Training a Gaming Agent on Brainwaves

Bartolomé Francisco, Moreno Juan, Navas Natalia, Vitali José,
Ramele Rodrigo, *Member, IEEE,* Santos Juan Miguel

***Abstract*—The field of Brain Computer Interfaces (BCI) has seen a bloom in the past few years. One of its applications is the training of systems using signals originated from the human brain. Out of all the methods used to acquire information from the central nervous system, the one that has gained most popularity is Electroencephalography (EEG), mainly because of its effectiveness, low cost and its noninvasive characteristic. Event-related potential (ERP) are brain signals found on EEG that are time-locked to a particular event. Error-related potential (ErrP) are a particular type of ERP that can be elicited by a person who attends a recognizable error. When subjects observe an agent who makes a mistake this potential can be discerned by analyzing the signals which are directly related with the cognitive interpretation of an erroneous outcome or decision. Reinforcement Learning (RL) is an Artificial Intelligence approach where an agent tries to maximize rewards obtained while interacting with an unknown environment. In this work, a gaming agent is trained using RL, and the feedback obtained from brainwaves of human critic observers. Results show that there is an effective transfer of information and that the agent learns successfully to solve the game efficiently.**

***Index Terms*—ErrP, BCI, EEG, RL, Agent, AI**

I. Introduction

THE effectiveness of today's human–machine interaction and artificial intelligence is limited by a communication bottleneck as humans are required to translate high-level concepts into a machine-mandated sequence of instructions [1], [2]. Hence, new interaction methods are required to increase the communication bandwidth between computers and human, or to produce alternative communications systems to increase the efficiency of this channel. In this respect, video games has been widely used as test tools to assess new means of interactions [3], [4]. Video gaming agents are computer programs that can sense the computer game environment, process information, and interact with it. They are used in the context of testing and evaluating artificial intelligence algorithms that aim to win the game or to behave like a real user player [5]. In this work, the feedback obtained from an observational human critic (OHC) in the form of electroencephalographic (EEG) signals is used to evaluate the operational performance of a gaming agent. Observational human critics are silent subjects observing a computer gaming agent playing the game.

Concurrently, the appealing idea of a direct interface between the human brain and an artificial system, called Brain Computer Interface (BCI) or Brain Machine Interfaces (BMI), has proved the feasibility of a distinct non-biological communication channel to transmit information from the Central Nervous System (CNS) to a computer device [6]. BCI systems provide a new input modality that can be used in the context of a computer game [7], [8], is relevant in the context of the accessibility for video games [9] and in the growing area of e-sports [10].

In this proposal, gaming agents are trained using only signals components called Error-related Potentials (ErrP) that can be identified from their brain signals. This type of signals can be found on EEG traces and are elicited when a subject is aware of the presence of an unexpected outcome, which she/he identifies as an error. It is currently an extensive area of research in the neuroscience community [11]. Error-related Potentials can be detected by signal processing and machine learning techniques [12] and they are also used in Brain-Computer Interfaces to implement or enhance artificial communication channels [13].

In order to train the agent, Reinforcement Learning (RL) [14] comes as a natural solution for this scenario from the field of Artificial Intelligence. This is an algorithmic learning strategy inspired on how biological agents learn by exploring their environment, and getting negative or positive feedback rewards. This strategy aims to maximize the amount of positive rewards while keeping the number of negative feedback low. Thus, the learning problem is posed as a stochastic optimization strategy [15]. Recently, this technique has been used extensively in the context of advances in artificial intelligence [16]. Nonneglected is the influence of DeepBrain's AlphaGo project, which was the first to reach a very high proficiency when it won the complex game Go against several world champions [17].

Previous research has explored the usage of RL with reward signals based on brain activity, recorded by an EEG-based BCI system during task execution. The papers [18], [19], [20] have successfully demonstrated that a robot can be controlled by obtaining a reward signal from a person's brain activity which is observing the robot solve a task. In the context of Brain Computer Interfaces a growing number of studies have demonstrated the feasibility of using ErrPs as rewards for RL schemes such as to enhance robotic behaviour[21], to assess air traffic controllers decissions [22] or to categorise actions as errors [23]. Other approaches have used the signal as an important feedback for human-robot interaction or to implement shared-control strategies [24]. Additionally, ErrPs have also been used in the context of games as an additional feedback channel that can be explored to improve gaming experience [25], [26].

Therefore, in this work, we aim to use the information extracted from brainwaves to enhance the performance of a gaming agent. The four contributions are (1) a very simple

R. Ramele and J.M.Santos are with the Department of Computer Engineering, Instituto Tecnológico de Buenos Aires(ITBA), Argentina, e-mail: rramele@itba.edu.ar.

Manuscript received December 9, 2019; revised August 20, 2020.

game that can elicit the ErrP potential, (2) the EEG brainwaves dataset from observational human observers, (3) results that confirm that even when ErrP classification accuracy is low and produces a noisy reward signal, that provides enough information for an agent to learn the optimal policy and solve the simple game and (4) that collaborative rewards from multiple observational human observers can compensate the lack of accuracy obtained from specific subjects.

In Section II the general layout of the cognitive game is described. Section III outlines the processing pipeline used to detect ErrP components. The following Section IV describes the agent learning phase. Results and Conclusions are exposed in Sections V and VI.

II. Materials and Methods

This first part consists in collecting signals from a person's brainwaves while they are watching a game where the agent knows the game rules but not how to win it. Hence, the agent can learn the winning strategy using the person's feedback to improve its own performance. This section describes the process of obtaining the person's brain signals that are used to make the agent more efficient.

