Human-In-The-Loop RL with an EEG Wearable Headset: On *Effective* Use of Brainwaves to Accelerate Learning

Mohit Agarwal
me.agmohit@gatech.edu
Georgia Institute of Technology

Shyam Krishnan Venkateswaran
shyam1@gatech.edu
Georgia Institute of Technology

Raghupathy Sivakumar
siva@ece.gatech.edu
Georgia Institute of Technology

ABSTRACT

Intrinsic *Human-In-The-Loop Reinforcement Learning (HITL-RL)* is an approach to obtain the human feedback implicitly by capturing brain waves through the use of *wearable electroencephalogram (EEG) headsets*. It can significantly accelerate the training convergence of RL algorithms while reducing the burden placed on the humans involved in the training loop. While a human naturally observes the performance of an RL agent, any erroneous behavior of the agent can be recognized through the error-potentials[1] (ErrP) in the EEG signal. This information can then be incorporated into the reward function of the RL algorithm to accelerate its learning. The detection accuracy of the error-potentials thus significantly affects the convergence time of the RL algorithm. The focus of this work is the reliable detection of error-potentials using the brain waves of the user detected using only an off-the-shelf EEG wearable. We first present a new error-potential decoding algorithm that leverages the spatial, temporal, and frequency-domain characteristics of the EEG signals. We develop three Atari-like game environments and recruit 25 volunteers for evaluation. The proposed algorithm achieves an accuracy performance of 73.71% (an improvement of 8.11% over the current state-of-the-art algorithm). We then show that a modified algorithm that intelligently discards low-confidence estimates is capable of boosting the accuracy to 79.51% (16.63% improvement).

ACM Reference Format:
Mohit Agarwal, Shyam Krishnan Venkateswaran, and Raghupathy Sivakumar. 2020. Human-In-The-Loop RL with an EEG Wearable Headset: On *Effective* Use of Brainwaves to Accelerate Learning. In *The 6th ACM Workshop on Wearable Systems and Applications (WearSys'20), June 19, 2020, Toronto, ON, Canada.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3396870.3400014

1 INTRODUCTION

Reinforcement Learning (RL) algorithms have become an integral part of end-user applications, including autonomous systems (e.g., recommendation engines, self-driving cars, etc.), and robotics where the primary purpose of such systems is to understand and act in unseen environments. However, training an RL algorithm for a real-world task is challenging due to the high-dimensional state-space, sparsity of reward functions, and the inherent requirement of a large number of training samples. In this context, HITL-RL (Human-In-The-Loop Reinforcement Learning) is a practical approach to significantly accelerate the learning and the convergence rate of RL algorithms. Methods like inverse RL (or learning through demonstrations), explicit human feedback (through labels, ratings, etc.) could reduce the search space or supplement the rewards, making the algorithm train more efficiently [1, 2]. Such approaches despite being effective, raise a conflict between the need to increase the richness of the reward function and minimizing the burden placed on the human to generate such rewards.

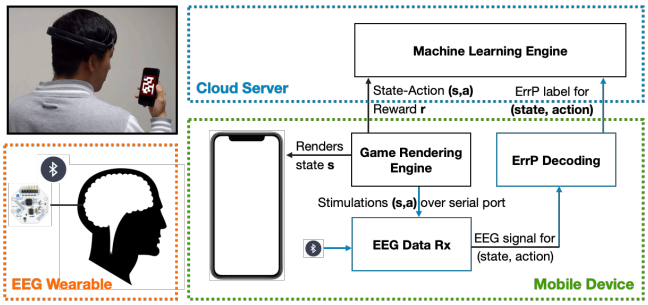

Figure 1: End-to-end system architecture design for the mobile HITL-RL, along with an illustration of the use case.

This has inspired the paradigm of intrinsic HITL-RL, where the human feedback is obtained intrinsically by capturing their brainwaves through the use of wearable electroencephalogram (EEG) headsets. While the human is silently observing the RL agent performing an incorrect (or suboptimal) action, the error-processing system inside the human brain elicits a natural reaction as a biological response for recognizing and possibly correcting the error [3–5]. This natural reaction is manifested in the form of error-potentials (also known as ErrPs) in the captured EEG through the wearable EEG headset. This approach allows us to obtain human feedback without requiring any explicit actions, thus significantly reducing the burden placed on human subjects. The recognized erroneous behavior of the agent is fed to the RL algorithm's reward function to improve the performance and the convergence rate. An illustration is shown in Fig. 1 (top-left), where a human is wearing an EEG headset, silently observing (and assessing) a computer agent interacting with a computer game environment.

