Early Prediction of Student Frustration

Scott W. McQuiggan, Sunyoung Lee, and James C. Lester

Department of Computer Science, North Carolina State University, Raleigh, NC 27695
{swmcquig, slee7, lester}@ncsu.edu

Abstract. Affective reasoning has been the subject of increasing attention in recent years. Because negative affective states such as frustration and anxiety can impede progress toward learning goals, intelligent tutoring systems should be able to detect when a student is anxious or frustrated. Being able to detect negative affective states early, i.e., before they lead students to abandon learning tasks, could permit intelligent tutoring systems sufficient time to adequately prepare for, plan, and enact affective tutorial support strategies. A first step toward this objective is to develop predictive models of student frustration. This paper describes an inductive approach to student frustration detection and reports on an experiment whose results suggest that frustration models can make predictions early and accurately.

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 [6], detecting frustration and stress [5, 22], modeling agents' emotional states [1, 9], devising affectively informed models of social interaction [10, 13, 19, 21], detecting student motivation [7], and diagnosing and adapting to student self-efficacy [3, 14]. All of this work seeks to increase the fidelity with which affective and motivational processes are modeled and utilized in intelligent tutoring systems in an effort to increase the effectiveness of tutorial interactions and, ultimately, learning. One such effort is to detect and provide support to students experiencing negative affect.

Intelligent tutoring systems should provide support that helps students cope with emotions such as anxiety and frustration and, if possible, increasing their tolerance for frustrating learning situations. Because feelings of anxiety or frustration can divert students' attention from learning tasks by causing them to become fixated on the source of frustration or causing them to worry excessively about failure, it is important that learning environments employee mechanisms to diagnosis situations that are likely to create anxious or frustrated students. Early detection of negative affective states could permit ITSs sufficient time to enact corrective "affective scaffolding" strategies. In this paper we focus on accurately predicting student frustration as early as possible by utilizing established affective computing techniques for obtaining student affective state information, including physiological response data [5, 6, 14, 22] and student self-reports [6, 7, 14].

This paper is structured as follows. Section 2 discusses affect recognition and the role of anxiety and frustration in learning. The modeling techniques used to detect frustration are described in Section 3. Section 4 introduces CRYSTAL ISLAND, our interactive learning environment test bed and describes the study. The results of the study are presented and discussed in Section 5, followed by concluding remarks and directions for future work in Section 6.

2 Background

2.1. Affect Recognition

Affect recognition is the task of identifying the emotional state of a user from a variety of physical and behavioral cues, which are produced in response to affective changes in the individual [20]. However, because affect is fundamentally a cognitive process in which the user appraises the relationship between herself and her environment [9, 24], affect recognition models should take into account both physiological and environmental information. For task-oriented and goal-oriented learning environments, affect recognition models can leverage knowledge of task structure and user goals to effectively reason about users' affective states. In particular, for such learning environments, affect recognition models can use appraisal theory [12] to recognize users' emotions generated in response to their assessment of how their actions and events in the environment relate to their goals.

2.2. Anxiety and Frustration

Frustration occurs when something or someone impedes an individual's progress towards a particular goal. As an emotional response, frustration is not fundamentally different from another negative affective response common to a variety of situations, anxiety. Anxiety is often more than merely an emotional response; it also consists of behavioral, cognitive, and physiological responses [23]. However, because our work focuses on interactive task-oriented learning environments [17], where the construction and achievement of goals is critical to student learning episodes, we primarily focus on frustration. Both anxiety and frustration can lead students to fixate on the impeding source of frustration, diverting attention from, and in some cases causing students to ignore, the task at hand. Anxiety particularly arises when students affectively respond to their focus on planning contingencies for potential future events. Detecting situations that will likely lead to student anxiety or frustration that in turn may eventually lead to student impasses would allow learning environments to intervene early, i.e., before the emotion is fully realized as the student approaches her threshold for the particular emotion.

Several strategies can be employed to identify levels of anxiety and frustration that are not detrimental to learning. Setting realistic expectations based on a student's abilities and observed past performance can contribute to student successes.

Encouragement, and specific feedback directed at particular behaviors, not merely global performance assessments, may help motivate students and provide them with guidance so that they can improve their self-assessment and help them cope with frustration and anxiety [18]. The central questions that must be answered are, "How can we detect and monitor anxiety and frustration levels so that our learning environments have sufficient time to plan and execute appropriate scaffolding?" and, "With what computational mechanisms can we draw inferences about the student, the task, and the environment to accurately predict student frustration?"