The experimental procedure is summarized in Figure 1. The central part is the retrieval of the OHC's brain activity. This process is called brainwave session. Subjects are recruited voluntarily. They are given a consent form with questions regarding their health (previous health issues and particular visual sensitivity), habits (sleeping hours, caffeine and alcohol consumption), as well as an approval petition to collect the required data. The brainwave sessions are performed with 8 subjects, 5 males and 3 females, average age 25.12 years, standard deviation 1.54 years, range of 22-28 years. All subjects have normal vision, are right-handed and have no history of neurological disorders.

After the form is filled out, a short description of the procedure is given to each subject. They are only told that the objective of the agent is to reach the goal and the four movements that the agent can make. When this concludes, the subject is introduced to the wireless digital EEG device (g.Nautilus, g.Tec, Austria) that she/he has to wear during the brainwave session. It has eight electrodes (g.LADYbird, g.Tec, Austria) on the positions Fz, Cz, Pz, Oz, P3, P4, PO7 and PO8, identified according to the 10-20 International System, with a reference set to the right ear lobe and ground set as the AFz position. The electrode contact points are adjusted applying conductive gel until the impedance values displayed by the program g.NeedAccess (g.Tec, Austria) are within the desired range. This process takes between 10 to 15 minutes. After this step, the subject is instructed to close their eyes and the same program is used to check the live channel values so that there are no dead ones and the expected values are displayed for eye movements or muscle chewing.

Once the headset is correctly applied, the OpenVibe Acquisition Server program, from the OpenVibe platform [27], is launched and configured with a sampling rate of 250 Hz. A 50 Hz notch filter is applied to filter out power line noise. An additional bandpass filter between 0.5 Hz and 60 Hz is applied as well. Data is handled and processed with the OpenVibe Designer, from the same platform, using 8 channels for the brain data (one channel per electrode) and an additional channel for the stimulus, which corresponds to a game movement performed by the agent. After everything is connected, the subject seats in a comfortable chair in front of a computer screen. The brightness of the screen is set to the maximum setting to avoid any visual inconvenience in which the subject can not distinguish the components of the game that appear on the screen. This dataset has been published on the IEEE DataPort initiative [28].

The Acquisition Server has the responsibility of receiving and synchronizing the signal data from the headset and any event information from the game, and transfers it to the OpenVibe Designer application. When the subject is ready, the Game Manager and the OpenVibe Designer programs are launched and configured to communicate with the previously mentioned Acquisition Server. A brainwave session consists of several experiences, each one being a game run. At the end, all the sequence of game movements and the signal data of each run are saved for offline processing.

A. Cognitive Game Procedure

The game parsimoniously consists of a $5x5$ grid of grey circular spots with a black background. A blue spot indicates current position of the agent whereas a green spot represents the goal, as shown in Figure 2. The agent's objective is to reach the goal. The circular spot representing the goal remains static at the bottom-right position of the grid, while the one representing the position of the agent always starts at the upper-left position of the grid. When the agent reaches the goal, the position where the agent and the goal are located turns red, showing that the experience has ended. There are four possible movements that the agent can perform: it can go upwards, downwards, towards the left and towards the right, and those movements are bounded to avoid the agent from leaving the grid. The movement direction is selected randomly and is executed once every 2 seconds. At the end, there is a pause of 5 seconds before the next experience starts. Each time an agent moves, the Game Manager program sends an event marker to the Acquisition Server. This is considered as a stimulus to the observational human critic. The experience is designed for it to be evident whenever there is an error (i.e. the agent moves away from the objective) so the subject can notice it immediately after the stimulus is presented, possibly triggering the expected cognitive response, which can be imprinted as an ErrP component within the EEG stream.

III. Signal Processing, Segmentation and Classification

To aid in detecting the ErrP response, an offline processing pipeline and classifier is constructed to identify whether the action taken by the agent is an error or not, from the human observer point of view. It is developed in Python using the "MNE" software platform [29], which is a package designed specifically for processing EEG and Magnetoencephalography

