical agents on students' self-efficacy (Baylor and Kim, 2004; Kim, 2005). Self-efficacy is useful for predicting which problems and sub-problems a student will select to solve, how long a student will persist on a problem, how much overall effort they will expend, as well as motivational traits such as level of engagement (Schunk and Pajares, 2002; Zimmerman, 2000). If ITSs could increase students' self-efficacy, then students might be more actively involved in learning, expend more effort, and be more persistent; it might also promote student coping behaviors when they experience learning impasses (Bandura, 1997).

To effectively reason about a student's self-efficacy, ITSs need to accurately model self-efficacy. Self-efficacy diagnosis should satisfy three requirements. First, it should be realized in a computational mechanism that operates at runtime. Self-efficacy may vary throughout a learning episode, so pre-learning self-efficacy instruments may or may not be predictive of self-efficacy at specific junctures in a learning episode. Second, self-efficacy diagnosis should be efficient. It should satisfy the real-time demands of interactive learning. Third, self-efficacy diagnosis should avoid interrupting the learning process. A common approach to obtaining information about a student's self-efficacy is directly posing questions to them throughout a learning episode. However, periodic self-reports are disruptive.

This article details the design and evaluation of an empirical approach to computational self-efficacy models. The empirical approach calls for a data-driven framework for modeling self-efficacy. The article proposes SELF (Self-Efficacy Learning Framework), a data-driven affective architecture and methodology for learning empirically informed models of self-efficacy from observation of student interactions. SELF has been evaluated in two experiments that investigate inductive approaches (naïve Bayes classifiers and decision tree classifiers) to constructing models of self-efficacy. In the foundational evaluation students interacted with the online tutorial system in the domain of genetics. In this experiment two families of self-efficacy models were induced: the model learner constructed (1) *static* models, which are based on demographic data and a validated problem-solving self-efficacy instrument (Bandura, 2006), and (2) *dynamic* models, which extend static models by also incorporating real-time physiological data. In the experiment, 33 students provided demographic data and were given an online tutorial in the domain of genetics. Next, they were given a validated problem-solving self-efficacy instrument, and they were outfitted with a biofeedback device that measured heart rate and galvanic skin response. Physiological signals were then monitored while students were tested on concepts presented in the tutorial. After solving each problem, students rated their level of confidence in their response with a "self-efficacy slider." Both families of resulting models, induced from collected data, operate at runtime, are efficient, and do not interrupt the learning process. The static models are able to predict students' real-time levels of self-efficacy with 82.9% accuracy, and the resulting dynamic models are able to achieve 86.9% predictive accuracy. Thus, the predictive power of non-intrusive static models can be increased by further enriching them with physiological data (dynamic models).

The results of the foundational evaluation of SELF-constructed models of self-efficacy in the online tutorial system indicated that an inductive approach offered potential as a method for modeling self-efficacy and called for further investigation of the techniques. The design of a second experiment was motivated by three factors: explicitly controlling the challenge levels of the learning environment; exploiting task structure to study self-efficacy with an appraisal-

theoretic (Lazarus, 1991) framework; and increasing the complexity of the learning environment and, therefore the dimensionality of the induction problem. In the second experiment, dynamic models (including real-time physiological data) of self-efficacy were induced. In the experiment, 42 students provided demographic data and were given an online tutorial in the domain of genetics. Next, they were given a validated problem-solving self-efficacy instrument, and they were outfitted with a biofeedback device that measured heart rate and galvanic skin response. Next students entered CRYSTAL ISLAND, an interactive learning environment in which the student plays the role of detective in a science mystery in the domain of genetics. Students used their recently acquired knowledge of genetics to solve the mystery. They periodically provided self-reports of self-efficacy via popup dialog boxes throughout their interaction. Resulting models are reasonably accurate, operate at runtime, are efficient, and do not interrupt the learning process.

This article is structured as follows. Section 2 discusses the role of self-efficacy in learning. Section 3 presents the SELF architecture and methodology, describing how SELF models of self-efficacy are induced. The foundational evaluation with the online tutorial system is described in Section 4. Section 5 then presents an evaluation of SELF in a rich, interactive narrative-centered learning environment, CRYSTAL ISLAND. Section 6 discusses the findings and associated design implications, and Section 7 offers concluding remarks and suggests directions for future work.

2. Affect and Self-efficacy

Founded in perception and decision-making, affect is a central component of human cognition. Affective reasoning has been the subject of increasing attention among cognitive scientists in recent years, and the study of affective computing is becoming a field in its own right. Affective computing investigates techniques for enabling computers to recognize, model, understand, express, and respond to emotion effectively. Such skills have been recognized as key components of human emotional intelligence essential to natural interaction (Goleman, 1995). Affect influences humans' interactions with one another, their behaviors, and cognitive processes, and it can contribute in important ways to a broad range of computational tasks (Picard, 1997). In particular, incorporating affective reasoning into education, training, and entertainment systems could enable them to create more effective, interesting, and engaging experiences for their users.

2.1. Affect Recognition

The complementary processes of affect synthesis and affect recognition have been studied extensively in the context of virtual environments. Work on affect synthesis has been done to control expressive models of embodied cognition and behavior in animated agents (André and Mueller, 2003; Gratch and Marsella, 2004; Paiva et al., 2005) and pedagogical agents that support emotive expression in intelligent tutoring systems (Johnson and Rizzo, 2004; Porayska-Pomsta and Pain, 2004). *Affect recognition* (Picard, 1997) is the task of identifying the affective state of an individual from a variety of physical cues, which are produced in response to affective changes in the individual. These include visually observable cues such as body and head posture, facial expressions, and posture, and changes in physiological signals such as heart rate, skin conductivity, temperature, and respiration (Allanson and Fairclough, 2004; Frijda, 1986). Psychologists have used electroencephalograms (EEG) to monitor users' brain activity for detection of task engagement (Pope et al., 1995) and user attention (Mekeig and Inlow, 1993),

electromyograms (EMG) to detect electrical activity in muscles to obtain measurements of users' sense of presence in virtual environments (Weiderhold et al., 2003), and eye tracking devices to measure pupil responses to emotional stimulations (Partala and Surakka, 2003). Heart rate measurements have been used to adapt challenge levels in computer games (Gilleade and Allanson, 2003), detect frustration and stress (Prendinger et al., 2005), and monitor anxiety and stress (Healey, 2000). Galvanic skin response (GSR) has been correlated with cognitive load (Verwey and Veltman, 1996) and used to sense user affective states, such as stress (Healey, 2000), student frustration for learning companion adaptation (Burleson, 2006), frustration for life-like character adaptation in a mathematical game (Prendinger et al., 2005), and multiple user emotions in an educational game (Conati, 2002). Heart rate and GSR have jointly been used to determine user affect (Prendinger and Ishizuka, 2005) based on the model of Lang (1995), which characterizes emotions in a two-dimensional space of valence (positive to negative) and arousal (low to high).

Affect recognition work has explored emotion classification from self reports (Beal and Lee, 2005), post-hoc reports (de Vicente and Pain, 2002), self-reports, peer reports, and judges' reports trained to recognize emotion in the face based on the work of Ekman and Friesen (1978) (Graesser et al., 2006), posture (Mota and Picard, 2003), and multimodal classifications including combinations of visual cues and physiological signals (Burleson, 2006; Burleson and Picard, 2004; Picard et al., 2001), and facial and head gestures, posture, and task information (Kapoor and Picard, 2005). Recent investigations have also begun to investigate linguistic features for prediction of affective states (Litman and Forbes-Riley, 2006) and comprehensive world models for predicting user physiological response to reduce the need for biofeedback apparatus in runtime environments (McQuiggan et al., 2006). Collectively, this body of work serves as a springboard for research described in this article, which, in part, reports on the use of measurements of user physiological response as a predictor of self-efficacy levels. Because users' physiological responses follow directly from their affective states, which are known to be correlated with levels of self-efficacy (Zimmerman, 2000), accurate measurements of physiological response could be used to enable interactive environments to effectively predict user levels of self-efficacy in order to guide customized interactions.

2.2. Self-efficacy

Self-efficacy[2] influences students' reasoning, their level of effort, their persistence, and how they feel; it shapes how they make choices, how much resilience they exhibit when confronted with failure, and what level of success they are likely to achieve (Bandura, 1995; Schunk and Pajares, 2002; Zimmerman, 2000). While it has not been conclusively demonstrated, many conjecture that given two students of equal abilities, the one with higher self-efficacy is more likely to perform better than the other over time. Highly efficacious students exhibit more control over their future through their actions, thinking, and feelings than inefficacious students (Bandura, 1986). Self-efficacy is intimately related to motivation, which controls the effort and persistence with which a student approaches a task (Lepper et al., 1993). Effort and persistence are themselves influenced by the belief the student has that she will be able to achieve a desired outcome (Bandura, 1997). Students with low self-efficacy perceive tasks to be more challenging

[2] Self-efficacy is closely related to the popular notion of confidence. To distinguish them, consider the situation in which a student is very confident that she will fail at a given task. This represents high confidence but low self-efficacy, i.e., she is exhibiting a strong belief in her inability (Bandura, 1997).

than they actually are, often leading to feelings of anxiety, frustration and stress (Bandura, 1986). In contrast, students with high self-efficacy view challenge as a motivator (Bandura, 1986; Malone and Lepper, 1987). Self-efficacy has been studied in many domains with significant work having been done in computer literacy (Delcourt and Kinzie, 1993) and mathematics education (Pajares and Kranzler, 1995). It is widely believed that self-efficacy is domain-specific; whether it crosses domains remains an open question. For instance, students with high self-efficacy in mathematics may be inefficacious in science, or a highly efficacious student in geometry may experience low efficacy in algebra.

A student's self-efficacy is influenced by four types of effectors (Bandura, 1997; Zimmerman, 2000). First, in *enactive mastery experiences*, the student performs actions and experiences outcomes directly. These are typically considered the most influential category as successful experiences typically raise self-efficacy, while unsuccessful experiences tend to lower self-efficacy. However, easy successes often encourage expectations of quick successes leading to a reduction in student resilience when faced with challenges. Second, in *vicarious experiences*, the student models her beliefs based on comparisons with others. These can include peers, tutors, and teachers, especially those with similarly perceived capabilities. Thus, seeing a perceived parallel peer succeed at a task typically increases self-efficacy. Vicarious experiences are particularly useful when the only way to gauge adequacy is to relate personal results with the performance of others. For instance, a student who completes a timed math test in 53 seconds has to gauge her performance by comparing completion times of her peers. Third, in *verbal persuasion*, the student experiences an outcome via a persuader's description. For example, she may be encouraged by the persuader, who may praise the student for performing well or comment on the difficulty of a problem. Her interpretation will be affected by the credibility she ascribes to the persuader. Thus, it is pedagogically constructive to suggest a student has the capabilities to succeed at a given task verbally, likely raising the student's self-efficacy. Verbal persuasion is particularly useful in enabling students to overcome self-doubt. Although verbal persuasion does not have a large impact on lasting student persistence it can encourage immediate action and effort. Fourth, with *physiological and emotional effects*, the student responds affectively to situations and their anticipation. These experiences, which often induce stress and anxiety, are manifested in physiological responses, such as increased heart rate and sweaty palms, call for emotional support and motivational feedback since they can be detrimental to success.

Student self-efficacy beliefs regulate human behavior through four major processes central to human performance (Bandura, 1997):

- **Cognitive Processes**. Self-efficacy affects student reasoning and problem-solving (Bandura, 1995; Schunk and Pajares, 2002; Zimmerman, 2000) to the point that performance can be elevated or impaired. High self-efficacy affords students the abilities to set ambitious future goals and a rigid commitment to achieve them. Furthermore, self-efficacious students are better able to select favorable problem-solving strategies and more quickly disregard inadequate approaches. On the other hand, low self-efficacy reduces the payoff of achieving weakly structured goals and elicits an inability to select optimal problem-solving strategies.
- **Motivational Processes**. Students with high self-efficacy are more likely to visualize successful outcomes. Setting challenging goals in turn yields elevated levels of motivation (Lepper et al., 1993), another construct affected by self-efficacy. Low self-efficacy interferes with visualizing, thereby reducing resilience and persistence abilities.

- **Selective Processes**. The activities that students choose to engage in significantly affects their potential to achieve. Students with high self-efficacy select challenging activities and environments that regularly present opportunities to exhibit persistence. Students with low self-efficacy tend to select activities and environments that present little or no challenge and can often be detrimental to the development of cognitive and social skills.
- **Affective Processes**. Self-efficacy influences students' abilities to regulate their own affective states. There are three fundamental ways in which self-efficacy influences affective state: self-control over thought, action, and affect (Bandura, 1997). First, *thought-oriented mode* refers to cognitive processes that are emotionally arousing and the ability to self-regulate such thoughts. Self-efficacy beliefs about one's ability to overcome risks and to persist through or avoid emotionally disturbing thoughts have great influence on behavior. Second, *action-oriented mode* refers to taking courses of action that effect change in the environment so that there is an increased potential for desirable affective outcomes. Third, *affect-oriented mode* refers to one's abilities to conceive adverse affective states when faced with adverse-emotion-invoking situations. Self-relaxation, calming inner monologue and controlled breathing are techniques often used to reduce undesirable emotional arousal.

Predicting self-efficacy holds great promise for intelligent tutoring systems and educational software in general. Self-efficacy beliefs have a stronger correlation with desired behavioral outcomes than many other motivational constructs (Graham and Weiner, 1996), and it has been recognized in multiple educational settings that self-efficacy can predict both motivation and learning effectiveness (Zimmerman, 2000). Thus, if it were possible to enable ITSs to accurately model self-efficacy, they might be able to leverage it to increase students' academic performance. Two recent efforts have explored the role of self-efficacy in ITSs. One introduced techniques for incorporating knowledge of self-efficacy in pedagogical decision making (Beal and Lee, 2005). Using a pre-test instrument and knowledge of problem-solving success and failure, instruction is adapted based on changes in motivational and cognitive factors. The second explored the effects of pedagogical agent design on students' traits, which included self-efficacy (Baylor and Kim, 2004; Kim, 2005). The focus of the experiments reported in this article is on the automated induction of self-efficacy models for runtime use by ITSs.