Since EEG signals represent a myriad of brain activity and thus are inherently noisy, the estimation of error-potentials is not fully reliable. Inaccurate detection of the error-potential (both false positives and false negatives) could misguide the RL agent, and negatively impact the convergence time of the RL algorithm. An accuracy rate of as low as 60% could make the HITL-RL paradigm completely ineffective (explained in section 3.2).

[1]ErrP and error-potential are used interchangeably in this work

WearSys'20, June 19, 2020, Toronto, ON, Canada

ACM ISBN 978-1-4503-8013-3/20/06…$15.00
https://doi.org/10.1145/3396870.3400014

The focus of this work is the reliable detection of intrinsic reactions (specifically, error-potentials) using the brain waves of the user detected using only an off-the-shelf EEG wearable. The state-of-the-art algorithm[2] for error-potential detection [6, 7] leverages the spatial distribution of the EEG signal power and performs with an average accuracy of 68.17% (more details in section 4.2). We propose a new algorithm to accurately detect error-potentials leveraging the spatial, temporal, and frequency-domain characteristics of the observed brain potential and demonstrate the gain in accuracy, among other metrics. In addition to this, our proposed algorithm is also capable of trading-off accuracy with sample efficiency, discarding the low-confidence estimations, and hence boosting the accuracy rate. We design three Atari-like game environment and collect the dataset of a total of 25 human volunteers to evaluate the performance of the proposed algorithm, and compare with the state-of-the-art algorithm for error-potential detection [6]. Our proposed algorithm performs with an average accuracy of 79.51% (a 16.63%) improvement over the baseline algorithm, achieving a training convergence with a 1.8x acceleration rate. We have made the source code for the implementation publicly available[3].

The rest of the paper is organized as follows. In section 2, we first provide some background information on EEG and error-potentials. We then present the use case and the end-to-end system architecture for a mobile HITL-RL. In section 3, we describe our system setup and data collection process, alongside outlining the motivation of our paper. We also present the baseline ErrP detection algorithm [6]. In section 4, we describe the proposed algorithm, and evaluate and compare the performances over the collected dataset. Finally, in section 5, we conclude the paper.

2 A CASE FOR HITL-RL IN MOBILE SYSTEMS

2.1 EEG and Error-related Potentials

EEG is the recording of the electrical activity of the brain using electrodes that are placed on a user's scalp, first recorded by Hans Berger in 1929. This electrical activity is the result of synchronized electrical firings of billions of neurons inside the brain responsible for the processing and communication of massive amounts of information. The raw analog electric potentials are tapped by placing electrodes (conductive disks, often mounted in a fabric cap) over the human scalp. The raw signals are further digitized and amplified through appropriate sensing hardware. The measurement and processing of such potentials provide a window into a myriad of activities inside the brain, including emotions, perception, attention, engagement, etc [8–10].

Wearable solutions for EEG recording: The advancements in the hardware and sensor design have made consumer-grade wearable EEG headsets commercially relevant. With EEG headsets like Neurosky Mindwave, Emotiv EPOC+, Muse, OpenBCI, EEG signals can be reliably tapped into by the user wearing the headset [11], digitized and communicated over a wireless link to a mobile device.

Error-related Potentials (ErrP): ErrP is a negative potential that is detected through EEG when the subject perceives or recognizes an error during a task [12] (Fig. 2(right)). According to [13],

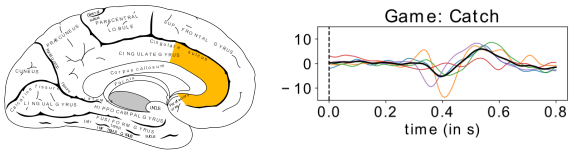

Figure 2: The left figure shows the Anterior Cingulate Cortex (ACC), the point of origin of the error-potential. The right figure shows the error-potentials over time-domain captured through a wearable EEG headset. The colored lines represent the recordings from different subjects, and the solid black line represents the average over all the subjects.

the elicited ErrP is maximally negative at around 50 ms after the occurrence of the perceived error. The origin of ErrP has also been mapped to the Anterior Cingulate Cortex (ACC) in the brain [14] (Fig. 2(left)). ErrPs have been studied in the context of Brain-Computer Interfaces (BCI) as a mechanism for *feedback* when the user perceives an error [15].