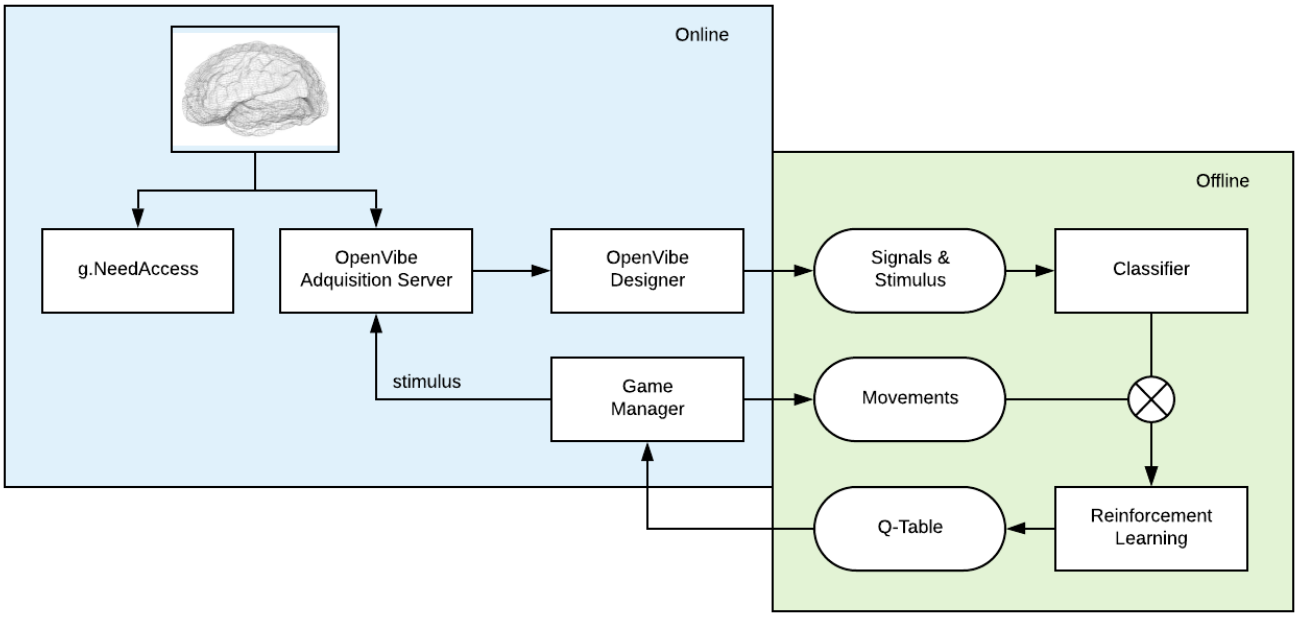

Fig. 1: Overview of the experimental procedure. Brainwaves are obtained by the OpenVibe Acquisition Server. The Game Manager is responsible for generating the game screen, the game mechanics, and the game movements performed by the gaming agent. It is also connected to the Acquisition Server to send stimulus information. The captured information is stored by the OpenVibe Designer. Offline, EEG signals are classified and they are linked to each game movement calculated by the Game Manager to determine proper rewards for each action. This information is used by a Reinforcement Learning algorithm to learn a Q-Table to improve the performance of the agent that plays the game.

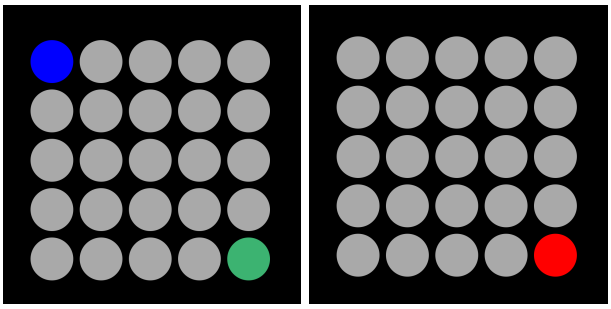

Fig. 2: Grid system representation used in the Cognitive Game Experience. The blue spot represents the initial location while the green spot represents the target location. Once the agent reaches the target spot, its color turns red to indicate the end of the play.

data, and built upon the machine learning library Scikit-Learn [30].

This pipeline consists of the offline processing of the collected signals in order to train a classifier that can decide whether an error potential is triggered. Firstly, the output of a brainwave session is read and an additional band pass filter of 0.1-20.0Hz is applied to the signal. Samples that correspond to the start of an event are tagged using the data from the stimulus channel.

After the data is loaded and tagged, epochs are extracted from the raw data. Epochs consist of all the sample points that take place between the start of the event and 2 seconds later (time for each action to take place), resulting in 500 samples per channel, as the sample frequency is 250 Hz. Thus, each epoch is composed of a matrix 500 x 8 channels.

Samples that do not correspond to an epoch (located beyond the 2 seconds frame after the onset of the event) are not used. Also, epochs referring to the start or finish of the experience are excluded. This is done because the start and the end of the experience doesn't involve the agent taking an action.

In this way, the raw data of a brainwave session is processed into an array of experiences where each element is an array of epochs tagged with a number specifying the prediction of the classifier, i.e. if the epoch corresponds to an action that made the agent moves further from the goal (hit) or an action that made the agent moves closer to the goal (no-hit). The ErrP is expected to be found in hits. To get the data ready for classification, the stimulus channel is removed in order to classify the signals using only the EEG data. Each epoch is regularized using a MinMaxScaler, i.e. substracting the minimun value in the epoch and dividing by the signal peak-to-peak amplitude [31]. The eight channels are concatenated using the MNE Vectorizer function to transform the data matrix into a single array sample. Lastly, this data is used by the classification module as information to train and test a classifier. Four different classification algorithms are used. Logistic Regression, Multi-layer Perceptron with a hidden layer of 100 neurons (i.e. default values for the Scikit-Learn MLPClassifier), Random Forest, KNeighbours

with k=3 and finally a linear kernel Support Vector Classifier (i.e. SVM) [32].

IV. Reinforcement Learning

Each experience consists of a list of game movement configurations and the associated epochs obtained from OHC's brainwaves. The set of experiences of each OHC is split into training and testing. Training experiences are used to train the classifier, whereas test experiences are used to test its performance. After a classifier is trained, the epochs of the experience are classified as hit or no-hit. A reward for each movement in the game run is produced, based on the classification of the epoch that correspond to that movement. The reward can either be -1 when the event is classified as a hit or 0 when it is classified as a no-hit. The accuracy of this rewards depends on the performance of the classifier. The list of game movements and their associated reward information are used to train the agent by a variant of Reinforcement Learning called Q-Learning algorithm.

A. Q-Learning

Q-Learning [33] is a form of model-free reinforcement learning algorithm where an agent tries an action at a particular state and evaluate its consequences in terms of the reward or penalty it receives. In order to represent rewards, a matrix $Q(s,a)$ is used, where rows correspond to all the possible states, whereas columns represent all possible actions. This matrix is known as a Q-Table. The algorithm proceeds by randomly choosing what action to do and updating iteratively the Q-Table based on the received reward r by the following equation

$$Q(s,a) \leftarrow Q(s,a) + \alpha[r + \gamma * \max_{\tilde{a}} Q(\tilde{s},\tilde{a}) - Q(s,a)] \quad (1)$$

where s is the current state, a is the action, α represents the learning rate and γ represents the discount factor, a value between 0 and 1 that determines the importance of long term results versus immediate rewards. Hence, $Q(s,a)$ is the expected value of the sum of discounted rewards that the agent will receive if in the s state, it takes the action a according to this policy. Once the environment has been extensively explored and the Q-Table has been obtained, the action chosen for a given state is the one that maximizes the expected reward according to the Q-Table matrix.