One can distinguish two fundamental approaches to modeling self-efficacy: analytical and empirical. In the *analytical* approach, models of self-efficacy can be constructed by analyzing the findings of the educational psychology literature. However, self-efficacy is not well understood computationally, i.e., the literature has not produced a set of rules defining precise characteristics of particular levels of self-efficacy. While we do have expressive computational models of affect, e.g., the OCC model (Ortony et al., 1988) and EMA (Gratch and Marsella, 2004) based on the Smith and Lazarus' appraisal theory of emotion (Lazarus, 1991), we do not have similarly rich, comprehensive models of self-efficacy. Moreover, because self-efficacy reasoning requires drawing inferences about a student's past experiences, her beliefs, her intentions, her affective state, her current situational context, and her capabilities, devising a complete and universal model of self-efficacy seems to be well beyond our grasp at the current juncture.

An alternative to analytically devising models of self-efficacy for intelligent tutoring systems is the *empirical* approach. If somehow we could create models of self-efficacy that were derived directly from observations of "efficacy in action," we could create empirically grounded models based on student behaviors exhibited during the performance of a specific task within a given

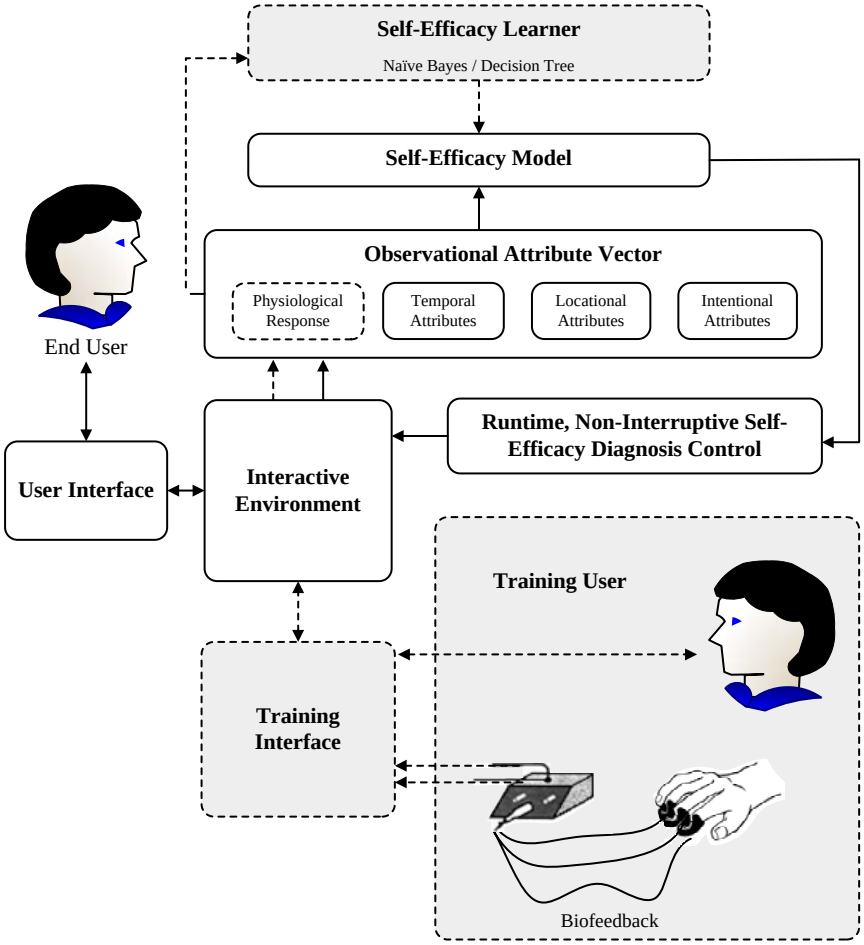


Figure 1. The SELF architecture. Dashed arcs represent self-efficacy model induction mode and solid arcs represent the runtime operation mode.

domain. While it is not apparent that this approach could produce a universal model of self-efficacy—a universal model may not even be achievable, at least in the near term—the empirical approach could nonetheless generate models of self-efficacy that significantly extend the pedagogical capabilities present in current educational software and intelligent tutoring systems.

3. Data-driven Self-efficacy Modeling

The prospect of creating a "self-efficacy learner" that can induce empirically grounded models of self-efficacy from observations of student interactions holds much appeal. To this end, we propose SELF, an affective data-driven paradigm that learns models of self-efficacy. SELF consists of a trainable architecture and a two-phase methodology of training and learning.

3.1. The SELF Architecture

The SELF framework operates in two modes: *self-efficacy model induction* in which the architecture interacts with a student trainer to gather data and *runtime operation*, in which it monitors student levels of self-efficacy based on observations of student interaction.

- **Self-efficacy Model Induction**. During model induction (depicted in Figure 1 with dashed arcs), SELF acquires training data and learns models of self-efficacy from training users interacting with the learning environment. The training user is outfitted with biofeedback equipment which monitors her heart rate and galvanic skin response. Biofeedback signals are recorded in training logs via the interactive environment, which also records an event stream produced by the training users' behaviors in the environment. In the online tutorial system this event stream included responses to the genetics questions, self-reports of self-efficacy, and temporal features, such as how long the student spent on the question. The interactive learning environment event stream also includes information regarding location and intention of the student in the 3D interactive environment. Together, the biofeedback signals and the corresponding elements in the event stream are assembled in temporal order into the observational attribute vector. After training sessions (typically involving multiple training users) are complete, the self-efficacy learner induces models from the observed situational data and physiological data. The students' self-reported self-efficacy levels serve as class labels for the training instances. The students are presented a "self-efficacy slider" with a scale ranging from 0 (low) to 100 (high). Students report their perceived levels of efficacy using this scale.
- **Runtime Operation**. During runtime operation (represented in Figure 1 with solid arcs), which is the mode employed when students interact with fielded learning environments, the induced models inform the pedagogical decision making of SELF-enhanced runtime components by predicting end-users' levels of self-efficacy. The learning environment again tracks all activities in the world and monitors the same observable attributes reported to the self-efficacy learner during self-efficacy model induction. The induced model is used by the self-efficacy diagnosis controller to (1) assess the situation to determine what level of self-efficacy the student is experiencing, and (2) determine which learning environment modules need to be informed of the changes (if any changes exist) in the students' level of self-efficacy. In runtime operation mode students may don biofeedback equipment if the model being used is a dynamic model, in which case the observational attribute vector expects to have a continuous feed of heart rate and skin conductance data.

3.2. Training Data Acquisition

To accurately model self-efficacy, an instrument needs to be devised that can provide a metric for the construct and that can be used by the induced models for prediction. Recall from Section 2 that a growing body of work reports on efforts to detect and recognize user affect from a variety of information sources including self-reports, peer reports, judges' reports, physiological response, body posture, eye tracking, and linguistic features of interactions. While sophisticated techniques have been developed for third-party detection of affect, e.g., analyzing recordings of facial expression (Ekman and Friesen, 1978), and a multitude of validated instruments have been devised for a broad range of affective phenomena, analogous techniques and instruments have not yet emerged for self-efficacy detection and measure. To date, self-reports have been the most widely used method for obtaining quantitative self-efficacy measurements (Baylor and Kim, 2004; Beal and Lee, 2005; Kim, 2005)[3]. Self-efficacy was therefore modeled with self-

[3] One approach to validating self-reports of efficacy is the test-retest method and the subsequent analysis to determine the reliability between self-reports for like questions. While this method is common in survey instruments for obtaining self-efficacy measurements, similar methodologies have yet to be devised for validating self-reports of efficacy gathered in real-time environments.

reports, which were represented with a 100 point scale. In the learning phase, self-report data were translated into multiple levels of granularity (2, 3, 4, and 5-level efficacy scales).

In addition, accurately modeling self-efficacy requires a representation of the situational context that satisfies two requirements: it must be sufficiently rich to support assessment of changing levels of self-efficacy, and it must be encoded with features that are readily observable at runtime. Because affect is fundamentally a cognitive process in which the user appraises the relationship between herself and her environment (Gratch and Marsella, 2004; Smith and Lazarus, 1990) and similarly, self-efficacy beliefs draw heavily on a student's appraisal of the situation at hand, affect recognition models (and models of self-efficacy) should take into account both physiological and environmental information. For task-oriented learning environments, self-efficacy models can leverage knowledge of task structure and the state of the student in the learning environment to effectively reason about students' efficacy levels. In particular, for such learning environments, self-efficacy models can employ concepts from appraisal theory (Lazarus, 1991) to recognize student efficacy levels generated from their assessment of how their abilities relate to the current learning objective and task. Thus, self-efficacy models can leverage representations of the information observable in the learning environment – note that this refers to the same information that students may use in their own appraisals – to predict student efficacy levels. The SELF framework therefore employs an expressive representation of all activities in the learning environment, including those controlled by users and the interactive system, by encoding them in an *observational attribute vector*, which is used in both the model induction mode and model usage mode of operation. During model induction, the observational attribute vector is passed to the self-efficacy learner for model generation; during runtime operation, the attribute vector is monitored by a SELF-enhanced runtime component that utilizes knowledge of user self-efficacy levels to inform effective pedagogical decisions. The observable attribute vector (Table 1) represents four interrelated categories of features for making decisions:

- **Temporal Features**: In the online tutorial system, SELF monitors the amount of time students spend on each question and how long the cursor resides in particular locations of the interface, since users tend to move their mouse according to the focus of their attention (Chen et al., 2001). In the interactive learning environment, SELF continuously tracks the amount of time that has elapsed since the student arrived at the current location, since the student achieved a goal, and since the student was last presented with an opportunity to achieve a goal. Temporal features are useful for measuring the persistence of the student on the current and past tasks.
- **Locational Features**: SELF tracks the location of the student's cursor in the online tutorial system. In the interactive learning environment, SELF continuously monitors the location of the student's character. It monitors locations visited in the past, locations recently visited, locations not visited, and locations being approached. There are 45 designated locations in the interactive learning environment (e.g., the laboratory, the living room of the men's quarters, and the area surrounding the waterfall). Locational features are useful for tracking whether students are in locations where learning tasks and current goals are achievable. When a student arrives in a location where a learning objective can be completed combined temporal attributes and locational features can aid in the prediction of the student exhibiting command of the learning task and associated levels of efficacy.

Table 1. Representative observational attributes monitored in the online tutorial system (OTS) and interactive learning environment (ILE), including temporal, locational, intentional and physiological features.

	Observational Attribute	Attribute Description	Possible Values	Applicable Environments OTS	ILE
Temporal Features	Question Time	The amount of time that has elapsed since the question was first displayed to the student	Positive real values	✓	
	Difference from Average Question Time	How does the amount of time the student has spent on the current question compare to the average time spent on previous questions (less or more)	Positive and negative real values	✓	
	Time in Current Location	The amount of time that the student has spent in a defined location of the interface	Positive real values	✓	✓
	Time on Current Learning Goal	The amount of time that the student has spent on current learning goal being attempted	Positive real values		✓
	Comprehensive Learning Time	The amount of time that has passed since the student began interacting	Positive real values	✓	✓
Locational Features	Current Location	The defined area in which the student's cursor is located (OTS) or the area in which the student's embodied character is located (ILE)	*OTS areas*: Question, Answer, Self-efficacy Slider, Submit *ILE areas*: Dining Hall, Waterfall, Lab Testing Area, Lab Reading Room, etc.	✓	✓
	Previous Location	The defined area in which the student's cursor was located (OTS) or the area in which the student's embodied character was located (ILE) immediately before the *Current Location*.	*Same as "Current Location" Observational Attribute above*	✓	✓
	Goal Achievable in Current Location	Whether or not the learning goal is achievable in the student's current location	True or False		✓
	Visited Location L	Whether or not the student has visited the particular location, L, for all locations, as designated by cursor location (OTS) and embodied character location (ILE)	True or False	✓	✓
	Number of visits to Location L	The number of times the student has visited the particular location, L, for all locations, as designated by cursor location (OTS) and embodied character location (ILE)	Positive integer values (values reset to 0 after each problem/goal)	✓	✓
Intentional Features	Problem/Goal	Identifier corresponding to individual problems (OTS) and learning goals (ILE)	*OTS*: Problem number (1-20) *ILE*: Goal name (test-milk, talk-to-Jin, locate-ill-characters, etc.)	✓	✓
	Progression	Number of problems/ goals solved	Positive integer values	✓	✓
	Progression Rate	Average amount of time required to solve problems and achieve goals	Positive real values	✓	✓
	Number of Locations Visited in Goal Pursuit	Average amount of time required to solve problems and achieve goals	Positive integer values	✓	✓
Physiological Features	Heart Rate	The student's beats per minute as measured by the interval between the last two heart beats	Positive real values	✓	✓
	Galvanic Skin Response	The electrical resistance of the student's skin as measured by the biofeedback apparatus	Positive real values	✓	✓
	Average HR and GSR	The student's average heart rate and galvanic skin response measured from the start of interaction	Positive real values	✓	✓
	Problem/Goal HR and GSR	The student's average heart rate and galvanic skin response measured from the start of the current problem/goal	Positive real values	✓	✓
	Sliding Window HR and GSR Averages	The student's average heart rate and galvanic skin response measured across multiple intervals of 5, 10, 15, 20, 30, 45 and 60 seconds	Positive real values	✓	✓
	Sliding Window HR and GSR Differences	The change in student's average heart rate and galvanic skin response measured across multiple intervals of 5, 10, 15, 20, 30, 45 and 60 seconds from the previous interval's window	Increasing, Decreasing, Same	✓	✓

- **Intentional Features**: In the interactive learning environment, SELF continuously tracks goals being attempted (as inferred from locational and temporal features, e.g., approaching a location where a goal can be achieved), goals achieved, the rate of goal achievement, and the effort expended to achieve a goal (as inferred from recent exploratory activities and locational features). These features enable models to incorporate knowledge of potential and student-perceived valence (positive and negative perceptions) of a given situation. Intentional features, such as goal progression, are useful for measuring how a student's abilities match the demands of the learning tasks. For example, a student that is rapidly achieving goals is more likely to be confident in their abilities to drive themselves towards success.
- **Physiological Response**: SELF continuously tracks readings from a biofeedback apparatus attached to the student's hand. Blood volume pulse and galvanic skin response readings are monitored at a rate of approximately 30 readings/second to accurately track changes in the student's physiological response. Blood volume pulse readings are used to compute student's heart rate and changes in their heart rate. SELF monitors trends in both student heart rate and galvanic skin response over a variety of fixed and sliding windows in addition to moment-to-moment readings. For instance, several fixed width averages of HR and GSR are monitored over the entire learning episode, for individual questions in the online tutoring system, fixed by the time the student takes to complete the question), and across individual learning objectives in the interactive learning environment, fixed by the time the student takes to complete the learning objective. SELF monitors HR and GSR trends in several sliding window frames of 5, 10, 15, 20, 30, 45, and 60 seconds. These sliding windows allow self-efficacy models to isolate changes in physiological response in the smaller windows that have little or no impact to the trends tracked in the longer windows. Other physiological response features include comparison attributes that monitor the change between current and past windows; summarizing the transition between the windows, i.e., whether HR and GSR are going up or down, and determining the rate of change between the windows.