2.2 Use case and System Architecture of HITL-RL in mobile systems

The paradigm of HITL for learning systems enables the training of artificial agents to adapt to novel situations. Computer games are used as a proxy for the complex environment since they are the fertile ground for the definition, understanding, and improvement of learning algorithms in a low overhead and speedy fashion. The paradigm of HITL leverages information-rich human knowledge as feedback for reward shaping in RL algorithms. The human feedback can be obtained by explicit questionnaires (e.g., ratings, reviews, labeling, etc.) or extracted implicitly from user behavior or reactions (for instance, error-potentials). The proliferation of consumer-grade EEG headsets has made it increasingly easier to capture such implicit feedback. Most of the commercial EEG headsets connect to smartphones seamlessly via Bluetooth Low Energy (BLE) and operate in a completely wireless manner both for power and communication (e.g., Muse, EPOC+) while being comfortable and aesthetically appealing to the user.

Since games can be deployed on mobile systems, and the EEG can be captured through wearables, this presents an interesting and ubiquitous use case to continuously collect EEG-based feedback through wearables and smartphones and to augment the RL algorithms. Current smartphones are equipped with powerful processors (and even GPUs), capable of performing on-device error-potential detection. Reliable and persistent connectivity of mobile devices to the Internet can be used for sending such labels to a central repository server, where the learning algorithms can be improved. An illustrative use case is the use of EEG data from a user observing an online game on a smartphone (similar to a user watching an advertisement on her mobile device). In game settings similar to the environments described in this paper, the game states can be attributed and synchronized with captured EEG signals. With proper detection of ErrP, a learning algorithm can utilize it as a reward function to learn the optimal strategy for that game.

End-to-end System Architecture The mobile HITL-RL system envisioned consists of three main components (as shown in Fig. 1): (i) *wearable device*, (ii) *mobile device*, and (iii) *cloud server*. The

[2] we also refer to the state-of-the-art algorithm as baseline algorithm
[3] https://github.com/meagmohit/errp_decode

[4] NOOP (No Operation) - the agent does not take any action at a particular time-step

Game	Environment	Goal	Action space	Start/restart sequence
Maze	10x10 grid with agent and target	2D navigation to a fixed target	←, ↓, ↑, →	Maze is fixed for all instances.
Catch	10x10 grid with egg and basket. Egg falls one grid each timestep.	1D navigation by the basket to catch the egg at the right time.	NOOP[4], ←, →	Egg starts at a random horizontal position from the top.
Wobble	1x20 grid with cursor and target	1D navigation to reach the target	←, →	Agent spawns at center of screen and target within 3 blocks of agent.

Table 1: Description of the game environments

wearable device captures and timestamps the EEG signals, and ships the digitized signals to the user's mobile device through a wireless link. The *mobile device* renders the game and collects the EEG data from the wearable device. The collected timestamped EEG data is properly attributed and synchronized with the mobile game state. The EEG signals are processed on the mobile device using the proposed error-potential decoding algorithm. The decoded EEG labels are sent to a *cloud server* which executes appropriate RL algorithms (e.g., Q-learning, Deep RL, etc.) incorporating the EEG-based human feedback.

2.3 Related work

Error-potentials in EEG signals are studied under two paradigms in human-machine interaction tasks, (i) *feedback and response ErrPs:* error made by humans [16], (ii) *interaction ErrPs:* errors made by machines in interpreting human intent [17]. There are several works that propose the use of ErrP from a passive (or silent) human observer as feedback to a learning system. In [3], a simple robotic system that performs a binary selection task using ErrP as feedback is studied both in open and closed-loop settings. This enables ErrPs to be used as a supplementary reward for the Q-learning[4] or deep Reinforcement Learning (RL) algorithm[5]. The use of error-potentials in human-computer interaction tasks, or for the acceleration of RL algorithms is underpinned upon the accurate detection of the error-potentials. Several approaches have been proposed in the literature to decode the error-potentials. [18] demonstrated the possibility of continuous and asynchronous detection of ErrP, while [17] proposed a statistical classifier. The state-of-the-art error-potential decoding algorithm relies on Riemannian geometry framework and was proposed by Baranchant et al [6]. It was later successfully applied for various classification paradigms in BCIs, namely, motor imagery, P300, SSVEP, etc. We provide an explanation of the above algorithm in section 8, and compare it with the proposed algorithm in section 4.2.

3 THE PROBLEM AND BASELINE ALGORITHM

3.1 System setup and data collection

We consider a setup where a human is wearing an EEG headset, silently observing (and assessing) a computer agent interacting with a computer game environment. The human's intrinsic reactions to the agent's behavior are sensed by a wearable EEG headset and monitored through the error-potentials. We use OpenBCI Cyton[5] platform along with BIOPAC CAP100-C (16-channels) as the wearable headset to collect and timestamp the EEG data. The game environments are designed in OpenAI Gym[19], where the information regarding the current state of the game and the actions taken by the agent is transmitted over the TCP port. OpenViBE[20] continuously listens to the TCP port, and synchronizes with the EEG data according to the timestamps. We have developed three Atari-like game environments, namely, Wobble, Catch, and Maze (Fig. 3 (left)), explained in Table 1.