3 Modeling Frustration

To create models that make accurate predictions of student frustration as early as possible, we first collect training data by observing students interacting with an intelligent tutoring system. From this training data, we then induce *n*-gram models to make early predictions of student frustration. Models based on *n*-grams models are useful for early prediction because they are induced from sequences of observations, making predictions with each new observation until they arrive at the final observation of the sequence. In the final observation, concrete evidence of student affect (used as the class label) is obtained. Each prediction from an *n*-gram model is attempting to determine the affective state of the student recorded in this final observation of the sequence. In many cases, *n*-gram model predictions will converge on the correct affective state early in a sequence of observations. The point at which an *n*-gram model first begins making the correct prediction and then continues to make a correct prediction for the remainder of the sequence is known as the *convergence point*.

While sequential models such as *n*-grams allow us to make early predictions, they are not computationally well suited to large multidimensional data. To address this issue, we investigate three non-sequential modeling techniques: naïve Bayes, decision trees, and support vector machines. To enable non-sequential models to make early predictions, we exploit the results of the *n*-gram models. Utilizing convergence point information, we construct data sets that contain only those observations preceding the convergence point. Thus, models induced from the newly constructed datasets are able to make predictions of student affect long before the observation in which the student's affective state was recorded.

Section 3.1 presents *n*-gram frustration modeling, followed by Section 3.2's description of naïve Bayes, decision tree, and support vector machine frustration modeling techniques. Section 3.3 then describes the training data, which are observations of students interacting with the CRYSTAL ISLAND learning environment.

3.1 *n*-Gram Models for Early Prediction of Frustration

Given an observation sequence $O_1, O_2, \ldots, O_n$, the objective of affect recognition is to identify the student's most likely affective state E^* (i.e., frustrated or not frustrated) such that:

$$E^* = \arg\max P(E \mid O_1, O_2, O_3, \ldots, O_n)$$
$$= \arg\max P(E \mid O_{1:n})$$

where each O_i is an observation encoding the user's goals, user's action, the location at which action was performed, and physiological responses such as heart rate and galvanic skin responses. The observation sequence O_1, O_2, …, O_n, is denoted by $O_{1:n}$. Applying Bayes rule and the Chain Rule, the equation becomes:

$$E^* = \arg\max P(O_n \mid O_{1:n-1}, E) P(O_{n-1} \mid O_{1:n-2}, E) \cdot P(O_{n-2} \mid O_{1:n-3}, E) \ldots P(O_1 \mid E) P(E)$$

However, estimating these conditional probabilities is impractical – it would require exponentially large training data sets – so we make a Markov assumption that an observation O_i depends only on the affective state *E* and a limited window of the preceding observations.

We explore two *n*-gram affect recognition models for detecting student frustration, a unigram model and a bigram model. The unigram model is based on the assumption that, given the affective state *E*, O_i is conditionally independent of all other observations. Thus, the affect recognition formula for the unigram model can be simplified to:

$$E^* = \arg\max P(E) \prod_{i=1}^{n} P(O_i \mid E)$$

The bigram model is based on the assumption that, given the affective state *E* and the preceding observation O_{i-1}, O_i is conditionally independent of all other observations. Thus, the affect recognition formula for the bigram model can be simplified to:

$$E^* = \arg\max P(E) \prod_{i=1}^{n} P(O_i \mid O_{i-1}, E)$$

The resulting formulae for the unigram and bigram models are very efficient because updating the affect prediction for each new observation only requires computing the product of the probability returned by the previous prediction and the current conditional probability.

During training, we estimate P(*E*), P(O_i|*E*), and P(O_i,| O_{i-1}, E) using training data acquired with an interactive learning environment as described below. Because training data is necessarily sparse, i.e., we are unlikely to observe all possible combinations of actions, locations, goals, and physiological response levels, the unigram and bigram models employ a standard smoothing technique (a flattening constant and simple Good-Turing frequency estimation [8] to re-evaluate zero-probability and low-probability *n*-grams.

3.2 Naïve Bayes, Decision Tree, and SVMs for Modeling Frustration

Naïve Bayes, decision tree, and support vector machine classifiers are effective machine learning techniques for generating preliminary predictive models. Bayes classification approaches produce probability tables that can be implemented in runtime systems and used to continually update probabilities for predicting student affective states, and, in the approach proposed here, for predicting whether students are frustrated or not. 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 diagnosing particular student emotions. Support vector machines (SVM) are also particularly effective at handling high-dimensional data. SVMs search for hyperplanes that linearly separate data into classes (affective states).