In the SELF implementation for the online tutorial system, the observational attribute vector encodes nearly 150 features while in the interactive learning environment, CRYSTAL ISLAND, the observational attribute vector encodes 283 features. During model induction, a continuous stream of physiological data is collected and logged approximately 30 times per second. In addition, an instance of the observational attribute vector is logged every time a significant event occurs, yielding, on average, hundreds of vector instances each minute. We define a significant event to be a manipulation of the environment that causes one or more features of the observational attribute vector to take on new values. At runtime, the same features are continuously monitored by the respective environment. This may or may not include physiological response data depending on the incorporated model type, static or dynamic.

3.3. Learning SELF Models

During SELF model induction, the framework learns models of self-efficacy from the observational attribute vectors. Many types of models can be learned. Work to date has investigated two families: rule-based models (decision trees) and probabilistic models (naïve Bayes). Naïve Bayes and decision tree classifiers are effective machine learning techniques for generating preliminary predictive models. Naïve Bayes classification approaches produce probability tables that can be incorporated into runtime systems and used to continually update

probabilities for predicting self-efficacy. Naïve Bayes classifiers make an unsupported assumption (referred to as the "naïve assumption") that the attributes of the observational attribute vector are conditionally independent. Thus, the probability of two conditionally independent events, A and B both occurring is P(A and B | C) = P(A | C)P(B | C), where C is an observed event. Under the naïve assumption, gaining knowledge of event A occurring, given that we already know C, has no effect on the probability of event B occurring, and vice versa (Russell and Norvig, 2003). This assumption does not hold in the environments described in this article. For example, in the interactive learning environment there are many actions that are dependent on the location of the student's character (i.e., experiments can only be run in the laboratory). Despite the inaccurate assumption that all observable attributes are conditionally independent, it has been found that naïve Bayes classifiers can nevertheless perform well and often with performance comparable to other classification methods (Han and Kamber, 2005).

Decision trees provide interpretable rules that support runtime pedagogical decision making. The decision trees induced in this work make use of the well known C4.5 software extension of the ID3 decision tree induction algorithm (Quinlan, 1986), which has been incorporated in the WEKA machine learning toolkit as the J48 algorithm (Witten and Frank, 2005). The decision tree induction algorithm makes use of a top-down, divide-and-conquer approach. At each node, an information gain analysis is used to select the attribute with the highest information gain, thus reducing the amount of information needed, to a minimum, to make classifications in the node's sub-tree (Han and Kamber, 2005).

With both naïve Bayes and decision tree classifiers, SELF-enhanced runtime tutorial control components can monitor the state of the attributes in the probability tables (for naïve Bayes) or rules (for decision trees) to determine when conditions are met for predicting particular varying levels of self-efficacy. Both naïve Bayes and decision tree classification techniques are useful for preliminary predictive model induction for large multidimensional data, such as the 278-attributes taken from the 283-observed attribute vector used for learning in the interactive learning environment. Two approaches can be distinguished in learning techniques: those that are completely automated, and those that require the knowledge provided by a domain expert. SELF experiments reported below focus on fully automated learning approaches. SELF model induction proceeds in four phases:

- **Data Construction**: Each training log is first translated into a full observational attribute vector. For example, blood volume pulse (BVP) and galvanic skin response (GSR) readings were taken nearly 30 times every second reflecting changes in both heart rate and skin conductivity. The 278 attributes observed directly in the environment were combined with the selected self-reported levels of self-efficacy class labels, since only one class label can be used. Thus, 4 datasets are constructed; one for each level of granularity. Consider observable attributes a_1, a_2, …, a_{278}, and class labels c_{279}, c_{280}, c_{281}, c_{282}, c_{283} (c_{279} corresponds to the raw self-efficacy reports, c_{280} corresponds to two-level self-efficacy self-reports, c_{281} corresponds to three-level, c_{282} corresponds to four-level, and c_{283} corresponds to five-level self-efficacy self-reports). Each constructed dataset consists of all observable attributes, a_1, …, a_{278}, and one non-raw self-efficacy self-report class label.
- **Data Cleansing**: After data are converted into an attribute vector format a dataset is generated that contains only instances in which the biofeedback equipment was able to successfully monitor BVP and GSR throughout the entire learning session. For example, in the foundational evaluation described below, data from two sessions had to be discarded for this reason: BVP (used for monitoring heart rate) readings were difficult to obtain from this

participant. Two sessions did not satisfy these requirements and were subsequently removed from the interactive learning environment evaluation.

- **Naïve Bayes Classifier and Decision Tree Learning**: Once the dataset is prepared, it is passed to the learning systems. Each dataset was loaded into the WEKA machine learning toolkit (Witten and Frank, 2005), a naïve Bayes classifier and decision tree were learned, and tenfold cross-validation analyses were run on the resulting models (See Section 4.3.1 for details). The entire dataset was used to generate several types of self-efficacy models. These included models with different granularities of self-efficacy level representations.

The following section will present a foundational evaluation of SELF in an online tutorial system. Then after an introduction to CRYSTAL ISLAND, a second empirical study is presented in which SELF was again evaluated in a rich, narrative-centered, interactive learning environment.

4. Online Tutorial System Evaluation

In this experiment, two families of self-efficacy models were induced: the model learner constructed (1) *static* models, which are based on demographic data and a validated problem-solving self-efficacy instrument (Bandura, 2006), and (2) *dynamic* models, which extend static models by also incorporating real-time physiological data. Both families of resulting models operate at runtime, are efficient, and do not interrupt the learning process.

4.1. Method

4.1.1. Participants and Design

In a formal evaluation, data was gathered from thirty-three subjects in an Institutional Review Board (IRB) of North Carolina State University approved user study. There were 6 female and 27 male participants varying in age, race, and marital status. Approximately 12 (36%) of the participants were Asian, 20 (60%) were Caucasian, and 1 (3%) was Black or African-American. 27% of the participants were married. Participants average age was 26.15 (SD=5.32).

4.1.2. Materials and Apparatus

The pre-experiment paper-and-pencil materials for each participant consisted of a demographic survey, tutorial instructions, Bandura's Problem-solving Self-Efficacy Scale (Bandura, 2006), and the problem-solving system directions. Post-experiment paper-and-pencil materials consisted of a general survey. The demographic survey collected basic information such as gender, age, ethnicity, and marital status. The tutorial instructions explained to participants the details of the task, such as how to navigate through the tutorial and an explanation of the target domain. Bandura's validated Problem-solving Self-Efficacy Scale (Bandura, 2006), which was administered after participants completed a tutorial in the domain of genetics, asked them to rate how certain they were in their ability to successfully complete the upcoming problems (which they had not yet seen). The problem-solving system directions supplied detailed task direction to participants, as well as screenshots highlighting important features of the system display, such as the "self-efficacy slider."

The computer-based materials consisted of an online genetics tutorial and an online genetics problem-solving system. The online genetics tutorial consisted of an illustrated 15-page web document which included some animation and whose content was drawn primarily from a

middle school biology textbook (Padilla et al., 2000). The online genetics problem-solving system consisted of 20 questions, which covered material in the online genetics tutorial. The problem-solving system presented each multiple-choice question individually and required participants to rate their confidence, using the "self-efficacy slider," in their answer before proceeding to the next question.

Apparati consisted of a Gateway 7510GX laptop with a 2.4 GHz processor, 1.0 GB of RAM, 15-in. monitor and biofeedback equipment for monitoring blood volume pulse (one sensor on the left middle finger) and galvanic skin response (two sensors on the left first and third fingers). Participants' right hands were free from equipment so they could make effective use of the mouse in problem-solving activities.

4.2. Procedure

Each participant entered the experimental environment (a conference room) and was seated in front of the laptop computer. First, participants completed the demographic survey at their own rate. Next, participants read over the online genetics tutorial directions before proceeding to the online tutorial. On average, participants took 17.67 (SD = 2.91) minutes to read through the genetics online tutorial. Following the tutorial, participants were asked to complete the Problem-Solving Self-Efficacy Scale considering their experience with the material encountered in the genetics tutorial. The instrument asked participants to rate their level of confidence in their ability to successfully complete certain percentages of the upcoming problems in the problem-solving system. Participants did not have any additional information about the type of questions or the domain of the questions contained in forthcoming problems. Participants were then outfitted with biofeedback equipment on their left hand while the problem-solving system was loaded. Once the system was loaded, participants entered the calibration period in which they read through the problem-solving system directions. This allowed the system to obtain initial readings on the temporal attributes being monitored, in effect establishing a baseline for HR and GSR.

The problem-solving system presented randomly selected, multiple-choice questions to each participant. The participants selected an answer and then manipulated the self-efficacy slider representing the strength of their belief in their answer being correct. Participants completed 20 questions. They averaged 8.15 minutes (SD = 2.37) to complete the problem-solving system. Finally, they were asked to complete the post-experiment survey at their own rate before concluding the session.

After all participants' sessions were completed, the procedure (described in Section 3.3 was used to induce models of self-efficacy ratings from the training sessions (Figure 2). Each session log, containing on average 14,645.42 (SD = 4,010.57) observation changes (e.g., a change in location, student heart beat detected, or changes in selected answer), was first translated into a full observational attribute vector. For example, BVP and GSR readings were taken nearly 30 times every second reflecting changes in both heart rate and skin conductivity. Blood volume pulse (used for monitoring HR) readings were difficult to obtain from two participants resulting in the elimination of that data. The entire dataset was used to generate several types of self-efficacy models, each predicting self-efficacy with varying degrees of granularity. These included two-level models (Low, High), three-level models (Low, Medium, High), four-level models (Very Low, Low, High, Very High), and five-level models (Very Low, Low, Medium, High, Very High).

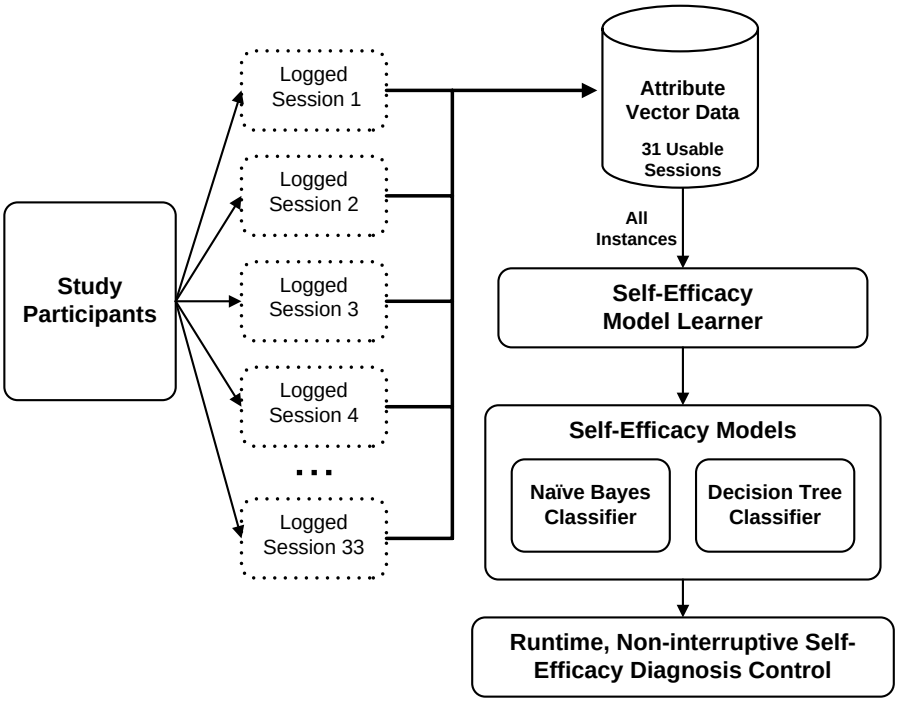


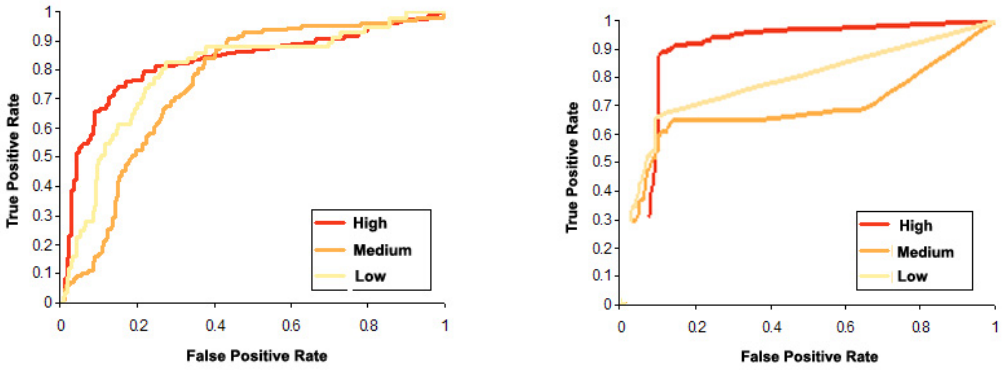
Figure 2. Online tutorial system foundational evaluation data flow.

