Proceedings of the Twenty-Ninth International
Florida Artificial Intelligence Research Society Conference

Towards Real Time Detection of Learners' Need of Help in Serious Games

Ramla Ghali[1], Claude Frasson[1] and Sébastien Ouellet[2]

Département d'Informatique et de Recherche Opérationnelle
Université de Montréal, Montréal, Canada H3C 3J7
[1]{ghaliram,frasson}@iro.umontreal.ca; [2]{sebouel}@gmail.com

Abstract

Providing an adequate help to a learner remains a challenge. In this paper we aim to find how to provide learners with real time help in an educational game. Detecting that a player is engaged or motivated is a good sign that he is progressing. For these reasons we need to assess learner's states while learning. In this study we gather a variety of data using three types of sensors (electroencephalography, eye tracking and automatic facial expression recognition) to build a reliable user adaptation system. The data result from an interaction of 40 players with LewiSpace game, that we built for experimental purpose to learn construction of Lewis diagrams. We used machine learning algorithms in order to identify the most important features gathered from each sensor. Two models were trained with these data: a generalized model, trained on all data available, and a personalized model, trained only on the current user during an early phase of the game experience. The predictive results showed that personalized model could outperform the generalized model.

Introduction

Serious games are environment that teach, train and inform players. Mainly, they combine two fundamental aspects: (1) fun aspect, and (2) educational content. In the last decade, researchers showed that these tools are very beneficial for learning purposes (Prensky 2001, Jackson et al. 2012). Therefore, we noticed that this learning environment has spread and concerned many population in different kinds of courses: Physics (Shute et al. 2013), data structure (Derbali & Frasson 2012), and medicine (Lester et al. 2014).

However, a main problem of these tools is that of focusing more on the playful aspect rather than the educational content. For that, detecting when users need more help or challenge, is very important. This task is very delicate and requires that the game detects learners' emotions, engagement and motivation. For instance, emotions could be used to detect if the learner is bored or frustrated and therefore the game should react by changing the learner's state and offer him more encouragement and help if needed. In the other case, if the game detects that learner is very excited it may be an indication that he needs more challenge (no help is required but learner should be in an ascending difficulty level of the game). Moreover, detecting that the player is engaged would be a good sign that he is progressing while playing the game. To summarize, in order to develop more reliable serious games (SG), it would be necessary to detect all the previous mentioned points and react accordingly.

In this paper we are interested in detecting exactly if learners need help in a LewiSpace game. The main objective of this paper is to provide learner with an adequate help when needed in order to avoid gaming behavior. To do that, we conducted an experiment where we collected data from three sensors (Electroencephalogram, Eye tracking, and FaceReader) and integrated them into machine learning algorithms. At the end of the game, the players were submitted to a self-report questionnaire on the need of help, for each mission of the game.

We will adopt two approaches: (1) a **generalized approach** where we train machine learning algorithms **offline on all the data acquired**, then we predict users' need of help **online** on the current participant, and (2) a **personalized approach** where we train several models in **real time** for each user at the beginning of the game session using two types of tasks (an easy and a difficult task). The prediction will be done also **online** using different descriptive vectors for this participant.

Related Work

In the community of Intelligent Tutoring Systems (ITS) or Serious Games (SG), researchers are always interested in improving these environments in order to offer learners with more adequate content. To do this, they are interested

Copyright © 2016, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

to different kinds of problems, such as enhancing comprehension, improving motivation and engagement, increasing positive emotional states which are more effective for learning, etc. (Baker et Rosso 2013, Ghali et al. 2015b). The main purpose of these works is to create adaptive environments for learning.

For instance, D'Mello and his team (D'Mello et al. 2012) have used *eye tracking* data to dynamically detect emotions of boredom and disengagement. Dynamic tracking of eye movements was integrated into a tutor that identifies when a learner is bored. In the case of student disengagement, the tutor tries to speak and attract the learner's attention. Besides, (Conati 2002) proposed a probabilistic model to monitor user's emotions and engagement in their educational game, PrimeClimb. Recently, (Ghali et al. 2015b) were interested to **unsolicited feedback**: whether or not students **need help** while interacting with an educational game.

Despite the effort of researchers, developing an adaptive tool (ITS or SG) remains a great problem because it depends on different factors, the actual works are mainly based on two approaches: (1) adaptation according learners' behaviors, and (2) adaptation according a physiological sensing approach. For instance, in the first category, we can cite as examples of adaptation: mouse movements, learner's behaviors (off-task and gaming behaviors), time response, learner's emotions, etc. Whereas, in the second category, we can mention as examples the mental states of distraction, workload, and engagement extracted from EEG, learner's gaze data, arousal and valence (of emotions), skin conductivity, etc.