These classification techniques are particularly useful for inducing models with large multidimensional data, such as the data gathered in the user study described below. 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. We have used the WEKA machine learning toolkit [25] to analyze naive Bayes, decision tree, and SVM approaches for generating models of student affect to predict student frustration as early as possible.

3.3 Training Data

Accurately modeling user affect requires a representation of the situational context that satisfies two requirements: it must be sufficiently rich to support assessment of changes in affect, 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 [9, 24] affect recognition models should take into account both physiological and environmental information. In task-oriented interactive learning environments, task structure is often explicit. To effectively reason about students' affect, affect models can leverage this knowledge of task structure as well as the state of the student's progress through a learning episode and physiological responses to unfolding events. In particular, affect models can rely on concepts from appraisal theory [12] to recognize student emotional responses generated from their assessment of their progress in the learning task. Thus, affect models can leverage representations of the information observable in the learning environment – note that this is the same information that students may use in their own appraisals – to predict student emotion, particularly frustration.

Therefore we employ 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 model induction and model usage. During model induction, the observational attribute vector is passed to the affect learner for model generation; during runtime operation, the attribute vector is monitored by runtime components that utilize knowledge of user affect to inform effective pedagogical decisions. The observable attribute vector represents four interrelated categories of features for making decisions:

- **Temporal Features**: The interactive learning environment 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**: The interactive learning environment continuously monitors the location of the student's character. It monitors locations visited in the past, and locations recently visited. There are 45 designated locations in the interactive learning environment test bed (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 inform the prediction of students' learning task progression and associated affective responses.
- **Intentional Features**: The interactive learning environment continuously tracks goals being attempted (in the interactive learning environment described below goals are explicitly presented to students by an onscreen textual display), 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. Here we circumvent the problem of goal recognition [4, 16] through explicit delivery of student goals. Integrating goal recognition and affect recognition is a promising direction for future work.
- **Physiological Response**: The interactive learning environment 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.

During training sessions later used for 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.

4 Evaluation

4.1 Crystal Island

To serve as an effective "laboratory" for studying user affect recognition in an interactive task-oriented learning environment, a test bed should pose the same kinds of challenges that affect recognition modelers are likely to encounter in future runtime learning environments. It should offer users a broad range of actions to perform and provide a rich set of tasks and goals in a nontrivial task-oriented virtual learning environment. The goals should exhibit some complexity, and the environment should be populated by manipulable artifacts and be inhabited by multiple characters. To this end, we have devised CRYSTAL ISLAND, a task-oriented learning environment test bed featuring a science mystery in the domain of genetics. The mystery is 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 role of the daughter (or son) of a visiting scientist who is attempting to discover the origins of an unidentified illness at the research station. The environment begins by introducing the student 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 outbreak. She is free to explore the world to collect physical evidence and interact with other characters. Through the course of her adventure she must gather enough evidence to correctly choose among candidate diagnoses including botulism, cholera, salmonellosis, and tick paralysis as well as identify the source of the disease relying on her knowledge of microbiology to solve the mystery.

The task-oriented learning environment of CRYSTAL ISLAND, the semiautonomous characters that inhabit it, and the user interface were implemented with Valve Software's Source™ engine, the 3D game platform for Half-Life 2. In CRYSTAL ISLAND, the user can perform a broad range of actions including performing experiments in the laboratory, interacting with other characters, reading "virtual books" to obtain background information on diseases, and collecting data about the food recently eaten by the members of the research team. Throughout the mystery, users can walk around the island and visit the infirmary, the lab, the dining hall, and the living quarters of each member of the team. In the current test bed, there are 20 goals users can achieve, three hundred unique actions the user can carry out, and over fifty unique locations in which the actions can be performed.

4.2 User Study

There were 5 female and 31 male participants varying in age, race, and marriage status. Approximately 44% of the participants were Asian, 50% were Caucasian, and 6% were of other ethnicities. Participants' average age was 26.0 (SD=5.4).

After filling out a consent form and demographic survey, participants began training sessions by first completing a practice task. The practice task allowed them to become familiar with the keyboard and mouse controls as well as interacting in a 3D virtual environment. Following the practice task, participants were presented a controlled backstory for CRYSTAL ISLAND situating them on the island and providing details about their task. For reference, participants had access to a cast of agents found in the CRYSTAL ISLAND environment as well as an overview map. Participants then interacted with the environment to solve the science mystery. The training test bed provided them with specific goals to focus on, guiding them through the solution to the mystery.