4.3. Results

Below we present the results of the naïve Bayes and decision tree classification models and provide analyses of the collected data. Various ANOVA statistics are presented for results that are statistically significant. Because the tests reported here were performed on discrete data, we report Chi-square test statistics (χ^2), including both likelihood ratio Chi-square and the Pearson Chi-square values. Fisher's Exact Test is used to find significant p-values at the 95% confidence level ($p < .05$).

4.3.1. Model Results

Naïve Bayes and decision tree classifiers are effective machine learning techniques for generating preliminary predictive models. Naïve Bayes classification approaches produce probability tables that can be implemented into runtime systems and used to continually update probabilities for assessing student self-efficacy levels. Decision trees provide interpretable rules that support runtime decision making. The runtime system monitors the condition of the attributes in the rules to determine when conditions are met for assigning particular values of student self-efficacy. Both the naïve Bayes and decision tree machine learning classification techniques are useful for preliminary predictive model induction for large multidimensional data, such as the 144-attribute vector used in this experiment. Because it is unclear precisely which runtime variables are likely to be the most predictive, naïve Bayes and decision tree modeling provide useful analyses that can inform more expressive machine learning techniques (e.g., Bayesian networks) that also leverage domain experts' knowledge. Both static and dynamic models of self-efficacy were induced using naïve Bayes and decision tree classification



(a) (b)

Figure 3. ROC curves for naïve Bayes (a) and decision tree (b) three-level models of self-efficacy. Overall the naïve Bayes model correctly classified 72% of the instances while the decision tree was able to correctly classify 83%.

techniques. Dynamic models were induced from all observable attributes, while static models excluded physiological response data.

All models were constructed using a tenfold cross-validation scheme. In this scheme, data is decomposed into ten equal partitions, nine of which are used for training and one used for testing. The equal parts are swapped between training and testing sets until each partition has been used for both training and testing. Tenfold cross-validation is widely used for obtaining the best estimate of error (Witten and Frank, 2005).

Cross-validated ROC curves are useful for presenting the performance of classification algorithms for two reasons. First, they represent positive classifications, included in a sample, as a percentage of the total number of positives, against negative classifications as a percentage of the total number of negatives (Witten and Frank, 2005). Second, the area under ROC curves is widely accepted as a generalization of the measure of the probability of correctly classifying an instance (Hanley and McNeil, 1982).

The ROC curves depicted in Figure 3 show the results of both a naïve Bayes and decision tree three-level model. Low-confidence was noted by a student self-efficacy rating lower than 33 (on a 0 to 100 scale). Medium-confidence was determined by rating between 33 and 67, while High-confidence was represented all ratings greater than 67. The smoothness of the curve in Figure 3(a) indicates that the data collected seems to have sufficiently covered the multidimensional space for inducing naïve Bayes models. The jaggedness of the curves in Figure 3(b) indicates that training data did not cover the entirety of the instance space. While sufficient data was collected for the induction process and modeling the phenomena of self-efficacy, further training may be useful to obtain complete coverage of the multidimensional space. In particular, further investigation will be required to gather data from situations in which there are more opportunities for students to experience low self-efficacy. Although training data did not cover all possible instances in the multidimensional space (notice how the ROC curves for induced decision tree models do not extend to the axis in Figure 3b), the decision tree model performed significantly better than the naïve Bayes model (likelihood ratio, $\chi^2 = 21.64$, Pearson, $\chi^2 = 21.47$, $p < .05$). The highest performing model induced from all data was the two-level decision-tree based dynamic model, which performed significantly better than the highest performing static model, which was a two-level decision tree model (likelihood ratio, $\chi^2 = 3.99$,

Pearson, $\chi^2 = 3.97$, $p < .05$). The three-level dynamic decision tree model was also significantly better than the static three-level decision tree (likelihood ratio, $\chi^2 = 18.26$, Pearson, $\chi^2 = 18.13$, $p < .05$). All model results are presented in Table 2.

The performance of two dynamic naïve Bayes models proved to be significantly better than baseline models. Both of the dynamic two-level model (likelihood ratio, $\chi^2 = 4.272$, $p = 3.87 \times 10^{-2}$, and Pearson, $\chi^2 = 4.26$, $p = 3.9 \times 10^{-2}$, $df = 1$) and the dynamic four-level model (likelihood ratio, $\chi^2 = 10.647$, $p = 1.1 \times 10^{-3}$, and Pearson, $\chi^2 = 10.615$, $p = 1.1 \times 10^{-3}$, $df = 1$) yielded significant improvements over the baseline models. No static naïve Bayes models' performance was significantly better than baseline models. The performance of static decision tree models also did not produce significant results over baseline performance. However, all dynamic decision tree models did perform significantly better than baseline models (Table 3).

Table 2. Model accuracy results (area under ROC curves). Static models were induced from non-intrusive demographic and Problem-Solving Self-Efficacy data. Dynamic models were also based on physiological data. Baseline models report the portion of the distribution pertaining to the most reported efficacy level (i.e., 80.6% of self-efficacy reports for the two-level models were High). * Value represents model performance statistically significant from baseline performance.

		Static Model Accuracy	Dynamic Model Accuracy
Two-level Models	Baseline (High)	80.6%	80.6%
	Naïve Bayes	82.2%	85.2%*
	Decision Tree	82.9%	86.9%*
Three-level Models	Baseline (High)	69.8%	69.8%
	Naïve Bayes	70.1%	71.8%
	Decision Tree	73.4%	83.4%*
Four-level Models	Baseline (Very High)	65.4%	65.4%
	Naïve Bayes	68.8%	74.7%*
	Decision Tree	69.0%	78.9%*
Five-level Models	Baseline (Very High)	60.9%	60.9%
	Naïve Bayes	63.4%	64.2%
	Decision Tree	63.9%	75.3%*

Table 3. Dynamic decision tree model improvements were statistically significant over baseline model accuracies.

Granularity	Baseline Accuracy	Decision Tree Accuracy	Likelihood Ratio χ^2	Likelihood Ratio p	Pearson χ^2	Pearson p	df
Two-level models	80.6%	86.9%*	8.264	4.0×10^{-3}	8.216	4.2×10^{-3}	1
Three-level models	69.8%	83.4%*	28.304	1.037×10^{-7}	27.967	1.234×10^{-7}	1
Four-level models	65.4%	78.9%*	23.764	1.089×10^{-6}	23.582	1.197×10^{-6}	1
Five-level models	60.9%	75.3%*	25.159	5.279×10^{-7}	24.995	5.747×10^{-7}	1

4.3.2. Model Attribute Effects on Self-efficacy

Heart rate and galvanic skin response had significant effects on self-efficacy predictions (Table 4). Participants' age was the only demographic attribute to have a significant effect on all levels of self-efficacy models. Table 4 presents several effects of physiological response and pre-experiment survey data, including demographic information and Bandura's problem-solving self-efficacy scale, on self-efficacy predictions. These results suggest that when modeling self-efficacy at higher-granularity it becomes more important to account for student demographics. Two-level self-efficacy models have the least significant effectors. This is likely due to the results of the two-level baseline model, in which 80.6% of the efficacy self-reports are classified with the label, "High".

Table 4. Chi-squared values representing the significance effects of physiological signals, demographics, and Bandura's problem-solving self-efficacy scale instrument on dynamic self-efficacy models ($p < 0.5$).

		Self-efficacy			
	Effect	Two-level	Three-level	Four-level	Five-level
Physiological Signals	Heart rate	-	9.58	15.35	12.78
	Galvanic Skin Response	-	9.24	17.96	14.82
Demographics	Gender	-	-	18.10	11.14
	Age	16.25	50.00	94.64	87.64
	Race & Ethnicity	-	-	-	-
Bandura's Problem-solving Self-efficacy Scale	Collective Responses	36.29	86.82	182.67	159.87
	10% Response	6.72	-	15.51	14.04
	20% Response	6.40	15.99	33.19	22.01
	30% Response	-	14.26	17.30	11.98
	40% Response	-	6.63	-	-
	50% Response	-	-	-	-
	60% Response	-	-	-	-
	70% Response	-	-	-	-
	80% Response	-	11.43	22.43	20.76
	90% Response	-	17.97	34.87	28.38
	100% Response	-	-	13.18	11.69

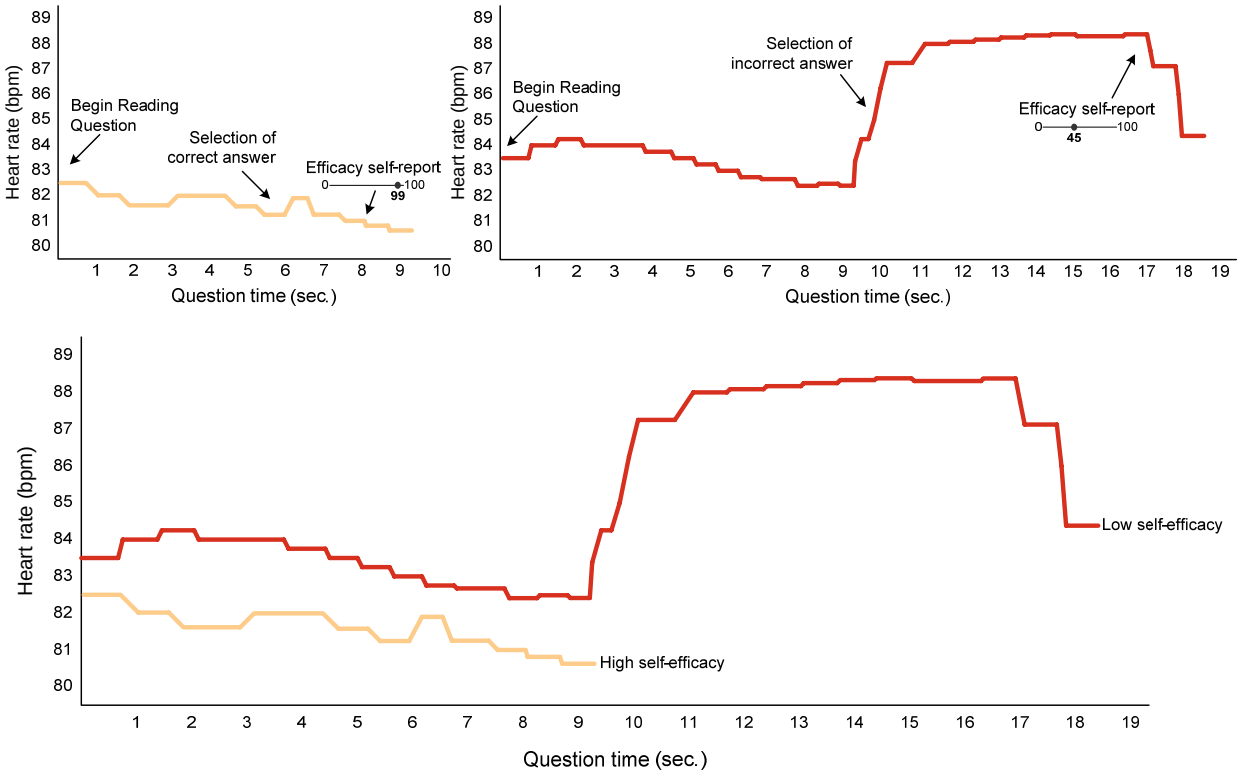


Figure 4. Heart rate for student reporting high self-efficacy (upper left image), heart rate for same student reporting low self-efficacy on a different problem (upper right image), and the student's heart rates combined (lower image).

4.4. Discussion

Self-efficacy is closely associated with motivational and affective constructs that both influence (and are influenced by) a student's physiological state. It is therefore not unexpected that a student's physiological state can be used to more accurately predict her self-efficacy. For example, Figure 4 shows the heart rates for one participant in the study over the course of solving two problems. In Figure 4, in the upper left image, the participant reported high levels of self-efficacy for a particular question, while the same participant whose heart rate progression is also shown in the upper right image of Figure 4 reported a low level of self-efficacy for another question. The heart rate for the student reporting high self-efficacy gradually drops as they encounter a new question, presumably because of their confidence in their ability to successfully solve the problem. In contrast, the heart rate for the same student reporting low self-efficacy spikes dramatically, an increase of 5 beats per minute in less than 2 seconds, when the student selects an incorrect answer. This phenomenon is noteworthy since the students were in fact not given feedback about whether or not their responses were correct. Instead the student's self-appraisal seems to lead to the determination of low efficacy, an inability to successfully achieve at the current task, without a requirement of confirmation of their assessment. It appears that through some combination of cognitive and affective processes the student's uneasiness with her response, even in the absence of direct feedback, was enough to bring about a significant physiologically manifested reaction. Curiously, there is a subsequent drop in heart rate after the student reports her low level of self-efficacy. In this instance, it seems

that providing an opportunity to acknowledge a lack of ability and knowledge to perform may itself reduce anxiety.

The experiment has two important implications for the design of runtime self-efficacy modeling. First, even without access to physiological data, induced decision-tree models can make predictions about students' self-efficacy that are more accurate than baseline models. Sometimes physiological data is unavailable or it would be too intrusive to obtain the data. In these situations, decision-tree models that learn from demographic data and data gathered with a validated self-efficacy instrument administered prior to problem solving and learning episodes, can model self-efficacy. Second, if runtime physiological data are available, they can significantly enhance self-efficacy modeling. Given access to HR and GSR, self-efficacy can be predicted more accurately.

In summary, the static models are able to predict students' real-time levels of self-efficacy with 73% accuracy, and the resulting dynamic models are able to achieve 83% predictive accuracy. Thus, non-intrusive static models achieve a statistically significant improvement over baseline performance, and their predictive power can be increased by further enriching them with physiological data at varying levels of granularity.

5. Interactive Learning Environment Evaluation

The results of the foundational evaluation reported in Section 4 indicated that an inductive approach offered potential as a method for modeling self-efficacy and called for further investigation of the techniques. The design of the second experiment was motivated by three factors: explicitly controlling the challenge levels of the learning environment; exploiting task structure to study self-efficacy with an appraisal-theoretic (Lazarus, 1991) framework; and increasing the complexity of the learning environment and, therefore the dimensionality of the induction problem.