The algorithm is developed in Python and uses the OpenAI Gym toolkit [34]. Gym is a toolkit for developing and comparing reinforcement learning algorithms. It makes no assumptions about the structure of an agent, and is compatible with any numerical computation library, such as TensorFlow or Theano [35].

B. Agent Learning

The Q-Table is initialized with zeros, unless a preexisting Q-Table is passed as a parameter. For the agent, in order to learn from the feedback generated by the OHC, the Q-Table is not used to determine the action to take at a given state. Instead, the policy which determines which action to take in a particular state is given by the agent's actions taken from the brainwave session results. This allows to learn the Q-Table based on the OHC's feedback from the movements the agent took, which are chosen pseudo-randomly, while executing the brainwave session. The previously mentioned feedback is not explicit as it comes from the interpreted brain signal data. This implies that the reward is determined by the OHC's brain activity.

Hence, following the iterative procedure based on Equation 1, the Q-Table is updated in each iteration. After the algorithm finishes iterating through all the training episodes, the Q-Table is stored to test the performance of the agent.

V. Results

Figure 3 show the binary classification accuracy obtained for the eight OHCs for the four different classification algorithms using a 10-fold cross validation procedure. The best overall performance is obtained using Logistic Regression.

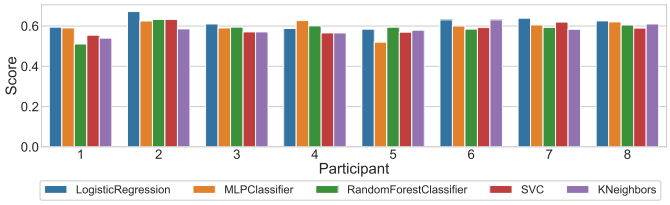

Fig. 3: Binary classification score using different classifiers while recognizing ErrP potentials for the eight OHCs. Chance Level is 0.5.

On the other hand, Figure 4 shows for each OHC the average amount of steps the agent takes to reach the goal as the Q-Table is progressively trained using the reward information obtained from their classified experiences. Each point corresponds to the average number of steps in 200 repetitions that it takes for the agent to reach the goal for a specific Q-Table. The first point represents the amount of steps the agent takes to reach the goal for a Q-Table that hasn't been trained at all, where movements are decided randomly. The next point corresponds to the amount of steps it takes to reach the goal using a policy derived from a Q-Table trained with one experience, and so on.

The results show that as the Q-Table is progressively trained the average amount of steps decreases, meaning that the agent learns. However, the rate at which it learns varies per OHC, depending on the classification accuracy of the extracted brainwaves. For example results for OHC 1 show faster learning than those of OHC 8 (Figure 4).

In the case for OHC 5 and 6, the reward information obtained from the brainwaves is not enough to train the agent effectively. Figures 4 for subjects 5 and 6 show no apparent learning, as the amount of steps to reach the goal doesn't decrease when trained. These results are also consistent with their classification ROC curves, shown on Figures 5 obtained for both OHCs, where the area under the curve are close to chance level. Both OHCs have less recorded data from the sessions in comparison to the rest of the OHCs. This

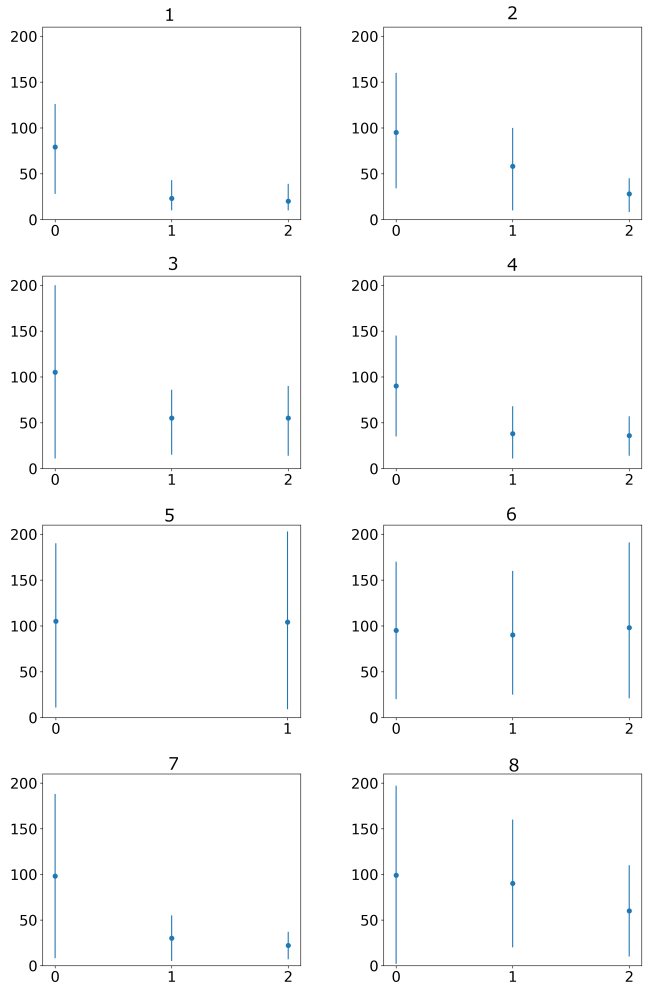

Fig. 4: Average number of steps for the agent to reach the goal when trained with experiences from OHCs 1-8. Y axis show the averaged number of steps, while x axis show the number of experiences used to cumulative train the Q-Table.

variation in performance for different OHCs has been studied extensively in BCI. Besides low data samples there are other reasons affecting the classification accuracy: cognitive reasons, this being the OHC not paying extensively attention to the game dynamics, very low Signal-To-Noise Ratio (SNR) of the ErrP component or even the BCI-illiteracy phenomena where the specific OHC's signals do not contain the expected component response [36].