Self-reporting mechanisms are frequently used in studies to obtain information about a subject's affective state [6, 7]. In the study reported here, periodically (every 75 seconds) a "self-report emotion dialog" box would appear requesting input from participants about their affective state. They were asked to select the emotion, from a set of six emotions (*excitement*, *fear*, *frustration*, *happiness*, *relaxation*, and *sadness*), that best summarized their own feelings since they were previously asked to assess their emotional state. This set of emotions was chosen to cover the affect space [11] so that most subjects would easily be able to relate their feelings during interaction to one of the six affective states. In addition to periodic reports, participants had the ability to trigger the self-report emotion dialog if they felt compelled to report a change in their affective state. This functionality proved in practice to be used sparingly. After solving the science mystery, participants completed a post-experiment survey before exiting the training session.

5 Results

Unigram, bigram, naïve Bayes, decision tree, and SVM affect recognition models for detecting student frustration were learned from the collected datasets.

The *n*-gram models were evaluated using the following the criteria [4]:

- *Accuracy*: Ratio of correct predictions to the total number of observations.
- *Converged*: Percentage of observation sequences in which the goal recognizer's final prediction is correct.
- *Convergence Point*: For observation sequences which converged, the point within the sequence when the affect recognizer started making the correct prediction and continued to make the correct prediction for the remainder of the sequence.
- *Average Observations of Converged*: Average number of observations contained in observation sequences in sequences which converged.

The induced n-gram models were tested using the standard k-fold cross validation evaluation methodology [15], with k=10. (In each fold, nine segments are used for training and one, which is held out of training, is used for testing.) The results of n-gram affect recognition models are presented in Table 1. Figure 1 shows a bigram convergence graph depicting the amount of data (actions) required by the model to converge on the correct affective state and the associated probability of that emotion classification. Note that the bigram model utilizing a flattening constant converged after consuming 6.5% of the records leading up to the student self-reported affective state (the class label). In instances where n-gram models converged the models were able to correctly classify whether the student was frustrated, on average, 35 seconds prior to the self-report.

Table 1. *n*-gram Results.

	Unigram Flattening Constant	Unigram Good-Turing	Bigram Flattening Constant	Bigram Good-Turing
Accuracy	68.5%	73.4%	73.6%	73.5%
Converged	39.7%	67.1%	67.8%	67.2%
Converged Point	22.6%	7.1%	6.5%	6.9%
Average Observations of Converged	54.3	51.7	51.8	51.8

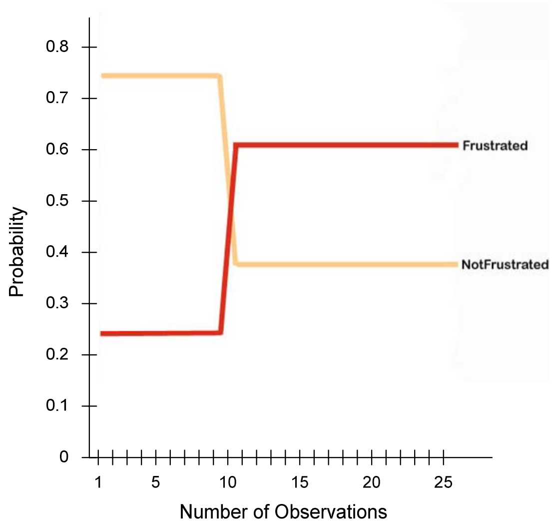

Fig. 1. Bigram convergence graph.

The results of the naïve Bayes, decision tree, and support vector machine (SVM) affect recognition models are presented in Table 2. Using the results of the n-gram analysis in which bigram models converged after 6.5% of observations contained in a defined window, we constructed datasets containing only these records for naïve Bayes, decision tree, and SVM model induction. Let us consider the observations O_1, O_2, …, O_n, as the observations leading up to a student self-reported affective state used by n-gram model induction. The data set consisting of observations O_1, O_2, …, O_m, where m equals n × 0.065, leading to each self-report are then used to induce naïve Bayes, decision tree, and SVM affect models. Thus, all induced models are able to predict student affective states (i.e., whether students are frustrated or not) early, i.e., long before we receive confirmation of the student's affective state from self-reports. These early prediction datasets contain the same data n-gram models consumed up to the convergence point. The constructed dataset allows induced models to make the same early predictions as the n-gram models, approximately 35 seconds prior to the self-reported affective state.

Table 2. Induced model results.