Data Collection: Subjects were asked to sit comfortably in front of a computer screen and to wear the EEG headset. We used electrode gel to establish surface contact between electrodes and the scalp. The electrodes used were Fp1, Fp2, Fpz, F3, F4, F7, F8, Fz, C3, C4, Cz, P3, P4, Pz, O1, and O2 (as per the 10-20 electrode system). After setup, an OpenBCI GUI software was used to verify the signal quality manually. The duration of each experiment was limited to less than 15 minutes per session with the agent taking actions once every 1500 ms. A total of 25 subjects were recruited for the data collection, with 12 subjects for the Maze game, 7 subjects for the Catch game, and 6 subjects for the Wobble game. All the research protocols for the user data collection were reviewed and approved by the Georgia Tech Institutional Review Board.

3.2 Motivation and Problem Statement

A simple yet effective strategy to incorporate human feedback in learning algorithms is *reward shaping* [21], where the human feedback is added to the reward function to guide the learning agent. In our case, human feedback is obtained in the form of error-potentials, on the actions taken by the machine agent while interacting with a game environment. This approach has been previously applied in the context of error-potentials based rewards [5]. If an error-potential is detected, a negative penalty is added to the reward function, to prevent such sub-optimal actions in the future. In [5], it has been shown that this approach can significantly accelerate the training convergence of the RL agent. We obtained the code from the authors and performed a sensitivity analysis of the acceleration performance for Maze game as per the classification accuracy of error-potentials. We present the performance degradation in acceleration in Fig. 3 (right). The current state-of-the-art error-potential detection algorithm [6] performs with an average accuracy of 68%, requiring $\tilde{1}40$ episodes[6] to master the Maze game, accounting for 1.22x acceleration. However, for an average detection accuracy of 80% the training convergence could boost up to 1.8x acceleration.

Problem Statement: Our goal in this paper is to improve the accuracy of the error-potential decoding algorithm, enabling higher convergence rates for the RL algorithms. Despite the attractive performance rate of [6], there is a significant room for improvement. We define the formal problem statement as follows, for a given labeled training data (X_{train}, y_{train}), and a raw EEG sample, X_{test}, the problem statement is to label the X_{test}, as "ErrP" or "non-ErrP"

[5] https://www.openbci.com

[6] episodes are defined as the full gameplay until the player wins the game

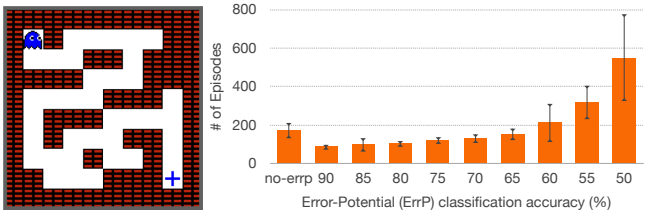

Figure 3: The left figure shows the screenshot of the Maze game, and the right figure shows the RL acceleration as per the accuracy of error-potential detection

with higher accuracy and high confidence in comparison with the state-of-the-art algorithm.

3.3 Baseline (State-of-the-art) algorithm

Algorithm 1: Riemannian Geometry based ErrP classification algorithm [6]

Input : raw EEG signals (X)

$X_f \leftarrow$ filtering(X, $freq\_band$, $filter\_order$) ;
$X_C \leftarrow$ covariance(X_f) ;
$X_D \leftarrow$ electrode_select (X_C, $nelec$) ;
$X_T \leftarrow$ tangent_space(X_D) ;
$X_N \leftarrow$ normalization(X_T, norm="l1") ;
$score \leftarrow$ elasticnet (X_N, λ_1, λ_2) ;
if $score > score_{th}$ **then return** True ;
else return False. ;

The algorithm parameters are explained in section 8

The principal idea in this approach is underpinned on the assumption that spatial distribution and power of the signal remain unaltered for a specific mental activity, which can be captured using the covariance matrix. Since the space of the covariance matrices is a subspace of Symmetric Positive Definite (SPD) matrices, it forms a differentiable Riemannian manifold. In this manifold, (i) the tangent space has an inner product that varies smoothly, and (ii) the distance between two points can be computed using Riemannian distance (or *geodesic*, δ_R) defined as,

$$\delta_R(C_1, C_2) = ||log(C_1^{-1}C_2)||_F = \left[\sum_{i=1}^{n} log^2(\lambda_i)\right]^{\frac{1}{2}} \quad (1)$$

Here, C_1 and C_2 represent the covariance matrices (corresponding to different data trials). $||.||_F$ represents Frobenius norm, and λ_i represents the i^{th} eigenvalue of $C_1^{-1}C_2$. One of the unique properties of this space is, $\delta_R(W^TC_1W, W^TC_2W) = \delta_R(C_1, C_2)$, for all invertible SPD W, implying that this space is invariant by projection (and hence less prone to noise and imperfect cap placements). The full algorithm is presented in Algorithm 1 and is explained below.