Figure 6 shows the result of an agent successively trained with experiences obtained from a brainwave session, generated with sham signals. In this case, random EEG signals were generated using OpenVibe Acquisition Server signal generator for all channels, as if they were generated from a human observer who doesn't pay attention to the game. As it can be seen, the agent learns nothing, and regardless of the amount of experiences that are used to learn the Q-Table, the number of steps required to reach the goal does not decrease. This pattern is also obtained when the experiences from OHCs 5 and 6 are used, showing that the reward labeling predicted by the trained classifier for those OHCs worked like a random classifier.

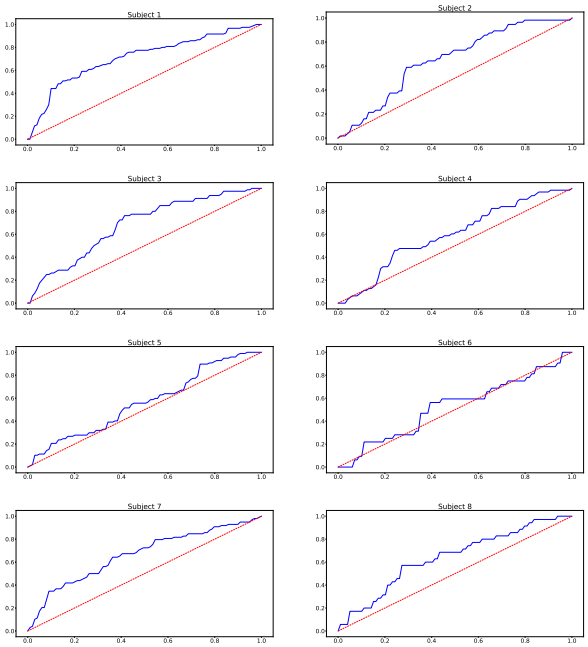

Fig. 5: ROC Curves for OHCs 1-8. True positive rate is on the vertical axis and false positive rate on the horizontal axis.

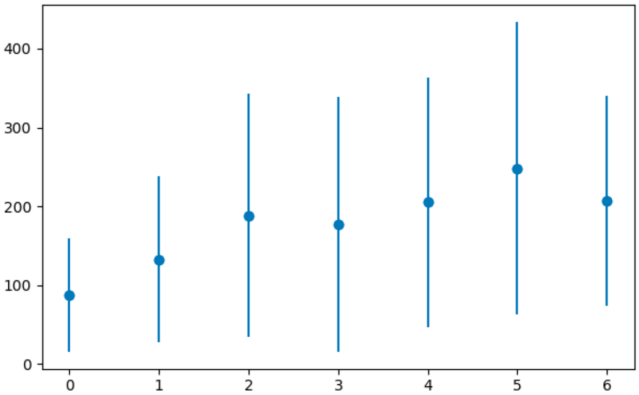

Fig. 6: Average steps using Q-Table trained with a noisy signal.

Electroencephalographic signals have high inter-subject variability [13]. This is evidenced in Figure 7 where the agent training is performed by using rewards obtained by classifying epochs from one OHC, but using a classifier which was trained using the brainwaves from a different OHC. No performance gain is evidenced, the agents learn nothing which implies that the reward information is useless.

Finally, Figure 8 shows the result of training an agent with cumulative experiences from OHCs 1,2,3,4,7, and 8. It can be seen that the overall performance of the agent improves as long as there are more experiences to be used to train it, regardless if they were generated from the brainwaves classification from different OHCs.

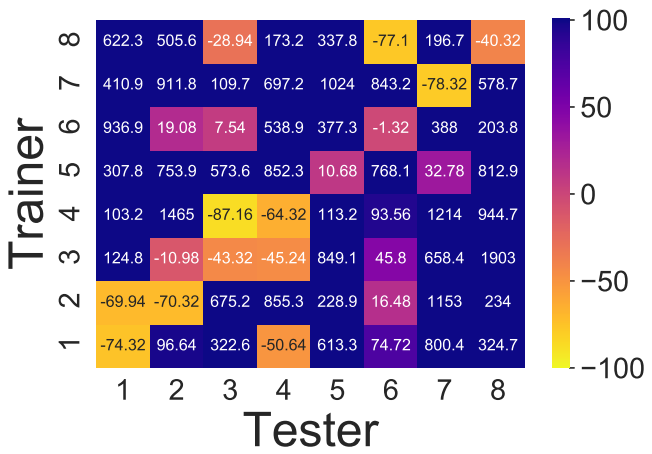

Fig. 7: (A) Average steps using Q-table trained with experiences from OHC 8 classified with a classifier trained with data from OHC 6. (B) Average steps using Q-table trained with experiences from OHC 1 classified with a classifier trained with data from OHC 3.

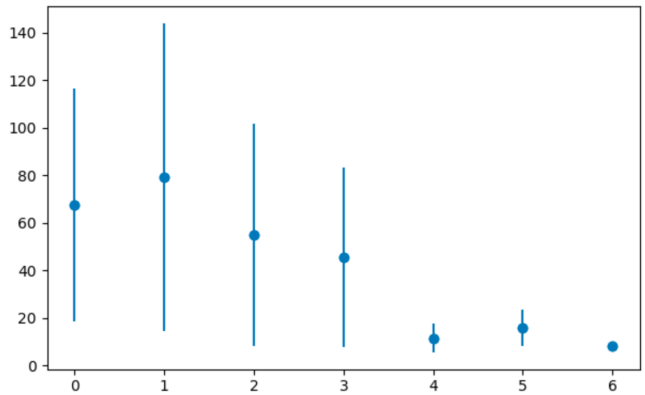

Fig. 8: Average steps using Q-Table trained with 6 experiences from different OHCs.