1. *Explicitly controlling the level of challenge of learning tasks in an effort to increase the frequencies of reported low self-efficacy.* In the first evaluation the majority of reported levels of self-efficacy were classified as "high" (see baseline model results in Section 4, Table 2). The dynamic nature of an interactive learning environment would allow for the design of tasks of varying degrees of difficulty, presenting a variety of challenge levels to study participants. Individual tasks could be designed to be more complicated, require more actions to complete, and elicit student persistence to reach achievement.
2. *Exploiting task structure and notions from appraisal theory (Lazarus, 1991) to model self-efficacy.* An immersive, visually-rich interactive learning environment would offer an ideal testbed in which to study the interaction between student self-appraisals and self-efficacy. Recall that self-efficacy beliefs arise from one's appraisal of the environment and the current situation in conjunction with appraisals of one's abilities to achieve goals given the current and possible future states of the surrounding environment. Thus, it is likely that a rich learning environment would elicit patterns of self-efficacy in response to student appraisals of unfolding events in learning episodes. In turn, the representation of the environment should then enable induced models to accurately predict student self-efficacy.
3. *Automatically inducing models of self-efficacy from observations in an increasingly complex interactive narrative-centered learning environment.* The induction task becomes increasingly difficult as more dimensions are added to represent more complex

learning environments. The second empirical study was designed to investigate the potential and the value of creating models of self-efficacy in more complex interactive learning environments, and to "stress-test" the induction approach in higher dimensions.

Together, these factors motivated the second experiment investigating SELF model induction in a rich interactive learning environment.

5.1. Interactive Narrative-Centered Learning Environments

Narrative is central to human cognition. Because of the motivational force of narrative, it has long been believed that story-based education can be both engaging and effective. Much educational software has been devised for story-based learning. These systems include both research prototypes and a long line of commercially available software. However, this software relied on scripted forms of narrative: they employed either predefined linear plot structures or simple branching storylines. In contrast, one can imagine a much richer form of narrative learning environment that dynamically crafts customized stories for individual students at runtime. Recent years have seen the emergence of a growing body of work on dynamic narrative generation (Cavazza et al., 2002; Riedl and Young, 2004; Si et al., 2005), and narrative has begun to play an increasingly important role in intelligent tutoring systems (Machado et al., 2001; Mott and Lester, 2006b; Riedl et al., 2005).

Narrative experiences are powerful. In his work on cognitive processes in narrative comprehension, Gerrig identifies two properties of reader's narrative experiences (1993). First, readers are transported, i.e., they are somehow taken to another place and time in a manner that is so compelling it seems real. Second, they perform the narrative. Like actors in a play, they actively draw inferences and experience emotions as if their experiences were somehow real. It is becoming apparent that narrative can be used as an effective tool for exploring the structure and process of "meaning making." For example, narrative analysis is being adopted by those seeking to extend the foundations of psychology (Bruner, 1990) and film theory (Branigan, 1992).

Learning environments may utilize narrative to their advantage. One can imagine narrative-centered curricula that leverage a student's innate metacognitive apparatus for understanding and crafting stories. This insight has led educators to recognize the potential of contextualizing all learning within narrative (Wells, 1986). Because of the active nature of narrative, by immersing learners in a captivating world populated by intriguing characters, narrative-centered learning environments can enable learners to participate in the construction of narratives, to engage in active problem solving, and to reflect on narrative experiences (Mott et al., 1999). These activities are particularly relevant to inquiry-based learning. Inquiry-based learning emphasizes the student's role in the learning process via concept building (Zachos et al., 2000) and hypothesis formation, data collection, and testing (Glaser et al., 1992). For example, a narrative-centered inquiry-based learning environment for science education could foster an in-depth understanding of how real-world science plays out by featuring science mysteries whose plots are dynamically created for individual students.

5.1.1. Affect and Motivation in Narrative-centered Inquiry-based Learning

Narrative-centered inquiry-based learning environments may also offer motivational benefits. Motivation is critical in learning environments, for it is clear that from a practical perspective,

educational software that fails to engage students will go unused. Game playing experiences and educational experiences that are extrinsically motivating can be distinguished from those that are intrinsically motivating (Malone, 1981). In contrast to extrinsic motivation, intrinsic motivation stems from the desire to undertake activities sheerly for the prospective reward. Narrative-centered discovery learning could provide the four key intrinsic motivators identified in the classic work on motivation in computer games and educational software (Malone and Lepper, 1987): challenge, curiosity, control, and fantasy.

Narrative-centered inquiry-based learning should feature challenging tasks of intermediate levels of difficulty, i.e., tasks that are not too easy and not too difficult, targeting desirable levels of student intrinsic motivation. Dynamically created narratives can feature problem-solving episodes whose level of difficulty is customized for individual students. In inquiry-based approaches, learning is inherently presented as a challenge, a series of problem-solving goals, that once achieved provide a deeper understanding of the domain. Devising narratives and providing tutorial feedback that both maintain a delicate level of uncertainty about the possibility of attaining each goal and sufficient reporting of student performance and progress is critical to sustaining effective levels of challenge.

Curiosity plays a central role in successful learning in narrative-centered inquiry-based learning environments. Since inquiry-based learning compels students to obtain knowledge throughout learning episodes on their own (materials are not provided explicitly prior to interaction) students are likely to question the completeness of their acquired knowledge as they progress, searching for new answers, stimulating their curiosity.

Narrative-centered inquiry-based learning environments can empower students to take control of their learning experiences; students can choose their own paths, both figuratively (through the solution space) and literally (through the storyworld), while being afforded significant guidance crafted specifically for them. The narrative structure of inquiry-based learning can provide unobtrusive direction by indirectly highlighting a subset of possible goals (i.e., blinking lights in a particular room in the environment, or a character audibly coughing in the student's right audio channel) for the student's next action consideration, maintaining the student's perception of control.

Narrative-centered inquiry-based learning is innately fantasy-based. Fantasy refers to a student's identification with characters in the interactive narrative and the imaginative situations created internally and off-screen by the student. All narrative elements ranging from plot and characters to suspense and pacing can contribute to vivid imaginative experiences. The openness of discovery learning provides scaffolding to support all levels of student imagination, increasing motivation and engagement. Effective narrative tutorials will engage characters in the storyworld that either the individual students perceive as possessing some cognitive, emotional, or physical similarities with themselves, or that the individual student admires, expresses feelings of compassion towards, or for which the student conveys empathetic feelings. In short, narrative can provide the guidance essential for effective inquiry-based learning and the "affective scaffolding" for achieving high levels of motivation and engagement.

5.1.2. The CRYSTAL ISLAND Learning Environment

In our laboratory we are developing a narrative-centered inquiry-based learning environment. Some components are fully designed and implemented while others are in the early stages of

Figure 5: CRYSTAL ISLAND.

design. The prototype learning environment, CRYSTAL ISLAND (Mott et al., 2006), is being created in the domains of microbiology and genetics for middle school students (Figure 5).

CRYSTAL ISLAND features a science mystery set on a recently discovered volcanic island where a research station has been established to study the unique flora and fauna. The user plays the protagonist attempting to discover the genetic makeup of the chickens whose eggs are carrying an unidentified infectious disease at the research station. The story opens by introducing her to the island and the members of the research team for which her father serves as the lead scientist. As members of the research team fall ill, it is her task to discover the cause of the specific source of the outbreak. She is free to explore the world and interact with other characters while forming questions, generating hypotheses, collecting data, and testing her hypotheses. Throughout the mystery, she can walk around the island and visit the infirmary, the lab, the dining hall, and the living quarters of each member of the team. She can pick up and manipulate objects, and she can talk with characters to gather clues about the source of the disease. In the course of her adventure she must gather enough evidence to correctly choose which breeds of chickens need to be banned from the island.

The virtual world of CRYSTAL ISLAND, the semi-autonomous characters that inhabit it, and the user interface were implemented with Valve Software's Source™ engine, the 3D game platform for Half-Life 2. The Source engine also provides much of the low-level (reactive) character behavior control. The character behaviors and artifacts in the storyworld are the subject of continued work. The narrative planner of CRYSTAL ISLAND has been implemented with an HTN planner that is based on the SHOP2 planner (Nau et al., 2001). For efficiency, the planner was designed as an embeddable C++ library to facilitate its integration into high-performance 3D gaming engines. A decision-theoretic "director" agent based on dynamic decision networks has been implemented to guide the narrative in the face of uncertain user actions (Mott and Lester, 2006a), and the method and operator libraries for the genetics and microbiology domains are currently being constructed.

Figure 6: CRYSTAL ISLAND character located in the laboratory.

To illustrate the behavior of the CRYSTAL ISLAND learning environment, consider the following situation. Suppose a student has been interacting within the storyworld and learning about infectious diseases, genetic crosses and related topics. In the course of having members of her research team become ill, she has learned that an infectious disease is an illness that can be transmitted from one organism to another. As she concludes her introduction to infectious diseases, she learns from the camp nurse that the mystery illness seems to be coming from eggs laid by certain chickens and that the source or sources of the disease must be identified. The student is introduced to several characters. Some characters are able to help identify which eggs come from which chickens while other characters, with a scientific background, are able to provide helpful genetics information (Figure 6). The student discovers through a series of tests that the bad eggs seem to be coming from chickens with white-feathers. The student then learns that this is a codominant trait and determines that any chicken containing the allele for white-feathers must be banned from the island immediately to stop the spread of the disease. The student reports her findings back to the camp nurse.

5.2. Method

5.2.1. Participants and Design

In a formal evaluation, data was gathered from 42 subjects in an Institutional Review Board (IRB) of North Carolina State University approved user study. There were 5 female and 37 male participants. Participants average age was 21.2 (SD = 1.96).

5.2.2. Materials and Apparatus

The pre-experiment materials for each participant consisted of an online demographic survey and Bandura's Self-Efficacy Scale (Bandura, 2006). The experiment materials consisted of the following: tutorial directions, the online genetics tutorial, the CRYSTAL ISLAND backstory and directions, the CRYSTAL ISLAND interactive environment control sheet, the CRYSTAL ISLAND

Figure 7. Interactive learning environment user outfitted with biofeedback apparatus.

character profiles and world map, the genetics problem-solving self-efficacy questionnaire (Bandura, 2006), the genetics problem-solving system directions, the online problem-solving system, and a post-experiment survey. The demographic survey collected participant information such as age, gender, race and ethnicity. Bandura's Self-Efficacy Scale rates the participants' self-efficacy in a variety of more general domains. The tutorial directions described the simple navigation controls and lack of time constraints for reading through the genetics tutorial. The CRYSTAL ISLAND backstory and directions presented the participant's task and some background information about their character. The controls reference sheet described which keys and mouse movements would be needed to manipulate their agent in the training task. The character profiles provided pictures with associated names and job descriptions of characters the participant might meet on the island. The CRYSTAL ISLAND map was a tool to help the participants maintain orientation within the environment and provide navigational assistance. The genetics problem-solving self-efficacy questionnaire was administered to gauge the participants' self-efficacy with respect to solving genetics problems after completing both the tutorial and CRYSTAL ISLAND interaction. The post survey was used to determine how participants would feel about using similar software in educational settings and their thoughts on affect and self-efficacy uses in videogames and educational software.

Apparati consisted of a Gateway 7510GX laptop with a 2.4 GHz processor, 1.0 GB of RAM, 15-in. monitor and biofeedback equipment for monitoring blood volume pulse (one sensor on the right ring finger) and galvanic skin response (two sensors on the right middle and little fingers).

5.3. Procedure

First participants completed the online demographic survey and the online general self-efficacy questionnaire (Bandura, 2006). Participants then completed the genetics tutorial which took anywhere from 5 minutes to 25 minutes. Next, participants were wired with biofeedback sensors similar to those worn by the user in Figure 7. The practice task was then completed allowing

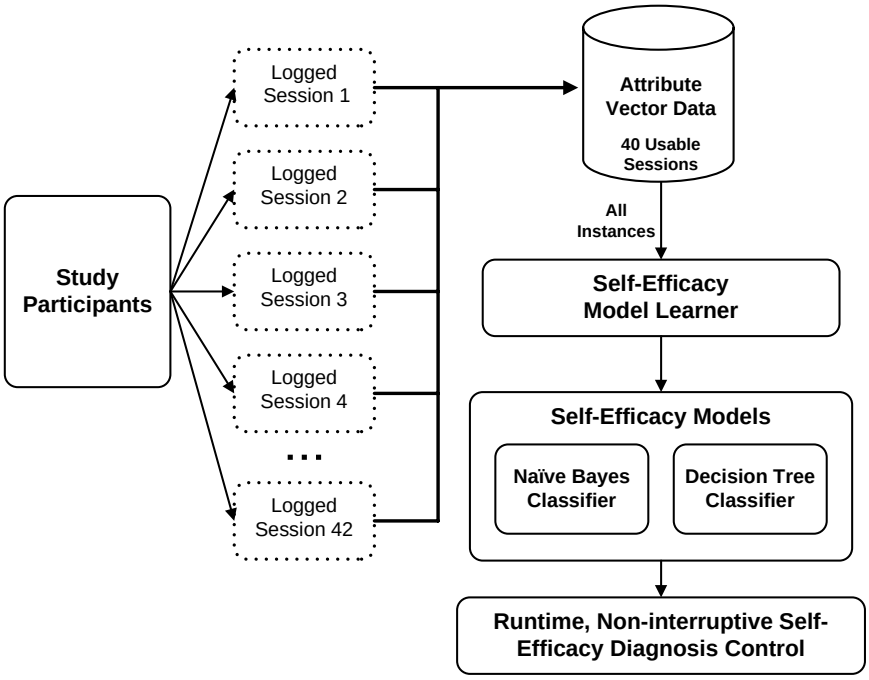


Figure 8. Interactive learning environment evaluation data flow.

participants to become familiar with the controls for CRYSTAL ISLAND. Participants were then presented the CRYSTAL ISLAND materials (backstory, controls, map, and character profiles) while the virtual environment was loaded. Once participants indicated they were prepared and had any questions answered by the research assistant, they began their interaction in CRYSTAL ISLAND. As participants solved the genetics mystery on CRYSTAL ISLAND, they were periodically asked to rate their current level of self-efficacy, i.e., their current belief in their abilities to solve the science mystery. Upon completion of interacting with CRYSTAL ISLAND, participants completed the genetics self-efficacy questionnaire (Bandura 2006) prior to receiving the problem-solving system directions. Once participants indicated they were prepared and physiological response measurements had been calibrated, they began solving 20 randomly displayed genetics problems. Each question was presented with 4 multiple-choice answers and a "self-efficacy slider" which participants adjusted indicating their belief in their ability to correctly solve the given problem. Finally, participants completed the post-experiment questionnaire before the experiment session concluded.