Algorithm Description: (Step 1) Firstly, the raw EEG data is bandpass filtered in a frequency range ($freq\_band$) of [0.5, 40] Hz, and epochs of 800ms duration were extracted relative to the pre-stimulus 200ms baseline. The epochs were then spatially filtered with "xDAWN Spatial Filter" [22–24]) to improve the signal to signal plus noise ratio (SSNR), where $filter\_order$ corresponds to the Xdawn components used to decompose the data for each event type. (Step 2) A covariance matrix is computed accounting for the spatial distribution of the signal power. (Step 3) As the raw input signal is high-dimensional, the spatially filtered signals are reduced to fewer relevant channels ($nelec$) using a backward elimination principle based on the Riemannian distance between spatial covariance matrices as the selection criterion [25]. (Step 4) The reduced covariance matrix is projected into the tangent space, allowing to manipulate features in the Euclidean space [6, 7]. (Step 5, 6) Finally, the features in the tangent space (X_T) are normalized using the L1 norm, and subjected to a linear regression model with L1 (λ_1) and L2 (λ_2) penalties. If the output of linear regression crosses the preset threshold ($score\_th$), the signal is labeled as an ErrP. $score\_th$ is set offline through maximizing accuracy over training samples.

4 EFFECTIVE DECODING OF BRAINWAVES

4.1 Proposed Algorithm

The baseline algorithm relies only on the spatial distribution of the scalp potentials (through the estimation of the covariance matrix) to classify the error-potentials. In practical situations, the error-potentials are not exactly time-locked, and manifest phase jitters due to the shift in user focus, synchronization issues, etc, resulting in reduced classification performance. Further, the distribution of power across time- and frequency-spectrum is known to provide additional information regarding the associated mental activity. We supplement the spatial- domain features along with the time- and frequency- domain features and we efficiently combine the information across these three dimensions based on a soft-voting based ensemble approach (presented in Algorithm 2).

Algorithm Description: The pre-processing steps (Step 1) and spatial filtering steps (Step 2-5) are similar to Algorithm 1. (Step 6) In spatial filtering, we use a squared hinge loss along with L1 and L2 penalties, and obtain the calibrated confidence scores (p_s) for spatial-domain based prediction based on [26].

Frequency-domain features: (Step 7) A multi-taper spectral estimation method [27] within 400ms to 1000ms time window ($time_f$) after stimulus onset is used to compute the power densities in 1-15 Hz frequency interval ($freq_f$). (Step 8) The obtained power spectral values are converted to a logarithmic scale (dB). (Step 9) A linear-kernel based Support Vector Machine (SVM) with a small-margin hyperplane is used to classify the frequency-based features, and the confidence scores (p_f) are estimated using Platt scaling [28].

Time-domain features: (Step 8) The spatially filtered signals are divided into multiple buckets ($bucket\_size$) of 50ms each. (Step 9) We compute the average amplitude of each bucket as the raw features representing time-domain variations in error-potentials. (Step 9-10) The mean amplitude based features are normalized using L2 norm, before feeding them to the linear SVM. Similar to the frequency-domain pipeline, we compute the probability estimations (p_t) representing the prediction confidence.

Ensemble classification: We use a soft voting based ensemble classification to predict the "ErrP" or "non-ErrP" class. In this method, we average the classification probability i.e. p_t, p_f and p_s to compute the final estimation probability, p. To improve the overall detection performance of the system, we discard the low-confidence predictions. We define a parameter, probability threshold (p_{th}), to identify the low-confidence predictions. If the ensemble classifier

prediction probability (i.e., p) lies between $[1-p_{th}, p_{th}]$, we discard the corresponding samples.