VI. Conclusion

This work aims to state whether ErrP signals could be used to train a gaming agent using reinforcement learning. The collected data show that ErrP signals can in fact be classified and used to train an agent effectively.

This proposal tries to keep the system as simple as possible, emphasizing information flow from the subjective error perception of the human critic, through the reward generation using the signal processing and classification pipeline, and finally the Q-Table updating to enhance the performance of the gaming agent.

While classifying, the better performing classifier is Logistic Regression. One important aspect of the classification results is the low percentage of false positives (Figure 9), showing a high specificity. It is not common that the agent learns that an action is wrong when in fact it is an action that takes it closer to the goal. On the other hand, the percentage of false negatives is generally higher. However, even though this implies that the agent misses frequently that an action taken is wrong, this is not hindering the overall performance and the agent is still learning.

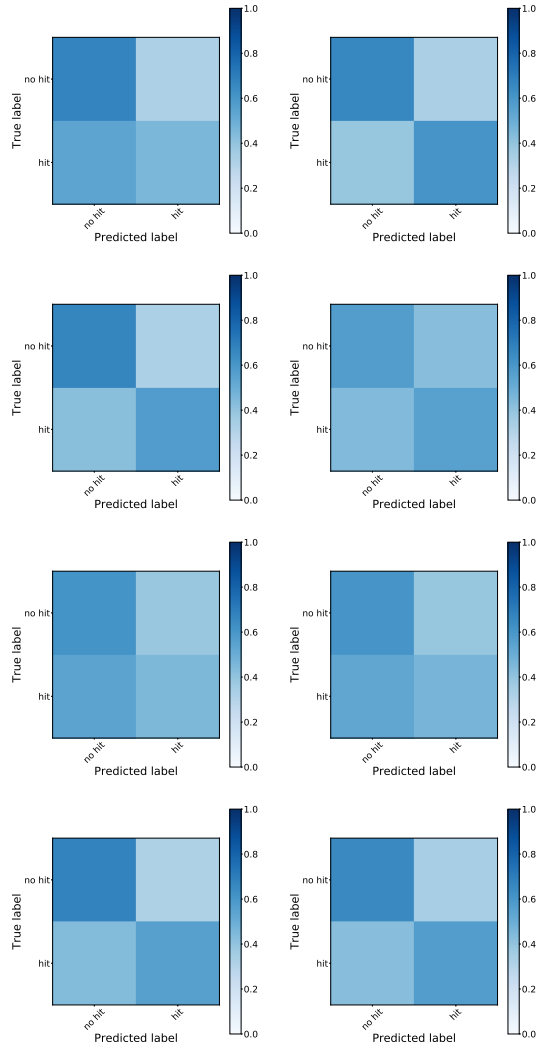

Fig. 9: Confusion Matrix for OHCs 1-8. Darker colors show higher values. It can be seen the lower percentage of false positives (upper right corner of each chart).

Once ErrP signals are identified they can be used to train an agent using a reinforcement learning algorithm. Brainwave sessions have a low amount of experiences in order to reduce fatigue from OHCs. However data suggests that longer sessions are required in order to reach better classification scores, since more data is available in order to train the classifier. It can be seen that OHCs with the largest amounts of data have the best classification. This can also be achieved designing a bigger game system that generates more samples with every session.

At the same time, effective agent training depends on the OHC's data that was used to train it. Results show that training a classifier with data of one OHC, but using it to classify the events of experiences of another OHC does not lead to an improvement on the performance of the agent. Despite that, the rewards generated from different subjects can be used to train the same Q-Table to improve its performance, which may

lead to strategies where the overall performance is improved based on the information from different human critics at the same time.

One additional aspect to remark is the robustness of the learning strategy based on Q-Learning [37], [38]. The obtained accuracy to discriminate ErrPs is low. However, even with such a lower accuracy values, the RL algorithm was able to extract meaningful information from rewards that were helpful to improve the agent performance.

Further work will be conducted in order to increase the complexity of the game to allow the possibility that the target position be dynamically changed. In that case the agent would start to learn to follow the target, instead of learning to go to a specific static destination spot. Additionally, the classifier could be enhanced to recognize more effectively the Error Potential [39].

Concluding, this research shows evidence that brain signals can be used as an interface between human and a gaming computer enabling an alternative communication with the system without explicit input from the user.

Acknowledgment

The authors would like to thank the Laboratory Centro de Inteligencia Computacional and to ITBA University.

Funding

This work was supported by the grant ITBACyT-15 issued by ITBA University.

References

[1] D. Xu, M. Agarwal, F. Fekri, and R. Sivakumar, "Playing Games with Implicit Human Feedback," in *AAAI-20 Workshop on Reinforcement Learning in Games*, New York, New York, USA, 2020. [Online]. Available: https://www.semanticscholar.org/paper/Playing-Games-with-Implicit-Human-Feedback-Xu-Agarwal/faba141483180e53f461c6035ce95041cfed9a8f

[2] T. O. Zander, L. R. Krol, N. P. Birbaumer, and K. Gramann, "Neuroadaptive technology enables implicit cursor control based on medial prefrontal cortex activity," *Proceedings of the National Academy of Sciences*, vol. 113, no. 52, pp. 14 898–14 903, dec 2016. [Online]. Available: http://www.pnas.org/lookup/doi/10.1073/pnas.1605155114

[3] M. Carter, J. Downs, B. Nansen, M. Harrop, and M. Gibbs, "Paradigms of games research in HCI: A review of 10 years of research at CHI," in *CHI PLAY 2014 - Proceedings of the 2014 Annual Symposium on Computer-Human Interaction in Play*. New York, New York, USA: ACM Press, 2014, pp. 27–36. [Online]. Available: http://dl.acm.org/citation.cfm?doid=2658537.2658708