After all participants' sessions were completed, the same procedure as the one described in Section 3.3 was used to induce models of self-efficacy ratings from the training sessions (Figure 8). Training sessions lasted at least eight minutes, and each session log contained at least 15,000 (32,487 at most) observation changes (e.g., a change in location, completing a goal, manipulating an object, or detected heart beat). These changes were first translated into a full observational attribute vector. For example, BVP and GSR readings were taken approximately 30 times every second reflecting changes in both heart rate and skin conductivity. After data were converted into an attribute vector format a dataset was generated that contained only records in which the biofeedback equipment was able to successfully monitor BVP and GSR throughout the entire training session and in which participants actively participated in the experiment by providing self-reports. Two training sessions from male participants did not satisfy these requirements.

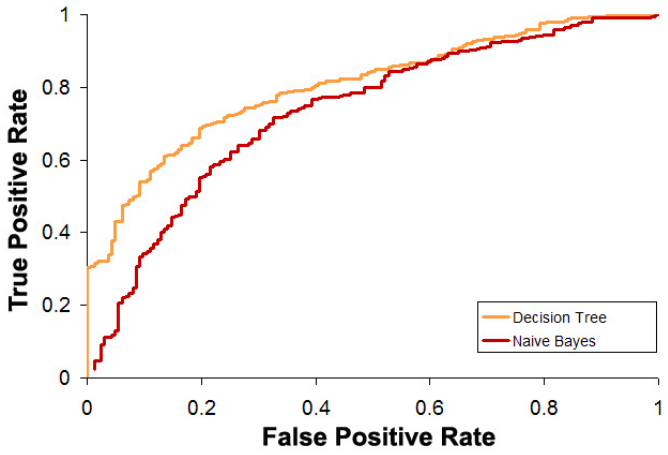


Figure 9: ROC curves for SELF three-level models induced from CRYSTAL ISLAND interactions.

Self-efficacy models were again produced at varying levels of granularity. These included two-level models (Low, High), three-level models (Low, Medium, High), four-level models (Very Low, Low, High, Very High), and five-level models (Very Low, Low, Medium, High, Very High).

5.4. Results

All models were evaluated using a tenfold cross-validation scheme for producing training and testing datasets. The ROC curves (Figure 9) show the results of decision tree and naïve Bayes modeling for predicting student levels of self-efficacy. The lack of smoothness of the curves indicates that training data did cover the entirety of the multidimensional space. However, collected training data was sufficient for inducing SELF models of self-efficacy. The highest performing model induced from interactive learning environment training data was the two-level decision tree model, correctly predicting more than 87% of reported levels of self-efficacy. Table 5 reports the results of the self-efficacy model induction mode of SELF. Decision tree models' prediction improvements over naïve Bayes models were significant at the two-level models (likelihood ratio, $\chi^2 = 7.321$, $p = 6.8 \times 10^{-3}$, and Pearson, $\chi^2 = 7.291$, $p = 6.9 \times 10^{-3}$, df = 1) and four-level models (likelihood ratio, $\chi^2 = 24.085$, $p = 9.218 \times 10^{-7}$, and Pearson, $\chi^2 = 23.96$, $p = 9.835 \times 10^{-7}$, df = 1). Furthermore, decision tree models performed significantly better than baseline models: two-level models (likelihood ratio, $\chi^2 = 29.319$, $p = 6.139 \times 10^{-8}$, and Pearson, $\chi^2 = 28.929$, $p = 7.506 \times 10^{-8}$, df = 1), three-level models (likelihood ratio, $\chi^2 = 62.443$, $p = 2.74 \times 10^{-15}$, and Pearson, $\chi^2 = 61.56$, $p = 4.29 \times 10^{-15}$, df = 1), and four-level models (likelihood ratio, $\chi^2 = 25.759$, $p = 3.869 \times 10^{-7}$, and Pearson, $\chi^2 = 25.617$, $p = 4.163 \times 10^{-7}$, df = 1). Naïve Bayes models performance was significantly better than baseline models for two-level models (likelihood ratio, $\chi^2 = 7.433$, $p = 6.4 \times 10^{-3}$, and Pearson, $\chi^2 = 7.412$, $p = 6.5 \times 10^{-3}$, df = 1) and three-level models (likelihood ratio, $\chi^2 = 43.494$, $p = 4.25 \times 10^{-11}$, and Pearson, $\chi^2 = 43.099$, $p = 5.2 \times 10^{-11}$, df = 1). Table 6 reports the results of self-efficacy models induced in both the online tutorial system and the interactive learning environment.

Table 5. Model results – area under ROC curves for dynamic self-efficacy models. * Value represents model performance statistically significant from baseline performance.

		Dynamic Model Accuracy
Two-level Models	Baseline (High)	76.1%
	Naïve Bayes	82.1%*
	Decision Tree	87.3%*
Three-level Models	Baseline (High)	61.2%
	Naïve Bayes	77.6%*
	Decision Tree	80.4%*
Four-level Models	Baseline (Very High)	63.5%
	Naïve Bayes	64.0%
	Decision Tree	72.9%*
Five-level Models	Baseline (Very High)	62.4%
	Naïve Bayes	59.2%
	Decision Tree	63.2%

Table 6. Model results – area under ROC curves for online tutorial system static and dynamic self-efficacy models, and interactive learning environment dynamic models. * Value represents model performance statistically significant from baseline performance.

		Online Tutorial System		Interactive Learning Environment
		Static Model Accuracy	Dynamic Model Accuracy	Dynamic Model Accuracy
Two-level Models	Baseline (High)	80.6%	80.6%	76.1%
	Naïve Bayes	82.2%	85.2%*	82.1%*
	Decision Tree	82.9%	86.9%*	87.3%*
Three-level Models	Baseline (High)	69.8%	69.8%	61.2%
	Naïve Bayes	70.1%	71.8%	77.6%*
	Decision Tree	73.4%	83.4%*	80.4%*
Four-level Models	Baseline (Very High)	65.4%	65.4%	63.5%
	Naïve Bayes	68.8%	74.7%*	64.0%
	Decision Tree	69.0%	78.9%*	72.9%*
Five-level Models	Baseline (Very High)	60.9%	60.9%	62.4%
	Naïve Bayes	63.4%	64.2%	59.2%
	Decision Tree	63.9%	75.3%*	63.2%

In the online tutorial system evaluation, the majority of self-efficacy self-reports were classified as being high efficacy, as indicated by the baseline models (the portion of the distribution belonging to the majority class). Thus, in the interactive learning environment development, some tasks were designed to present more challenging scenarios to students than were presented in the online tutorial system in an effort to elicit a higher percentage of low efficacy self-reports. While the baseline results indicate that the majority of self-efficacy self-reports in the interactive learning environment evaluation were also classified as high and very high efficacy, we obtained significantly more instances of students reporting low efficacy. Table 7 reports the baseline dynamic models from both evaluations and likelihood ratio and Pearson's statistics indicating the reduced accuracy in the interactive learning environment dynamic baseline models to be statistically significant. Since baseline models are composed of high self-efficacy report instances, a drop in baseline models (drop in the count of high self-efficacy reports) corresponds directly to an increase in counts of low self-efficacy report instances. This observation of a reduction in the quantity of high self-efficacy reports indicates a significant gain in the quantity of low self-efficacy reports. This fact is supported by the results presented in Table 7.

Table 7. Baseline comparisons between the online tutorial system and the interactive learning environment evaluations. The percentage increase in the number of instances in which students reported low levels of self-efficacy from the online tutorial system to the interactive learning environment evaluation was statistically significant. * For the five-level dynamic baseline model, comparison p-values are slightly above .05 indicating weak significance.

	Online Tutorial System	Interactive Learning Environment	Likelihood Ratio		Pearson		
			χ^2	p	χ^2	p	df
Two-level Dynamic $Baseline_{(High)}$ Model	80.6%	76.1%	17.264	3.253×10^{-5}	17.028	3.253×10^{-5}	1
Three-level Dynamic $Baseline_{(High)}$ Model	69.8%	61.2%	9.476	2.1×10^{-3}	9.434	2.1×10^{-3}	1
Four-level Dynamic $Baseline_{(Very\ High)}$ Model	65.4%	63.5%	5.004	2.53×10^{-2}	4.972	2.58×10^{-2}	1
Five-level Dynamic $Baseline_{(Very\ High)}$ Model	60.9%	62.4%	3.470	$*6.25 \times 10^{-2}$	3.458	$*6.29 \times 10^{-2}$	1

5.5. Discussion

A notable difference between the online tutorial system evaluation and the interactive learning environment evaluation is the dimensionality of the observational attribute vector. Recall that 150 features were observed in the online tutorial system, while in the interactive learning environment over 275 features were continuously monitored. This added dimensionality called for a larger dataset covering a larger space to improve the predictive accuracy of self-efficacy modeling. The training data obtained from the 40 usable sessions appears to have been sufficient for modeling self-efficacy in CRYSTAL ISLAND. The design of CRYSTAL ISLAND learning tasks, and particularly the varying challenge levels of the tasks, led to an increase in reports of low-efficacy in the interactive learning environment evaluation. This observation may explain why

SELF-induced models of self-efficacy obtained similar levels of accuracy among comparable models in the interactive learning environment as they did in the online tutoring system. It is noteworthy considering the increased dimensionality and complexity constraints placed on the induction process for learning self-efficacy models in the CRYSTAL ISLAND learning environment.

One of the challenging tasks in the design of SELF for the interactive learning environment evaluation was selecting observable attributes to monitor throughout student interactions that would also be used in student appraisal and self-efficacy determination. Because of the difficult nature of identifying attributes used by most students in appraisal, we elected to monitor the large 283-dimensional space designed for CRYSTAL ISLAND. The performance of induced models suggests that there is overlap between the features contained in the observable attribute vector and the attributes of the learning environment used by students in appraisal and realized in reports of self-efficacy.

We have considered a variety of models in the online tutorial system and the interactive learning environment along three dimensions: static vs. dynamic data, classification technique, and granularity. The online tutorial system evaluation found that dynamic models (inclusion of physiological data) performed significantly better, i.e., they correctly classified student self-efficacy more accurately, than static models (exclusion of physiological data) of self-efficacy. This result motivated the focus of investigating only dynamic models in the interactive learning environment evaluation. We hypothesize the performance improvements of dynamic models stems from the relationship between self-efficacy and physiological response. Because physiological responses follow from emotional reactions to situation appraisals (Frijda 1986; Picard, 1997) and self-efficacy beliefs arise from a similar cognitive appraisal process (Bandura, 1997), it seems appropriate to infer that changes in physiology are perhaps generated in response to a combination of interacting affective factors, such as emotional state, self-efficacy beliefs, and motivational states. Following research that has demonstrated the ability to recognize affective state from classification of physiological data (Burleson, 2006; Conati, 2002; Healey, 2000; Picard et al., 2001; Prendinger et al., 2005), it seems reasonable to infer that physiological response data may also be useful in predictions of self-efficacy.

Both evaluations investigated two families of classification techniques: rule-based models (decision trees) and probabilistic models (naïve Bayes). In the online tutorial system and the interactive learning environment, decision tree models outperformed naïve Bayes models. We hypothesize that this is likely due to the naïve Bayes assumption that all observable attributes are conditionally independent. As noted above, this is clearly not the case in CRYSTAL ISLAND where particular events can only occur in particular locations, such as running an experiment on an artifact, which requires the use of stationary machinery that can only be found in the laboratory on CRYSTAL ISLAND.

Induced models of self-efficacy also vary in the levels of granularity in which they predicted student efficacy. There is a noticeable decay in model performance as the granularity is increased in both evaluations. For instance, the performance of dynamic decision tree models from the interactive learning environment evaluation were 87.3%, 80.4%, 72.9%, and 63.2% for two, three, four, and five-level models respectively. Despite the trend of decreasing accuracy with increasing levels of granularity there are several instances worth noting, such as the performance of the two-level dynamic decision tree model from the interactive learning environment which accurately predicted 80.4% of instances, outperforming the associated baseline by 19.2% (the baseline model achieved 61.2% accuracy). However, it remains clear

that as granularity is increased the multidimensionality of the observation attribute vector hinders the ability to accurately predict student efficacy levels. For runtime environments, this decay effect raises the question of which level of granularity should implemented models use to predict self-efficacy. The answer to this question must consider the tradeoffs between models which calls for analyzing the increasing number of misclassifications associated with each additional level of granularity and how misclassifications affect system performance. For instance, consider the two-level dynamic decision tree model from the interactive learning environment which was able to predict 87.3% instances correctly. The 12.7% of instances that were incorrectly classified were predicted to be in the other class of the two-level model, i.e., instances of high self-efficacy were misclassified as low, and low self-efficacy instances were misclassified has high in 12.7% of all instances. After introducing another level of granularity, yielding a three-level model, we notice performance slips to 80.4% with an increase in misclassifications accounting for 19.6% of all instances. While higher granularity models do indeed provide more information than low levels of granularity, misclassifications can increase. This highlights the tradeoff question: when should models with higher levels of granularity (and therefore more precision) but with lower predictive accuracy be preferred to models with lower levels of granularity (and therefore less precision) but with higher predictive accuracy? In the future, it will be important to consider the tradeoff question in evaluations of runtime self-efficacy models.

6. Discussion and Design Implications

Both the foundational evaluation with an online tutorial system and the follow-up evaluation with an interactive learning environment suggest that it is possible to model self-efficacy from observable attributes with induced models achieving statistically significant improvements in performance over baseline models. The two experiments suggest that it may be possible to devise empirically based models that can then be used to support learning in interactive settings. Recall from Section 2 that Bandura distinguishes four types of self-efficacy effectors: enactive mastery experiences, vicarious experiences, verbal persuasion, and physiological and affective state (Bandura, 1997). Here, for each type of effector, we consider how ITSs may employ tutorial strategies to enhance and maintain ideal levels of student self-efficacy in conjunction with a SELF-like self-efficacy diagnostic framework.