Besides, researchers prove that efficient **help seeking behavior** (students' request of help) can improve learning outcomes and reduce learning duration (Wood & Wood 1999). However, the abuse of help seeking can reduce learner's performance and reflect more gaming behavior (Baker et al. 2011). For all these reasons, we focus on the following to detect if a learner really needs help in SG. To do that we studied the possibility of integrating two machine models (Support Vectors Machine (SVM) and K Nearest Neighbors (KNN)) used with a simple feature extraction to detect when it is necessary to provide learner with more help and pedagogical content in our educational game, described in the next section.

A brief description of our game: LewiSpace

LewiSpace is a game intended to teach Lewis diagrams for college students. For a detailed description, the reader is referred to (Ghali et al. 2015a, Ghali et al. 2015b).

Our game is a puzzle-game designed using Unity 4.5 (a 3D environment) integrating EEG and Eyetracking sensors data using the Emotiv SDK v2.0 LITE and the Tobii SDK 3.0. In this game, the learner appears as an astronaut exploring a planet's surface. The astronaut falls into a cavern and for surviving he has to accomplish five missions elaborated in an ascending order of difficulty (see table 1).

Table 1. Missions' distribution in LewiSpace game

Missions	Molecules to construct
Mission 1	Produce water (H2O)
Mission 2	Produce methane gas (CH4)
Mission 3	Produce a sulfuric acid (H2SO4)
Mission 4	Craft a refrigerant (C2F3Cl)
Mission 5	Refuel the fuel tank with ethanol (C2H6O)

In each mission, he has to gather atoms randomly distributed in the environment and construct then a molecule's Lewis diagram in order to overcome obstacles, progress in the game and find his lander to return on the earth. The tool for constructing molecules is presented in figure 1.

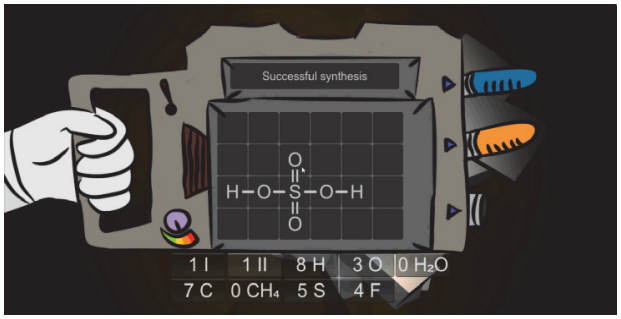

Figure 1. Lewis diagram of the sulfuric acid

Our approach: physiological sensing

In this section, we describe in detail our proposed approach for realizing real time user's adaptation in our game LewiSpace. The adaptation will be done according to learner's need of help (known as **unsolicited feedback**). In fact, as mentioned before, we use a multimodal approach that combines mainly three types of non-intrusive sensors: **electroencephalography** (EEG) for different bands and indices extraction (more specifically the emotions' classification using the Affectiv Suite provided by Emotiv EPOC (Gherguluscu 2014), **eye tracking** for tracking eyes' motions (more precisely pupil diameter (Bartel & Marshall 2012) to measure the mental state of workload), and **facial expression recognition** using FaceReader (seven basic emotions defined by Ekman (Ekman 1970)). Moreover, for users' need of help we used at the end of our game a **self-report questionnaire** that identify if the users need or not more of help in each mission of our game ranging from 1 (no help required) to 3 (more help required).

To build and validate our approach, we conducted an **experiment** and collected data using the above mentioned types of sensors, where **40** students (25 males and 15 females) participated voluntarily in the study and had no prior knowledge about Lewis diagrams. The participants were asked to play LewiSpace while we collected mainly three types of physiological data:

- EEG data with a sampling rate of 128Hz at a second (using Emotiv EPOC),
- pupil diameter to measure workload (Bartels et al. 2012) using eyetracking (Tobii Tx300), and
- 7 basic emotions: happy, sad, angry, surprised, scared, disgusted, and neutral with their valence and arousal (using FaceReader).