[4] P. Barr, J. Noble, and R. Biddle, "Video game values: Human-computer interaction and games," *Interacting with Computers*, vol. 19, no. 2, pp. 180–195, mar 2007. [Online]. Available: https://academic.oup.com/iwc/article-lookup/doi/10.1016/j.intcom.2006.08.008

[5] Y. Zhao, I. Borovikov, F. De Mesentier Silva, A. Beirami, J. Rupert, C. Somers, J. Harder, J. Kolen, J. Pinto, R. Pourabolghasem, J. Pestrak, H. Chaput, M. Sardari, L. Lin, S. Narravula, N. Aghdaie, and K. Zaman, "Winning Is Not Everything: Enhancing Game Development with Intelligent Agents," *IEEE Transactions on Games*, vol. 12, no. 2, pp. 199–212, jun 2020. [Online]. Available: https://ieeexplore.ieee.org/document/9104019/

[6] G. A. M. Vasiljevic and L. C. de Miranda, "Brain–Computer Interface Games Based on Consumer-Grade EEG Devices: A Systematic Literature Review," *International Journal of Human-Computer Interaction*, vol. 36, no. 2, pp. 105–142, jan 2020. [Online]. Available: https://www.tandfonline.com/doi/full/10.1080/10447318.2019.1612213

[7] R. Scherer, M. Pröll, B. Allison, and G. R. Müller-Putz, "New input modalities for modern game design and virtual embodiment," in *Proceedings - IEEE Virtual Reality*. IEEE, mar 2012, pp. 163–164. [Online]. Available: http://ieeexplore.ieee.org/document/6180932/

[8] A. Nijholt and D. Tan, "Playing with your brain: Brain-computer interfaces and games," in *ACM International Conference Proceeding Series*, vol. 203. New York, New York, USA: ACM Press, 2007, pp. 305–306. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1255047.1255140

[9] J. Aguado-Delgado, J. M. Gutiérrez-Martínez, J. R. Hilera, L. De-Marcos, and S. Otón, "Accessibility in video games: a systematic review," *Universal Access in the Information Society*, vol. 19, no. 1, pp. 169–193, mar 2020. [Online]. Available: http://link.springer.com/10.1007/s10209-018-0628-2

[10] L. Yakovlev, N. Syrov, G. Nikolai, and A. Kaplan, "BCI-Controlled Motor Imagery Training Can Improve Performance in e-Sports," in *International Conference on Human-Computer Interaction HCII 2020*, A. M. Stephanidis C., Ed., vol. 1. Springer, Cham, jul 2020, pp. 581–586. [Online]. Available: http://link.springer.com/10.1007/978-3-030-50726-8{_}76

[11] C. B. Holroyd, O. E. Krigolson, R. Baker, S. Lee, and J. Gibson, "When is an error not a prediction error? An electrophysiological investigation," *Cognitive, Affective and Behavioral Neuroscience*, vol. 9, no. 1, pp. 59–70, mar 2009. [Online]. Available: http://www.springerlink.com/index/10.3758/CABN.9.1.59

[12] G. Dornhege, *Toward brain-computer interfacing*. MIT Press, 2007. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/6281216

[13] R. Chavarriaga, A. Sobolewski, and J. d. R. Millán, "Errare machinale est: The use of error-related potentials in brain-machine interfaces," *Frontiers in Neuroscience*, vol. 8, no. 8 JUL, p. 208, jul 2014. [Online]. Available: http://journal.frontiersin.org/article/10.3389/fnins.2014.00208/abstract

[14] R. S. Sutton and A. G. Barto, *Reinforcement Learning, Second Edition: An Introduction*. MIT press, 2018.

[15] J. M. Santos and C. Touzet, "Exploration tuned reinforcement function," *Neurocomputing*, vol. 28, no. 1-3, pp. 93–105, oct 1999. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231298001179

[16] T. T. Nguyen, N. D. Nguyen, and S. Nahavandi, "Deep Reinforcement Learning for Multiagent Systems: A Review of Challenges, Solutions, and Applications," *IEEE Transactions on Cybernetics*, pp. 1–14, 2020. [Online]. Available: https://ieeexplore.ieee.org/document/9043893/

[17] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, Y. Chen, T. Lillicrap, F. Hui, L. Sifre, G. Van Den Driessche, T. Graepel, and D. Hassabis, "Mastering the game of Go without human knowledge," *Nature*, vol. 550, no. 7676, pp. 354–359, oct 2017. [Online]. Available: http://www.nature.com/articles/nature24270

[18] I. Iturrate, L. Montesano, and J. Mínguez, "Robot reinforcement learning using EEG-based reward signals," in *Proceedings - IEEE International Conference on Robotics and Automation*. IEEE, may 2010, pp. 4822–4829. [Online]. Available: http://ieeexplore.ieee.org/document/5509734/

[19] S. K. Kim, E. A. Kirchner, A. Stefes, and F. Kirchner, "Intrinsic interactive reinforcement learning-Using error-related potentials for real world human-robot interaction," *Scientific Reports*, vol. 7, no. 1, p. 17562, dec 2017. [Online]. Available: http://www.nature.com/articles/s41598-017-17682-7

[20] J. Omedes, I. Iturrate, L. Montesano, and J. Minguez, "Using frequency-domain features for the generalization of EEG error-related potentials among different tasks," in *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*. IEEE, jul 2013, pp. 5263–5266. [Online]. Available: http://ieeexplore.ieee.org/document/6610736/