Algorithm 2: Proposed algorithm for classification of error-potentials

Input : raw EEG signals (X)

$X_f \leftarrow$ filtering(X, $freq\_band$, $filter\_order$) ;
/* Spatial Filtering */
$X_C^S \leftarrow$ covariance(X_f) ;
$X_D^S \leftarrow$ electrode_select (X_C^S, $nelec$) ;
$X_T^S \leftarrow$ tangent_space(X_D^S) ;
$X_N^S \leftarrow$ normalization(X_T^S, norm="l1") ;
$p_s \leftarrow$ linear_classification(X_N^S, λ_1, λ_2) ;
/* Frequency-domain */
$X_T^F \leftarrow$ multitaper_PSD(X_f, $time_f$, $freq_f$);
$X_N^F \leftarrow$ log_normalization(X_T^F) ;
$p_f \leftarrow$ svm(X_N^F) ;
/* Time-domain */
$X_B^T \leftarrow$ time_bucketing(X_f, $bucket\_size$);
$X_P^T \leftarrow$ average_power(X_B^T);
$X_N^T \leftarrow$ normalization(X_B^T, norm="l2") ;
$p_t \leftarrow$ svm(X_N^T) ;
/* Ensemble Learning */
$p \leftarrow$ soft_voting (p_s, p_f, p_t) ;
if $p > p_{th}$ **then return** True ;
else if $p < 1 - p_{th}$ **then return** False ;
else return None. ;

The algorithm parameters are explained in section 4.1

4.2 Evaluation

Here, we evaluate the performance of the proposed error-potential decoding algorithm, and compare it with the baseline algorithm.

Methodology: The code for the baseline algorithm was obtained from the public GitHub repository of the authors[7]. For the proposed algorithm, we have set the $filter\_order$ to 4 (for xDAWN Spatial Filtering), and, λ_1 and λ_2 to 0.001 and 0.02, respectively. The proposed algorithm is evaluated over the probability threshold parameter p_{th}. The algorithms are evaluated on the data collected for three environments, namely Maze, Catch, and Wobble (as explained in section 3.1). The evaluation was performed using a 10-fold cross-validation scheme, and a separate classifier is used for each subject and each game. We also present the *overall* performance over all the subjects and the game environments.

Metrics: We employ four different metrics to evaluate the performance gain of the proposed algorithm over the baseline algorithm. *(i) Accuracy* presents the average accuracy of both classes (ErrP and non-ErrP). *(ii) F1 Score* is the harmonic mean of recall and precision of the ErrP class, and provides an unbiased measure in the case of uneven class distribution. *(iii) Area Under Curve (AUC)* computes the area under the receiver operating characteristic curve and provides a measure of separability of the two classes. Finally, *(iv) Sample Efficiency* provides the percentage of data samples that can be confidently assigned to one class. Note that sample efficiency is 100% for the baseline algorithm, and proposed algorithm with $p_{th} = 0.5$, since none of the samples are discarded (i.e., all the samples are assigned to one of the classes). The sample efficiency decreases when p_{th} is increased over 0.5.

Performance: We present the overall detection accuracy of the proposed algorithm and compare it with the baseline in Fig. 4. The proposed algorithm without discarding any samples (p_{th}=0.5) performs with an average accuracy of 73.71% (± 6.81), an 8.11% improvement over the state-of-the-art. The accuracy is further boosted to 77.47% (13.6% improvement) and 79.51% (16.63% improvement) by increasing the p_{th} (dropping the low confidence samples) to 0.6 0.7 respectively. This improvement is achieved at the cost of sample efficiency of 88% (±6.01) and 72.3% (±13.33), for the p_{th} value of 0.5 and 0.6 respectively (as shown in Fig. 7). Among all three games, the accuracy rate of the Maze game (77.28%) is higher pertaining to its simple and intuitive user interface.

Fig. 5 presents the cumulative distribution of the accuracy over a total of 25 recordings. It can be noted that for 50% of samples, the baseline algorithm performs over 70%, while the proposed algorithm (with $p_{th=0.5}$) performs over 80%. This trend is more clearly seen in Fig. 5, where the cumulative distribution of the proposed algorithm with higher p_{th} lies over those with lower p_{th} and the baseline algorithm below all other. In Fig. 8, we present the cumulative distribution of sample efficiency over all subjects. The baseline algorithm and proposed algorithm (with $p_{th} = 0.5$) perform with 100% sample efficiency since no sample is dropped. However, increasing the low-confidence threshold range, i.e., p_{th}, the sample efficiency reduces. For $p_{th} = 0.6$, the sample efficiency is above 85% for at least 75% of the users, making the algorithm practical and universal for subjects. With $p_{th} = 0.7$, the classifier performs with very high accuracy, with a sample efficiency of over 50% for more than 90% of the users. This simply translates to the fact that one out of two error-potential can be effectively labeled with this approach.

The improvement in performance can also be observed from Fig. 9 in the AUC scores where the overall average over three games is 74.4% for baseline algorithm and 83.2% with $p_{th} = 0.7$ for proposed algorithm (9% improvement). Since the AUC score is independent of the classification threshold, we can see that the AUC score of the proposed algorithm for various values of p_{th} is similar. A similar trend is observed from Fig. 6 in the F1 scores with over 20% increase on average on all three games.