The experiment is respectively preceded and finished by a pre and post test to see if the student learnt some concepts from the game. After collecting these data, we merged them and removed the noisy ones. This step (data preprocessing) will be described in detail further (section 5). Then, we developed and trained different statistical machine learning algorithms in order to find the best algorithm that provides the highest accuracy. All the algorithms were developed using the Python library, the **Scikit-learn** (Pedrogosa et al. 2011). After training these algorithms, we select from each algorithm the most important features for modelling purposes in order to reduce our feature vectors and take only into consideration those that have an impact on varying the model's accuracy. We validate our models using **leave-one-out method** and **cross-validation**. Finally, after building and validating the statistical machine learning models, a decision stage consists at selecting the best model with the highest accuracy and investigate the most potentially useful features from it (figure 2).

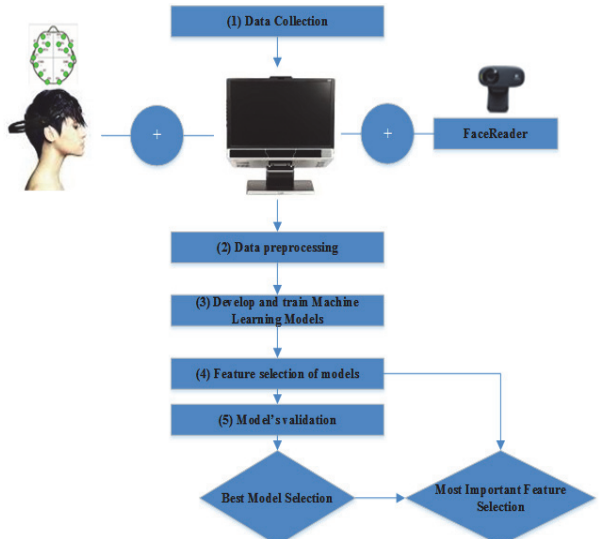

Figure 2. The physiological sensing approach

Data preprocessing, results and discussion

We eliminated seven participants' data (out of forty) due to technical errors that happened during the recording of the sessions, making data unavailable for analysis (some data segments were lost during the recording, in a manner unrelated to the participants). Fifteen signals are used throughout our analysis: pupil diameter, five Affectiv indices (short-term excitement, long-term excitement, meditation, boredom, and frustration), and nine facial expression indices (neutral, sad, happy, angry, surprised, scared, disgusted, valence (of the current facial expression), and arousal). At the end of the session, participants filled out an evaluation task indicating if more or less help was needed in each mission. The scale ranges from 1 to 3, where 1 indicates that the mission was too easy (no help required), 2 that it was adequate, and 3 that the mission was too hard (help is required). Signals (EEG, eyetracking and facial expressions recognition) occurring before a mission was completed (and after the completion of the mission) were labeled with the category self-reported by the participant (help or no help required for each mission as described above). Each mission can be further segmented into trials, where one trial is the time spent between two attempted answers for a given mission (in cases where the player immediately grasps the solution, one mission would be such a sequence (or a trial), but it is expected that many players will produce a few incorrect answers before completing the mission). In a total, **633** such **sequences** (or samples) were gathered. We extracted *four statistical parameters* for each signal, producing **feature vectors** with **60 dimensions** (4 * 15 outputs of signals) for each sequence. The four parameters are the mean, the standard deviation, the minimum, and the maximum of each signal for a given duration of time.

For feature and model selection of the best machine learning algorithm, we considered two contexts of analysis (and their application): 1) **a generalized model** that consists of training **offline** a single model on **all previous participants** before a game session and using its predictions for a new user, and 2) **a personalized model** that consists of training multiple models, one for each current user, **in real-time** at the beginning of a game session using two types of tasks (easy task: construct H2 molecule, and a hard one: construct CH4N2O2 molecule). These tasks will be used to calibrate the personalized model.

For the first context, we used a **leave-one-participant-out scheme**, where a model was trained on all participants' data except for the current participant (i.e. the one on which the model was tested). The 633 sequences mentioned above were reduced to 60-dimensional vectors and used as samples. For the second context, we gathered a training sample consisting of a single sequence of each class, testing all other sequences, for a single participant's

data. The samples were overlapping time slices of the signals, and each slice was reduced to a 60-dimensional vector.

For the first context, **Support Vector Machine models** (the best algorithm obtained with highest accuracy) were used as classifiers, with the best hyper-parameters selected through **a grid search** through C and gamma (C of 10^{-6} to 10^{4} and Gamma of 10^{-9} to 10^{3}).

For the first context, we tested models by ignoring some features during the training and classification tasks, allowing us to compare the accuracy of each model depending on which available features. Table 1 presents those results, showing that the **Affectiv indices** are the **best features (54.1%)**.