[21] T. J. Luo, Y. C. Fan, J. T. Lv, and C. L. Zhou, "Deep reinforcement learning from error-related potentials via an EEG-based brain-computer interface," in *Proceedings - 2018 IEEE International Conference on Bioinformatics and Biomedicine, BIBM 2018*. IEEE, dec 2019, pp. 697–701. [Online]. Available: https://ieeexplore.ieee.org/document/8621183/

[22] S. K. Goh, N. P. Tran, D. T. Pham, S. Alam, K. Izzetoglu, and V. Duong, "Construction of Air Traffic Controller's Decision Network Using Error-Related Potential," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Springer, Cham, jul 2019, vol. 11580 LNAI, pp. 384–393. [Online]. Available: http://link.springer.com/10.1007/978-3-030-22419-6{_}27

[23] C. Wirth, P. M. Dockree, S. Harty, E. Lacey, and M. Arvaneh, "Towards error categorisation in BCI: Single-trial EEG classification between different errors," *Journal of Neural Engineering*, vol. 17, no. 1, p. 016008, dec 2020. [Online]. Available: https://iopscience.iop.org/article/10.1088/1741-2552/ab53fe

[24] L. Schiatti, J. Tessadori, N. Deshpande, G. Barresi, L. C. King, and L. S. Mattos, "Human in the Loop of Robot Learning: EEG-Based Reward Signal for Target Identification and Reaching Task," in *Proceedings - IEEE International Conference on Robotics and Automation*. IEEE, may 2018, pp. 4473–4480. [Online]. Available: https://ieeexplore.ieee.org/document/8460551/

[25] D. Plass-Oude Bos, B. Reuderink, B. van de Laar, H. Gürkök, C. Mühl, M. Poel, A. Nijholt, and D. Heylen, "Brain-Computer Interfacing and Games," in *Brain-Computer Interfaces, Applying our minds to Human Computer Interaction*. Springer, London, 2010, pp. 149–178. [Online]. Available: http://link.springer.com/10.1007/978-1-84996-272-8{_}10

[26] S. E. Kober, M. Ninaus, E. V. Friedrich, and R. Scherer, "Bci and games: Playful, experience-oriented learning by vivid feedback?" in *Brain–Computer Interfaces Handbook*. CRC Press, 2018, pp. 209–234.

[27] Y. Renard, F. Lotte, G. Gibert, M. Congedo, E. Maby, V. Delannoy, O. Bertrand, and A. Lécuyer, "OpenViBE: An Open-Source Software Platform to Design, Test, and Use Brain–Computer Interfaces in Real and Virtual Environments," *Presence: Teleoperators and Virtual Environments*, vol. 19, no. 1, pp. 35–53, feb 2010. [Online]. Available: http://www.mitpressjournals.org/doi/10.1162/pres.19.1.35

[28] F. B. J. M. N. N. J. Vitali, "Errp-dataset," 2019. [Online]. Available: http://dx.doi.org/10.21227/6emh-wb46

[29] A. Gramfort, M. Luessi, E. Larson, D. A. Engemann, D. Strohmeier, C. Brodbeck, R. Goj, M. Jas, T. Brooks, L. Parkkonen, and M. Hämäläinen, "MEG and EEG data analysis with MNE-Python," *Frontiers in Neuroscience*, vol. 7, no. 7 DEC, p. 267, dec 2013. [Online]. Available: http://journal.frontiersin.org/article/10.3389/fnins.2013.00267/abstract

[30] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[31] T. Zhou and J. P. Wachs, "Spiking Neural Networks for early prediction in human–robot collaboration," *International Journal of Robotics Research*, vol. 38, no. 14, pp. 1619–1643, dec 2019. [Online]. Available: http://journals.sagepub.com/doi/10.1177/0278364919872252

[32] F. Lotte, L. Bougrain, A. Cichocki, M. Clerc, M. Congedo, A. Rakotomamonjy, and F. Yger, "A review of classification algorithms for EEG-based brain-computer interfaces: A 10 year update," *Journal of Neural Engineering*, vol. 15, no. 3, p. 031005, jun 2018. [Online]. Available: http://stacks.iop.org/1741-2552/15/i=3/a=031005?key=crossref.9cd2b15ab65c8ad34b475584b43dc509

[33] C. J. C. H. Watkins and P. Dayan, "Q-learning," *Machine Learning*, vol. 8, no. 3-4, pp. 279–292, may 1992. [Online]. Available: http://link.springer.com/10.1007/BF00992698

[34] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba, "Openai gym," 2016.

[35] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: http://tensorflow.org/

[36] R. Yousefi, A. Rezazadeh Sereshkeh, and T. Chau, "Development of a robust asynchronous brain-switch using ErrP-based error correction," *Journal of neural engineering*, vol. 16, no. 6, p. 066042, nov 2019. [Online]. Available: https://iopscience.iop.org/article/10.1088/1741-2552/ab4943

[37] R. Bauer and A. Gharabaghi, "Reinforcement learning for adaptive threshold control of restorative brain-computer interfaces: A Bayesian simulation," *Frontiers in Neuroscience*, vol. 9, no. FEB, p. 36, feb 2015. [Online]. Available: http://journal.frontiersin.org/Article/10.3389/fnins.2015.00036/abstract

[38] J. Rubin, O. Shamir, and N. Tishby, "Trading value and information in MDPs," in *Intelligent Systems Reference Library*. Springer, Berlin, Heidelberg, 2012, vol. 28, pp. 57–74. [Online]. Available: http://link.springer.com/10.1007/978-3-642-24647-0{_}3

[39] F. Iwane, R. Chavarriaga, I. Iturrate, and J. Del Millan, "Spatial filters yield stable features for error-related potentials across conditions," in *2016 IEEE International Conference on Systems, Man, and Cybernetics, SMC 2016 - Conference Proceedings*. IEEE, oct 2017, pp. 661–666. [Online]. Available: http://ieeexplore.ieee.org/document/7844316/