ITSs can facilitate mastery learning (Bloom, 1984) by creating experiences in which the difficulty of the task or specific problems is adapted to the individual student. Diagnosing self-efficacy can better inform the pedagogical decisions bearing on the selection of problem difficulty by ensuring that the student has not only mastered the concept but believes in her abilities to use acquired knowledge in the domain. When self-efficacy models determine that a student has low efficacy beliefs during particular problem-solving tasks, an ITS can redirect the student's tasks to prior concepts or sub-problems that will help the student gain confidence in the skills required to solve the problem eliciting low self-efficacy. Self-efficacy models could contribute to improved pedagogical planning by informing the planner when replanning is necessary for individual students. Self-efficacy models could also contribute to error correction decision making, and they could play a role in determining when to intervene to provide tutorial guidance. Since efficacious students are likely to persist longer than students with low self-efficacy, pedagogical monitoring components might permit efficacious students to work through their own mistakes and consider intervening when mistakes are made by inefficacious students.

Challenge is an intrinsic motivator that is often employed by human tutors (Lepper et al., 1993). Self-efficacy models could inform decisions about the appropriate challenge level of tasks to create adaptive learning experiences that sustain ideal levels of self-efficacy and motivation, which in turn support effective learning. The amount of learning that takes place relates to the amount of mental effort students exhibit which has an "inverted U" relationship to self-efficacy (Clark, 1999). Thus, the difference between low self-efficacy and high self-efficacy needs to be handled delicately by ITSs. Just as too low self-efficacy can constrain learning, so too can too high self-efficacy.

The adaptability of ITSs may enable them to create vicarious experiences, which are sometimes difficult to elicit in a classroom setting. In particular, peer learning companions (Aimeur et al., 2000; Burleson and Picard, 2004; Chan and Baskin, 1990; Chou et al., 2003; Goodman et al., 1998; Kim, 2004) can create adaptable vicarious experiences for students. Student observation of similar peers succeeding may enhance the observing student's self-efficacy if she believes she can also succeed at the same or similar tasks (Schunk, 1987). Consider an ITS in which a peer companion agent fails or struggles at a task. Witnessing this event may enable less efficacious students to exert more effort if they believe their abilities to be greater than the companion agent's abilities. Likewise, highly efficacious students may persist as a companion agent begins to succeed at similar tasks and problems. This form of competition with a learning companion could contribute to increases in student efficacy. It has been determined that student perception of a companion agent's knowledge level can have a material effect on student self-efficacy (Baylor and Kim, 2004). Monitoring such perceptions could support the orchestration of agent and environment behaviors, and it could inform the adaptive selection of agent personae that most effectively support interactions with individual students. Enabling an ITS to adaptively control the perception of peers in the learning environment through personae selection, agent task completion, and interactive dialogue to demonstrate agent knowledge (or lack there of) are promising techniques for enhancing and maintaining student self-efficacy.

Verbal persuasion is a common motivational tool used by tutors (Lepper et al., 1993), both human and automated. Tutors who express confidence in a student's abilities can have a profound effect on the student's own self-efficacy beliefs. The impact is determined by the value the student places on the persuader, so an established relationship between a tutor and the student makes verbal persuasion all the more powerful. ITS research has considered several approaches to providing feedback (Aleven et al., 2004; Corbett and Anderson, 2001; Moreno, 2004), but feedback that improves self-efficacy can also be less performance-driven. In a study that targeted students with academic problems, direct feedback on success did not affect self-efficacy; rather, feedback on the selected cognitive strategies to develop a solution substantially enhanced student self-efficacy beliefs (Schunk and Rice, 1987). This is not to discount the potential effects of rewarding performance, especially verbally. Verbal performance feedback ensures that students are aware of goal progression, immersed in challenging tasks, and may contribute to student task persistence. Verbal persuasion is not as powerful as enactive mastery or vicarious experiences, particularly for inducing lasting effects on student efficacy beliefs (Bandura, 1997). Verbal persuasion is a technique that learning companions might employ if students are closing in on learning goals and self-efficacy models are beginning to detect declining student efficacy. In short, verbal persuasion can quickly elicit short bursts of efficacy to motivate students at critical junctures in learning episodes.

The final effector Bandura considers is physiological and affective state. This calls for self-efficacy modeling and affect recognition to operate in tandem. Changes in affective state and the subsequent changes in student physiology will impact self-appraisals of efficacy. Thus, devising strategies to guide students toward affective states with lower arousal levels will diminish the adverse effects of high-arousal physiological responses on student efficacy. For example, stress elicits aroused responses, such as increased heart rate and sweaty palms. Such responses may cause adverse self-appraisals of efficacy. Employing affect recognition combined with self-efficacy models can inform interactive pedagoical components to take action when situations of arousal and low self-efficacy co-occur. One approach to addressing student affect is to respond appropriately, given the social interactive context of an ongoing learning episode, through empathetic companion agents (Kim, 2005; McQuiggan and Lester 2006a, Paiva et al., 2005; Prendinger and Ishizuka, 2005). The empathetic nature of such agents may help students better self-regulate their own affective state leading to stronger senses of efficacy. Recognizing that physiological and affective states influence self-efficacy beliefs and in turn, that self-efficacy affects affective processes (Bandura, 1997), self-efficacy modeling can play an important role in the affective-loop of ITSs.

7. Conclusions and Future Work

Self-efficacy is an affective construct that may be useful for increasing the effectiveness of tutorial decision making by intelligent tutoring systems. It may contribute to increasing students' level of effort, the degree of persistence with which they approach problem solving, and, ultimately, the levels of success they achieve. However, to provide accurate and useful information, self-efficacy models must be able to operate at runtime, i.e., during problem-solving episodes, they must be efficient, and they must avoid interrupting learning. A promising approach to constructing models of self-efficacy is inducing them rather then manually constructing them. In controlled experiments, it has been demonstrated that *static* models induced from demographic data, a validated self-efficacy instrument, and information from the tutorial system can accurately predict student's self-efficacy during problem solving. It has also been empirically demonstrated that *dynamic* models enriched with physiological data can more accurately predict student's self-efficacy during problem solving.

The findings reported here contribute to the growing body of work on affective reasoning for learning environments. They represent a first step towards a computational theory of self-efficacy that can be leveraged to increase motivation and learning effectiveness. The foundational study evaluated SELF in an online tutorial system generating predictive models of self-efficacy. This study served as a proof-of-concept and guided the design of a second evaluation which investigated self-efficacy modeling in an interactive learning environment, CRYSTAL ISLAND. The interactive learning environment evaluation results extend the findings of the foundational study and suggest self-efficacy can be modeled in intelligent tutoring systems.

Several directions for future work are suggested by the results. First, the effect of specific pedagogical actions on students' self-efficacy should be investigated. It may be possible to quantitatively gauge the influence of competing tutorial strategies on students' self-efficacy, which might further increase learning effectiveness. Second, SELF generated models can be incorporated into a full scale intelligent tutoring system so that the impact of SELF-informed tutorial components on learning can be empirically investigated. It is important to gauge the manner and degree to which students benefit from tutorial strategy selection informed by self-

efficacy models. Finally, self-efficacy information might be used to enhance models of cognitive, motivational, selective, and affective processes. For example, prediction of self-efficacy combined with affect recognition models that can detect student frustration may more accurately predict how students will cope with negative affect, such as frustration, which could lead to predictions of how long the student will persist in frustrating situations. Such mechanisms could contribute to pedagogical strategies that enable students to learn more effectively and to increase their self-efficacy. Investigating new frameworks and methodologies for modeling processes that integrate efficacy information is an important next step in incorporating self-efficacy diagnosis into intelligent tutoring systems.

Acknowledgements

The authors wish to thank the members of the IntelliMedia Center for Intelligent Systems at North Carolina State University for their assistance in implementing SELF and CRYSTAL ISLAND, the Decision Systems Laboratory of the University of Pittsburgh for access to the SMILE Bayesian inference engine, Omer Sturlovich and Pavel Turzo for use of their 3D model libraries, and Valve Software for access to the Source™ engine and SDK. Thanks also to Michael Young and the members of the Liquid Narrative Group at NC State University for insightful discussions on narrative. This research was supported by the National Science Foundation under Grant REC-0632450. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

Aïmeur, E., C. Frasson, H. Dufort: 2000, Co-operative Learning Strategies for Intelligent Tutoring Systems. *Applied Artificial Intelligence*. **14**(5), 465-490.

Aleven, V., B. McLaren, I. Roll, and K. Koedinger: 2004, Toward Tutoring help seeking: Applying Cognitive modeling to meta-cognitive skills. *Seventh International Conference on Intelligent Tutoring Systems*, Maceió, Brazil, pp. 227-239.

Allanson, J., and S. Fairclough: 2004, A Research Agenda for Physiological Computing. *Interacting with Computers*, **16**(5), 857-878.

André, E., and M. Mueller: 2003, Learning Affective Behavior. *Tenth International Conference on Human-Computer Interaction*, Heraklion, Crete, Greece, pp. 512-516.

Bandura, A.: 1986, *Social Foundations of Thought and Action: A Social Cognitive Theory*. Englewood Cliffs, N.J.:Prentice-Hall.

Bandura, A.: 1995, Exercise of Personal and Collective Efficacy in Changing Societies. In: A. Bandura (ed.) *Self-efficacy in Changing Societies*. Cambridge, MA: Cambridge University Press, pp.1-45.

Bandura, A.: 1997, *Self-efficacy: The Exercise of Control*. New York, NY: Freeman.

Bandura, A.: 2006, Guide for constructing self-efficacy scales, in *Self-Efficacy Beliefs of Adolescents*, F. Pajares and T. Urdan, Eds. Greenwich, Connecticut: Information Age Publishing, pp. 307-337.

Baylor, A., and Y. Kim: 2004, Pedagogical Agent Design: The Impact of Agent Realism, Gender, Ethnicity, and Instructional Role. *Seventh International Conference on Intelligent Tutoring Systems*, Maceió, Brazil, pp. 592-603.

Beal, C., and H. Lee: 2005, Creating a Pedagogical Model that uses Student Self Reports of Motivation and Mood to Adapt ITS Instruction. *Workshop on Motivation and Affect in Educational Software, in conjunction with the Twelfth International Conference on Artificial Intelligence in Education*, Amsterdam, Netherlands, pp. 39-46.

Beer, J., E. Heerey, D. Keltner, R. Knight, and D. Scabini: 2003, The regulatory function of self-conscious emotion: Insights from patients with orbitofrontal damage. *Journal of Personality and Social Psychology*, **85**, 594-604.

Bloom, B.: 1984, The 2 Sigma Problem: The Search for Methods of Group Instruction as Effective as One-to-One Tutoring. *Educational Researcher*, **13**, 4-16.

Branigan, E.: 1992, *Narrative Comprehension and Film*. London, UK: Routledge.

Bruner, J.: 1990, *Acts of Meaning*. Cambridge, MA: Harvard University Press.

Burleson, W.: 2006, Affective Learning Companions: Strategies for Empathetic Agents with Real-Time Multimodal Affective Sensing to Foster Meta-Cognitive and Meta-Affective Approaches to Learning, Motivation, and Perseverance, PhD thesis, Massachusetts Institute of Technology.

Burleson, W., and R. Picard: 2004, Affective Agents: Sustaining Motivation to Learn Through Failure and a State of Stuck. *Workshop of Social and Emotional Intelligence in Learning Environments, in conjunction with the Seventh International Conference on Intelligent Tutoring Systems*, Maceió, Brazil.

Cavazza, M., F. Charles, and S. Mead: 2002, Interacting with Virtual Characters in Interactive Storytelling. *First International Joint Conference on Autonomous Agents and Multi-Agent Systems*, Bologna, Italy, pp. 318-325.

Chan, T.W., and A. Baskin: 1990, Learning companion systems. *Intelligent Tutoring Systems: at the Crossroads of Artificial Intelligence and Education*. C. Frasson and G Gauthier, Eds., Norwood, NJ: Ablex Publishing, pp. 6-33.

Chen, M., J. Anderson, and M. Sohn: 2001, What Can a Mouse Cursor Tell Us More? Correlation of Eye/mouse Movements on Web Browsing. *Extended Abstracts CHI 2001*, New York, NY:ACM Press, 281-282.

Chou, C.Y., T.W. Chan, C.J. Lin: 2003, Redefining the Learning Companion: the Past, Present, and Future of Educational Agents, *Computers and Education*, **40**, 255-269.

Clark, R.: 1999, Yin and Yang Cognitive Motivational Processes Operating in Multimedia Learning Environments. In van Merrienböer, J. (Ed.) *Cognition and Multimedia Design*. Herleen, Netherlands: Open University Press, pp.73-107.

Conati, C.: 2002, Probabilistic Assessment of User's Emotions in Educational Games. *Applied Artificial Intelligence*, **16**, 555-575.

Conati, C., and H. Maclaren: 2005, Data-driven Refinement of a Probabilistic Model of User Affect. *Tenth International Conference on User Modeling*. New York, NY, pp. 40-49.

Corbett, A., and J. Anderson: 2001, Locus of Feedback Control in Computer-based Tutoring: Impact on Learning Rate, Achievement and Attitudes. *Proceedings of CHI Letters*, **3**(1), 245-252.

Delcourt, M., and M. Kinzie: 1993, Computer Technologies in Teacher Education: the Measurement of Attitudes and Self-efficacy. *Journal of Research and Development in Education*. **27**(1), 35-41.

De Vicente, A., and H. Pain: 2002, Informing the Detection of the Students' Motivational State: an Empirical Study. *Sixth International Conference on Intelligent Tutoring Systems*. New York, NY:Springer-Verlag, 933-943.

Ekman, P, and W. Friesen: 1978, *The facial action coding system: A technique for the measurement of facial movement*. Palo Alto: Consulting Psychologists Press.

Frijda, N.: 1996, *The Emotions*. Cambridge, UK: Cambridge University Press.