5 CONCLUSIONS AND FUTURE WORK

The context of this paper is the reliable detection of intrinsic reactions using the brain waves of the user detected using only an off-the-shelf EEG wearable. For Human-In-The-loop Reinforcement Learning (HITL-RL) systems, the detection accuracy of error-potentials plays a significant role in the convergence time of the RL algorithm. We present a new ErrP decoding algorithm leveraging multi-dimensional aspects of the EEG (namely, spatial, frequency, and time-domain) to increase the accuracy of detecting ErrP. The proposed algorithm is capable of selective use of high-confidence estimates to further improve the accuracy at the expense of sample efficiency. We also provide the system architecture consisting of

[7] https://github.com/alexandrebarachant/bci-challenge-ner-2015

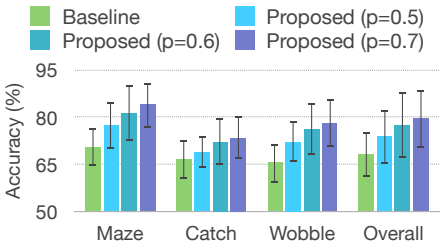

Figure 4: Accuracy

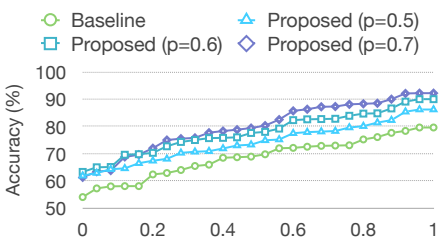

Figure 5: Accuracy CDF

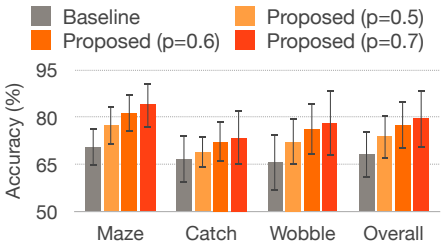

Figure 6: F1 Score

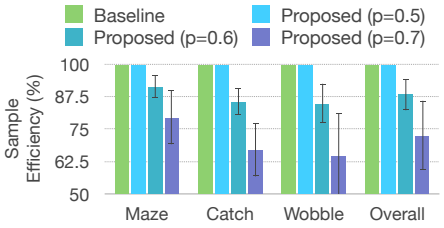

Figure 7: Sample Efficiency

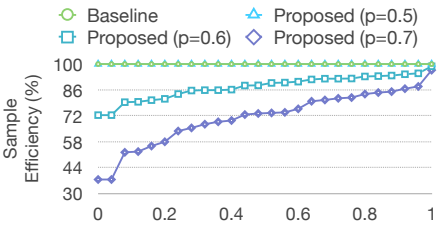

Figure 8: Sample Efficiency CDF

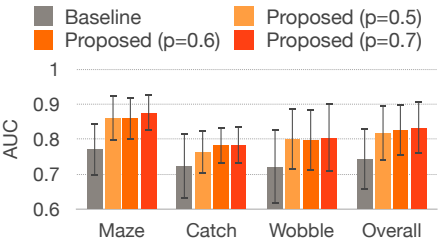

Figure 9: Area Under Curve

a wearable EEG headset with a mobile device and a cloud server utilizing intrinsic human brain potentials to achieve acceleration in convergence time of RL algorithms.

We plan to extend the study of HITL-RL for a variety of complex game environments. The cost of sample efficiency (discarding low-confidence samples) and the effect of the difficulty of the games on the accuracy of ErrP are the subjects of future study.

Acknowledgments: This work was supported in part by the Wayne J. Holman Endowed Chair and the National Science Foundation under grants CNS-1813242 and CPS-1837369.

REFERENCES

[1] W Bradley Knox and Peter Stone. Reinforcement learning from simultaneous human and mdp reward. In *AAMAS*, pages 475–482, 2012.

[2] Ajinkya Jain and Scott Niekum. Learning hybrid object kinematics for efficient hierarchical planning under uncertainty. *arXiv preprint arXiv:1907.09014*, 2019.

[3] Andres F Salazar-Gomez, Joseph DelPreto, Stephanie Gil, Frank H Guenther, and Daniela Rus. Correcting robot mistakes in real time using eeg signals. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6570–6577. IEEE, 2017.

[4] Iñaki Iturrate, Luis Montesano, and Javier Minguez. Robot reinforcement learning using eeg-based reward signals. In *2010 IEEE International Conference on Robotics and Automation*, pages 4822–4829. IEEE, 2010.