Table 2. Balanced accuracies depending on feature selection

	All features	No pupil diameter	No Affectiv indices	No facial expression	Only Affectiv indices
Correct (%)	53.4	53.1	46.1	53.8	**54.1**

As for the second context, table 3 presents the confusion matrix for the best model found using a **cross validation**. The best algorithm found was **KNN** with a **Euclidean distance** and **7 neighbors**. Horizontal values are true labels, and vertical values are predicted labels.

Table 3. Confusion matrix for a personalized model

	Too easy	Adequate	Too hard	Total
Too easy	**0.776**	0.034	0.190	58
Adequate	0.092	**0.515**	0.392	291
Too hard	0.101	0.165	**0.733**	907
Total	164	302	790	**0.675**

Conclusion

In the current paper, we described a study aimed at exploring whether we could predict in real time if users needed help to solve problems in LewiSpace. We suggested two approaches where machine-learning models can be trained well enough as to offer usefully accurate predictions, using widely available algorithms and a simple feature extraction process from physiological sensors.

Future work will consist of developing a second study where our game is augmented with a combination of the two proposed approaches, testing the impact of unsolicited feedback in learners' performance.

Acknowledgements

We acknowledge the CRSH, more precisely LEADS project, and NSERC for funding this work.

References

Baker, R. S., & Rossi, L. M. 2013. Assessing the Disengaged Behaviors of Learners. *Design Recommendations for Intelligent Tutoring Systems,* 155.

Baker, R.S.J.D, Godwa, S.M., Corbett, A.M. 2011. Automatically detecting a student's preparation for future learning: Help use is key. *4th International conference of Data mining*, 179-188.

Bartels, M., Marshall, S. P. 2012. Measuring Cognitive Workload Across Different Eye Tracking Hardware Plateforms. *ETRA 2012,* Santa Barbara, CA.

Conati, C. 2002. Probabilistic assessment of user's emotions in educational games, *Journal of Applied Artificial Intelligence (AAI) ,* vol. 16, no. 7-8, pp. 555-575, 2002.

Derbali, L., Frasson, C. 2012. Exploring the Effects of Prior Video-Game Experience on Learner's Motivation during Interactions with HeapMotiv. *The 11th International Conference On Intelligent Tutoring Systems (ITS 2012).* Chania, Greece. June 14-18.

D'Mello, S., Olney, A., Williams C. and Hays, P. 2012. "Gaze tutor: A gaze-reactive intelligent tutoring system." *International Journal of Human-Computer Studies* 70(5): 377-398.

Ekman, P. 1970. Universal facial expressions of emotion. *California Mental Health Research Digest,* 8, 151-158.

Ghali, R., Ouellet, S., Frasson, C., 2015a. LewiSpace: An Exploratory Study with a Machine Learning Model in an Educational Game". *Journal of Education and Training Studies,* vol. 3, no. 6, November 2015.

Ghali, R., Ouellet, S., Frasson, C. 2015b. LewiSpace: an educational Puzzle Game with a Multimodal Machine Learning Environment. *KI 2015: the 38th German Conference on Artificial intelligence, short paper, Dresden, Germany, September 21-25.*.

Gherguluscu, I., Muntean, C. H. 2014. A Novel Sensor-Based Methodology for Learner's Motivation Analysis in Game-Based Learning. *Interactive with Computers,* 26(4).

Jackson, G. T., Dempsey K.B. and McNamara D.S. (2012). Game based Practice in a Reading Strategy Tutoring System: Showdown in iSTART-ME. *Computer games:* 115-138.

Jaques, N., Conati, C., Harley, J., and Azevedo, R. 2014. *Predicting Affect from Gaze Data During Interaction with an Intelligent Tutoring System. ITS conference 2014.*

Lester, J., Spires, H., Nietfeld, J., Minogue, J., Mott, W. and Lobene, E. 2014. Designing game-based learning environments for elementary science education: A narrative-centered learning perspective. *Information Sciences* (264): 4-18.

Pedregosa et al. 2011. Scikit-learn: Machine learning in Python, *JMLR* (12): 2825-2830.

Prensky, M. 2001. *Digital game based learning.* New York: McGraw-Hill.

Shute, V., Ventura, M., Kim, Y.J. 2013. Assessment and learning of quantitative physics in Newton's playground. *Journal of Education Research,* 106, 423-430.

Wood, H., Wood, D. 1999. Help seeking, learning and contingent tutoring. *Computers and education.* 33, 153-169.