Gerrig, R.: 1993, *Experiencing Narrative Worlds: On the Psychological Activities of Reading*. New Haven, CT: Yale University Press.

Gilleade, K., and J. Allanson: 2003, A Toolkit for Exploring Affective Interface Adaptation in Videogames, Proceedings of *Human–Computer Interaction International*, Crete, Greece, 370-374.

Glaser, R., L. Schauble, K. Raghavan, and C. Zeitz: 1992, Scientific Reasoning Across Different Domains. In E. De Corte, M. Linn, H. Mandle, and L. Verschaffel (eds.): *Computer-Based Learning Environments and Problem Solving*, Berlin, Germany: Springer-Verlag, pp. 345-373.

Goleman, D.: 1995, *Emotional Intelligence*. New York, NY: Bantam Books.

Goodman, B., A. Soller, F. Linton, and R. Gaimari: 1998, Encouraging Student Reflection and Articulation Using a Learning Companion. *International Journal of Artificial Intelligence in Education*, **9**, 237–255.

Graham, S., and B. Weiner: 1996, Principles and Theories of Motivation. In D. Berliner, and R. Calfee (eds.): *Handbook of Educational Psychology*. New York, NY: MacMillan Publishing, pp.63-84.

Graesser, A., B. McDaniel, P. Chipman, A. Witherspoon, S. D'Mello, and B. Gholson: 2006, Detection of Emotions During Learning with AutoTutor. *Twenty-eighth Annual Conference of the Cognitive Science Society*, 285-290.

Gratch, J., and S. Marsella: 2004, A Domain-independent Framework for Modeling Emotion. *Journal of Cognitive Systems Research*, **5**(4), 269-306.

Han, J., and M. Kamber: 2005, *Data Mining: Concepts and Techniques*, Second Edition. San Francisco, CA: Morgan Kaufmann Publishers.

Hanley, J., and B. McNeil: 1982, The Meaning and Use of the Area Under the Receiver Operating Characteristic (ROC) Curve. *Radiology*, **143**, 29-36.

Healey, J.: 2000, Wearable and Automotive Systems for Affect Recognition from Physiology. *PhD thesis*, Massachusetts Institute of Technology.

Johnson, L., and P. Rizzo: 2004, Politeness in Tutoring Dialogs: "Run the Factory, That's What I'd Do". *Seventh International Conference on Intelligent Tutoring Systems*, Maceió, Brazil, pp. 67-76.

Kapoor, A., and R. Picard: 2005, Multimodal Affect Recognition in Learning Environments, ACM Multimedia, Hilton, Singapore, 677-682.

Kim, Y.: 2005, Empathetic Virtual Peers Enhanced Learner Interest and Self-efficacy. *Workshop on Motivation and Affect in Educational Software, in conjunction with the Twelfth International Conference on Artificial Intelligence in Education*, Amsterdam, Netherlands, pp. 9-16.

Kim, Y.: 2004, Pedagogical Agents as Learning Companions: the Effects of Agent Affect and Gender on Learning, Interest, Self-efficacy, and Agent Persona. *PhD thesis*, The Florida State University.

Lang, P.: 1995, The emotion probe: Studies of motivation and attention. *American Psychologist*, **50**(5), 372-385.

Lazarus, R.: 1991, *Emotion and Adaptation*. Oxford, UK: Oxford University Press.

Lepper, M., M. Woolverton, D. Mumme, and J. Gurtner: 1993, Motivational Techniques of Expert Human Tutors: Lessons for the Design of Computer-based Tutors. In S. Lajoie and S. Derry (eds.): *Computers as Cognitive Tools*, Hillsdale, NJ: Erlbaum, pp. 75-105.

Lester, J., S. Towns, C. Callaway, J. Voerman, and P. FitzGerald: 2000, Deictic and Emotive Communication in Animated Pedagogical Agents. In J. Cassell, J. Sullivan, S. Prevost, and E. Churchill (eds.): *Embodied Conversational Agents*, Cambridge, MA: MIT Press, pp. 123-154.

Litman, D., and K. Forbes-Riley: 2006, Recognizing Student Emotions and Attitudes on the Basis of Utterances in Spoken Tutoring Dialogues with both Human and Computer Tutors. Speech Communication, **48**(5), 559-590.

Machado, I., P. Brna, and A. Paiva: 2001, Learning by Playing: Supporting and Guiding Story-Creation Activities. *Tenth International Conference on Artificial Intelligence in Education*, San Antonio, Texas, pp. 334-342.

Malone, T.: 1981, Toward a Theory of Intrinsically Motivating Instruction. *Cognitive Science*, **5**(4), 333-369.

Malone, T. and M. Lepper: 1987, Making Learning Fun: a Taxonomy of Intrinsic Motivations for Learning. In R. Snow and M. Farr (eds.), *Aptitude, Learning, and Instruction: Conative and Affective Process Analyses*, vol. 3, Hillsdale, NJ: Erlbaum, pp. 223-253.

McQuiggan, S., S. Lee, and J. Lester: 2006, Predicting User Physiological Response for Interactive Environments: An Inductive Approach, *Second Conference on Artificial Intelligence and Interactive Digital Entertainment*, Marina del Rey, CA, pp. 60-65.

McQuiggan, S., and J. Lester: 2006a, Learning Empathy: A Data-Driven Framework for Modeling Empathetic Companion Agents. *Fifth International Joint Conference on Autonomous Agents and Multi-Agent Systems*, Hakodate, Japan, 961-968.

McQuiggan, S., and J. Lester: 2006b, Diagnosing Self-efficacy in Intelligent Tutoring Systems: An Empirical Study. *Eighth International Conference on Intelligent Tutoring Systems*, Jhongli, Taiwan, 565-574.

Mekeig, S., and M. Inlow: 1993, Lapses in Alertness: Coherence of Fluctuations in Performance and EEG Spectrum. *Electroencephalography and Clinical Neurophysiology*, **86**, 23–25.

Moreno, R.: 2004, Decreasing Cognitive Load for Novice Students: Effects of Explanatory versus Corrective Feedback in Discovery-based Multimedia. *Instructional Science*, **32**, 99-113.

Mota, S., and R. Picard: 2003, Automated Posture Analysis for Detecting Learner's Interest Level. First IEEE Workshop on Computer Vision and Pattern Recognition for Human-Computer Interaction, Madison, WI.

Mott, B., C. Callaway, L. Zettlemoyer, S. Lee, and J. Lester: 1999, Towards Narrative-Centered Learning Environments. *Proceedings of the 1999 Fall Symposium on Narrative Intelligence*, Cape Cod, MA, pp. 78-82.

Mott, B., and J. Lester: 2006a, U-DIRECTOR: A Decision-Theoretic Narrative Planning Architecture for Storytelling Environments. *Fifth International Conference on Autonomous Agents and Multi-Agent Systems*, Hakodate, Japan, 977-984.

Mott, B., and J. Lester: 2006b, Narrative-centered Tutorial Planning for Inquiry-based Learning Environments, *Eighth International Conference on Intelligent Tutoring Systems*, Jhongli, Taiwan, pp. 675-684.

Mott, B., S. McQuiggan, S. Lee, S.Y. Lee, and J. Lester: 2006, Narrative-centered Environments for Guided Discovery Learning, *Workshop on Agent-Based Systems for Human Learning in conjunction with fifth International Conference on Autonomous Agents and Multi-Agent Systems*, Hakodate, Japan, pp.22-28.

Nau, D., H. Muñoz-Avila, Y. Cao, A. Lotem, and S. Mitchell: 2001, Total-Order Planning with Partially Ordered Subtasks. *Seventeenth International Joint Conference on Artificial Intelligence*. Seattle, WA, pp.999-1004.

Ortony, A., G. Clore, and A. Collins: 1988, *The Cognitive Structure of Emotions*. Cambridge, UK: Cambridge University Press.

Padilla, M., I. Miaoulis, and M. Cyr: 2000, *Science Explorer: Cells and Heredity*. Teacher's Edition, Upper Saddle River, NJ: Prentice Hall.

Pajares, F., and J. Kranzler: 1995, Self-Efficacy Beliefs and General Mental Ability in Mathematical Problem Solving. *Contemporary Educational Psychology*, **20**, 426-443.

Paiva, A., J. Dias, D. Sobral, R. Aylett, S. Woods, L. Hall, and C. Zoll: 2005, Learning by Feeling: Evoking Empathy with Synthetic Characters. *Applied Artificial Intelligence*, **19**, 235-266.

Partala, T., and V. Surakka: 2003, Pupil Size Variation as an Indication of Affective Processing. *International Journal of Human–Computer Studies*, **59**, 185–198.

Picard, R.: 1997, *Affective Computing*. Cambridge, MA: MIT Press.

Picard, R., E. Vyzas, and J. Healey: 2001, Toward Machine Emotional Intelligence: Analysis of Affective Physiological State. *IEEE Transactions Pattern Analysis and Machine Intelligence* **23**(10), 1185-1191.

Pope, A., E. Bogart, and D. Bartolome: 1995, Biocybernetic System Evaluates Indices of Operator Engagement in Automated Task. *Biological Psychology*, **40**, 187–195.

Porayska-Pomsta, K., and H. Pain: 2004, Providing Cognitive and Affective Scaffolding Through Teaching Strategies: Applying Linguistic Politeness to the Educational Context. *Seventh International Conference on Intelligent Tutoring Systems*, Maceió, Brazil, pp. 77-86.

Prendinger, H., and M. Ishizuka: 2005, The Empathic Companion: A Character-based Interface that Addresses Users' Affective States. *Applied Artificial Intelligence*, **19**, 267-285.

Prendinger, H., J. Mori, and M. Ishizuka: 2005, Using Human Physiology to Evaluate Subtle expressivity of a virtual Quizmaster in a Mathematical Game. *International Journal of Human-Computer Studies*, **62**, 231-245.

Quinlan, J.: 1986, Induction of decision trees. *Machine Learning*, **1**(1), 81-106.

Riedl, M., H. Lane, R. Hill, and W. Swartout: 2005, Automated Story Direction and Intelligent Tutoring: Towards a Unifying Architecture. *Workshop on Narrative Learning Environments at the Twelfth International Conference on Artificial Intelligence in Education*, Amsterdam, Netherlands, pp. 23-30.

Riedl, M. and M. Young: 2004, An Intent-Driven Planner for Multi-Agent Story Generation. *Third International Joint Conference on Autonomous Agents and Multi-Agent Systems*, New York, NY, pp. 186-193.

Russell, S., and P. Norvig: 2003, Artificial Intelligence: A Modern Approach, Second Edition. Upper Saddle River, NJ: Prentice Hall.

Schunk, D.: 1987, Peer Models and Children's Behavioral Change. *Review of Educational Research*, **57**, 149-174.

Schunk, D., and J. Rice: 1987, Enhancing Comprehension Skill and Self-efficacy with Strategy Value Information, *Journal of Reading Behavior*, **19**, 285-302.

Schunk, D. and F. Pajares: 2002, The Development of Academic Self-efficacy. In A. Wigfield and J. Eccles (eds.): *Development of Achievement Motivation* San Diego, CA: Academic Press, pp. 15-31.

Si, M., S. Marsella, and D. Pynadath: 2005, Thespian: Using Multi-Agent Fitting to Craft Interactive Drama. *Fourth International Conference on Autonomous Agents and Multi-Agent Systems*, Utrecht, Netherlands, pp. 21-28.

Smith, C., and R. Lazarus: 1990, Emotion and Adaptation. In Pervin (Ed.), *Handbook of Personality: Theory & Research*, New York NY: Guilford Press, 609-637.

Verwey, W., H. Veltman: 1996, Detecting Short Periods of Elevated Workload: a Comparison of Nine Workload Assessment Techniques. *Journal of Experimental Psychology: Applied*, **2**(3), 270–285.

Wiederhold, B., D. Jang, M. Kaneda, I. Cabral, Y. Lurie, T. May, M Wiederhold, and S. Kim: 2003, An Investigation into Physiological Responses in Virtual Environments: an Objective Measurement of Presence. In G. Riva and C. Galimberti, Eds., *Towards Cyberpsychology: Minds, Cognitions and Society in the Internet Age*. Amsterdam, The Netherlands:IOS Press, 175-184.

Wells, C.: 1986, *The Meaning Makers: Children Learning Language and Using Language to Learn*. Portsmouth, NH: Heinemann.

Witten, I., and E. Frank: 2005, *Data Mining: Practical Machine Learning Tools and Techniques*, 2nd Edition, San Francisco, CA: Morgan Kaufman.

Zachos, P., L. Hick, W. Doane, and C. Sargent: 2000, Setting Theoretical and Empirical Foundations for Assessing Scientific and Discovery in Educational Programs. *Journal of Research in Science Teaching* **37**(9), 938-962.

Zimmerman, B.: 2000, Self-efficacy: An Essential Motive to Learn. *Contemporary Educational Psychology* **25**, 82-91.

Authors' Vitae

Scott W. McQuiggan
Scott W. McQuiggan is a Ph.D. candidate in Computer Science at North Carolina State University. He received his B.S. degree in Computer Science from Susquehanna University in 2003. His primary interests lie in the areas of intelligent tutoring systems, affective computing, and machine learning. The joint research with his co-authors in this article reflects an interest in investigating the development of affective student modeling approaches for intelligent tutoring systems.

Bradford W. Mott
Dr. Bradford W. Mott is a software engineer at Emergent Game Technologies. He received his B.S. degree in Computer Engineering and his B.S., M.C.S., and Ph.D. degrees in Computer Science from North Carolina State University. Dr. Mott has worked in several areas of artificial intelligence including intelligent tutoring systems, user modeling, and computational linguistics. His long-term interests in game technologies include the application of AI to interactive entertainment.

James C. Lester
Dr. James C. Lester is Associate Professor of Computer Science at North Carolina State University, where he directs the IntelliMedia Center for Intelligent Systems. Dr. Lester received his B.A. degree in History from Baylor University and his B.A. (Phi Beta Kappa), M.S.C.S., and Ph.D. degrees in Computer Sciences from the University of Texas at Austin. Dr. Lester has worked in several areas of artificial intelligence including computational linguistics, intelligent tutoring systems, and intelligent user interfaces.

Contents

Figures

Tables