[5] Duo Xu, Mohit Agarwal, Faramarz Fekri, and Raghupathy Sivakumar. Playing games with implicit human feedback. *Workshop on Reinforcement Learning in Games, AAAI*, 2020.

[6] Alexandre Barachant, Stéphane Bonnet, Marco Congedo, and Christian Jutten. Multiclass brain–computer interface classification by riemannian geometry. *IEEE Transactions on Biomedical Engineering*, 59(4):920–928, 2011.

[7] Alexandre Barachant, Stéphane Bonnet, Marco Congedo, and Christian Jutten. Classification of covariance matrices using a riemannian-based kernel for bci applications. *Neurocomputing*, 112:172–178, 2013.

[8] Mohit Agarwal and Raghupathy Sivakumar. Think: Toward practical general-purpose brain-computer communication. In *Proceedings of the 2Nd International Workshop on Hot Topics in Wireless*, pages 41–45, 2015.

[9] Mohit Agarwal and Raghupathy Sivakumar. Blink: A fully automated unsupervised algorithm for eye-blink detection in eeg signals. In *2019 57th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 1113–1121. IEEE, 2019.

[10] Mohit Agarwal and Raghupathy Sivakumar. Cerebro: A wearable solution to detect and track user preferences using brainwaves. In *The 5th ACM Workshop on Wearable Systems and Applications*, pages 47–52, 2019.

[11] Mohit Agarwal and Raghupathy Sivakumar. Charge for a whole day: Extending battery life for bci wearables using a lightweight wake-up command. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–14, 2020.

[12] Michael Falkenstein, Jörg Hoormann, Stefan Christ, and Joachim Hohnsbein. Erp components on reaction errors and their functional significance: a tutorial. *Biological psychology*, 51(2-3):87–107, 2000.

[13] Greg Hajcak and Dan Foti. Errors are aversive: Defensive motivation and the error-related negativity. *Psychological science*, 19(2):103–108, 2008.

[14] Wolfgang HR Miltner, Ulrike Lemke, Thomas Weiss, Clay Holroyd, Marten K Scheffers, and Michael GH Coles. Implementation of error-processing in the human anterior cingulate cortex: a source analysis of the magnetic equivalent of the error-related negativity. *Biological psychology*, 64(1-2):157–166, 2003.

[15] Nico M Schmidt, Benjamin Blankertz, and Matthias S Treder. Online detection of error-related potentials boosts the performance of mental typewriters. *BMC neuroscience*, 13(1):19, 2012.

[16] Cameron S Carter, Todd S Braver, Deanna M Barch, Matthew M Botvinick, Douglas Noll, and Jonathan D Cohen. Anterior cingulate cortex, error detection, and the online monitoring of performance. *Science*, 280(5364):747–749, 1998.

[17] Pierre W Ferrez and José del R Millán. You are wrong!—automatic detection of interaction errors from brain waves. In *Proceedings of the 19th international joint conference on Artificial intelligence*, number CONF, 2005.

[18] Martin Spüler and Christian Niethammer. Error-related potentials during continuous feedback: using eeg to detect errors of different type and severity. *Frontiers in human neuroscience*, 9:155, 2015.

[19] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.

[20] Yann Renard, Fabien Lotte, Guillaume Gibert, Marco Congedo, Emmanuel Maby, Vincent Delannoy, Olivier Bertrand, and Anatole 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*, 19(1):35–53, 2010.

[21] Andrew Y Ng, Daishi Harada, and Stuart Russell. Policy invariance under reward transformations: Theory and application to reward shaping. In *ICML*, volume 99, pages 278–287, 1999.

[22] Bertrand Rivet, Antoine Souloumiac, Virginie Attina, and Guillaume Gibert. xdawn algorithm to enhance evoked potentials: application to brain–computer interface. *IEEE Transactions on Biomedical Engineering*, 56(8):2035–2043, 2009.

[23] Alexandre Barachant and Marco Congedo. A plug&play p300 bci using information geometry. *arXiv preprint arXiv:1409.0107*, 2014.

[24] Marco Congedo, Alexandre Barachant, and Anton Andreev. A new generation of brain-computer interface based on riemannian geometry. *arXiv preprint arXiv:1310.8115*, 2013.

[25] Alexandre Barachant and Stephane Bonnet. Channel selection procedure using riemannian distance for bci applications. In *2011 5th International IEEE/EMBS Conference on Neural Engineering*, pages 348–351. IEEE, 2011.

[26] Bianca Zadrozny and Charles Elkan. Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers. In *Icml*, volume 1, pages 609–616. Citeseer, 2001.

[27] Donald B Percival, Andrew T Walden, et al. *Spectral analysis for physical applications*. cambridge university press, 1993.

[28] John Platt et al. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74, 1999.