	Unigram Flattening Constant	Unigram Good-Turing	Bigram Flattening Constant	Bigram Good-Turing	Naïve Bayes	SVM	Decision Tree
Accuracy	68.5%	73.4%	73.6%	73.5%	75.7%	82.2%	88.8%
Precision	60.1%	60.3%	61.6%	60.8%	76.3%	82.2%	88.7%
Recall	52.6%	59.6%	60.3%	59.9%	75.7%	81.9%	88.9%

Below 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. To analyze the performance of induced models we first establish a baseline level. Because six affective states were reduced to a two-class predictive classifier (frustrated vs. not frustrated), we consider chance as a baseline measure of performance. If our baseline model were to predict the most frequent classifier (not frustrated, n=3859), then the model would correctly predict a student's frustration state 65% of the time. Using this model as a baseline, we observe that all induced models outperform the baseline model. The lowest performing induced model, a unigram model using a flattening constant accurately predicted 68.5% of instances correctly in testing. This performance is statistically significantly better than the baseline (likelihood ratio, $\chi^2 = 16.075$, $p = 6.089 \times 10^{-5}$, and Pearson, $\chi^2 = 16.067$, $p = 6.1 \times 10^{-5}$, df = 1). Thus, the performance of all induced models is a statistically significant improvement over the baseline. On the high performing end, the induced decision tree model performed best, accurately predicting 88.8% of all test instances. This is statistically significant compared to the baseline (likelihood ratio, $\chi^2 = 980.87$, $p = 2.6 \times 10^{-215}$, and Pearson, $\chi^2 = 943.92$, $p = 2.8 \times 10^{-207}$, df = 1), and is also statistically significant compared to the next highest performing model, the induced SVM model (likelihood ratio, $\chi^2 = 105.28$, $p = 1.06 \times 10^{-24}$, and Pearson, $\chi^2 = 104.49$, $p = 1.58 \times 10^{-24}$, df = 1).

The experiment has two important implications for the design of runtime student frustration modeling. First, by monitoring student physiological response, the student's learning task, and events unfolding in the learning environment, induced models can make early, accurate predictions of forthcoming student frustration. Second, using models that can make early predictions of student frustration creates a significant window of opportunity for the learning environment to take corrective action; early-prediction models offer an improvement over traditional approaches that predict affective states and self-reports on a moment-by-moment basis.

6 Conclusion and Future Work

Recent advances in affective reasoning have demonstrated that emotion plays a central role in human cognition and should therefore play an equally important role in the human-computer interaction, especially in intelligent tutoring systems. To support effective, enjoyable tutorial interactions, affect-informed systems must be able to accurately and efficiently recognize user affect from available resources, including negative affect. Following appraisal theory, representations of users' actions and goals enable affect recognition models to consider the same relationship that users continually assess in order to predict their affective states.

This paper has introduced an inductive approach to generating affect recognition models for early detection of student frustration. In this approach, models are induced from observations of students interacting within a task-oriented learning environment in which student actions, locations, goals, and temporal information are monitored. After problem-solving traces have been recorded, affect recognition models are induced that are both accurate and efficient.

The findings reported here contribute to the growing body of work on affective reasoning for learning environments. In the future, it will be useful to investigate the issue of individual differences in tolerance levels for anxiety and frustration. Because students can tolerate different levels of anxiety and frustration, in the future, learning environments should be equipped to adapt to this variance in students' negative affect thresholds. To determine how much frustration a student can persist through, it may be possible to utilize models of student self-efficacy [14]. Self-efficacy has been identified as a predictor of persistence and levels of student effort [2]. Determining frustration thresholds may allow learning environments to monitor students' persistence, intervening only when necessary, so that student efficacy and affect are maintained at levels that best support effective learning. Extending early predictive models to a broader range of emotions could contribute to more comprehensive models of student affect. Evaluating the resulting models as runtime control components in task-oriented learning environments is a critical next step in investigating affect-informed interactive learning.

Acknowledgements

The authors would like to thank the members of the IntelliMedia Center for Intelligent Systems at North Carolina State University for their contributions to the implementation of Crystal Island. The authors also wish to thank Valve Software for authorizing the use of their Source™ engine and SDK. 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

1. André, E., and Mueller, M. (2003) Learning affective behavior. In *Proceedings of the 10th International Conference on Human-Computer Interaction.* Lawrence Erlbaum, Mahwah, NJ,, 512-516.
2. Bandura, A. (1997) *Self-efficacy: The exercise of control.* New York: Freeman..
3. Beal, C. and Lee, H. (2005) Creating a pedagogical model that uses student self reports of motivation and mood to adapt ITS instruction. In *Workshop on Motivation and Affect in Educational Software, in conjunction with the 12th International Conference on Artificial Intelligence in Education.*

4. Blaylock, N. and Allen, J. (2003) Corpus-based, statistical goal recognition. In *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence*, Acapulco, Mexico, pp.1303-1308.
5. Burleson, W., and Picard, R. (2004) Affective agents: Sustaining motivation to learn through failure and a state of stuck. In *Proceedings of the ITS Workshop of Social and Emotional Intelligence in Learning Environments*, Maceio, Alagoas, Brazil.
6. Conati, C., and Mclaren, H. (2005) Data-driven refinement of a probabilistic model of user affect. *Tenth International Conference on User Modeling*. New York, NY, pp. 40-49.
7. de Vicente, A., and Pain, H. (2002) Informing the detection of the students' motivational state: an empirical study. In *Proceedings of the 6th International Conference on Intelligent Tutoring Systems*. Springer-Verlag, New York, NY, pp. 933-943.
8. Gale, A. and Sampson, G. (1995) Good-Turing frequency estimation without tears. *Journal of Quantitative Linguistics*, 2(3): 217-237.
9. Gratch, J., and Marsella, S. (2004) A domain-independent framework for modeling emotion. *Journal of Cognitive Systems Research*, 5(4), 269-306.
10. Johnson, L., and Rizzo, P. (2004) Politeness in tutoring dialogs: "Run the factory, that's what I'd do". *7th International Conference on Intelligent Tutoring Systems*, Maceio, Brazil, pp.67-76.
11. Lang, P. (1995) The emotion probe: Studies of motivation and attention. *American Psychologist*, 50(5), 372-285.
12. Lazarus, R. (1991) *Emotion and Adaptation*. Oxford University Press, New York, NY.
13. McQuiggan, S., and Lester, J. (2006) Learning empathy: A data-driven framework for modeling empathetic companion agents. In *Proceedings of the 5th International Conference on Autonomous Agents and Multiagent Systems*, Hakodate, Japan, 961-968.
14. McQuiggan, S., and Lester, J. (2006) Diagnosing self-efficacy in intelligent tutoring systems: An empirical study. In *Proceedings of the 8th International Conference on Intelligent Tutoring Systems*, Jhongli, Taiwan, pp. 565-574.
15. Mitchell, T. (1997) *Machine Learning*, McGraw-Hill, OH.
16. Mott, B., Lee, S., and Lester, J. (2006) Probabilistic goal recognition in interactive narrative environments. In *Proceedings of the Twenty-first National Conference on Artificial Intelligence*, Boston, MA, 187-192.
17. Mott, B., and Lester, J. (2006) Narrative-centered tutorial planning for inquiry-based learning environments. In *Proceedings of the 8th International Conference on Intelligent Tutoring Systems*, Jhongli, Taiwan, pp. 675-684.
18. Ormrod, J. (2002) *Educational Psychology: Developing Learners*, (4th ed.). Prentice Hall, Upper Saddle River, NJ.
19. Paiva, A., Dias, J., Sobral, D., Aylett, R., Woods, S., Hall, L., and Zoll, C. (2005) Learning by feeling: evoking empathy with synthetic characters. *Applied Artificial Intelligence*, 19, 235-266.
20. Picard, R. (1997) *Affective Computing*. MIT Press, Cambridge, MA.
21. Porayska-Pomsta, K., and Pain, H. (2004) Providing cognitive and affective scaffolding through teaching strategies: applying linguistic politeness to the educational context. *Seventh International Conference on Intelligent Tutoring Systems*, Maceio, Alagoas, Brazil, pp. 77-86.
22. Prendinger, H., and Ishizuka, M. (2005) The empathic companion: a character-based interface that addresses users' affective states. *Applied Artificial Intelligence*, 19, 267-285.
23. Seligman, M., Walker, E. and Rosenhan, D. (2001) *Abnormal psychology*, (4th ed.) New York: W.W. Norton & Company, Inc.
24. Smith, C., and Lazarus, R. (1990) Emotion and adaptation. In Pervin (Ed.), *Handbook of Personality: theory & research* (pp. 609-637). NY: Guilford Press.
25. Witten, I., and Frank, E. (2005) *Data Mining: Practical machine learning tools and techniques*. 2nd Edition, Morgan Kaufman, San Francisco, CA.