Accelerating Reinforcement Learning Agent with EEG-based Implicit Human Feedback

Duo Xu*, Mohit Agarwal*, Ekansh Gupta, Faramarz Fekri, Raghubathy Sivakumar
Georgia Institute of Technology
Atlanta, GA, 30332
Emails: {dxu301, mohit, fekri, siva}@ece.gatech.edu

Abstract—Providing Reinforcement Learning (RL) agents with human feedback can dramatically improve various aspects of learning. However, previous methods require human observer to give inputs explicitly (e.g., press buttons, voice interface), burdening the human in the loop of RL agent’s learning process. Further, it is sometimes difficult or impossible to obtain the explicit human advise (feedback), e.g., autonomous driving, disabled rehabilitation, etc. In this work, we investigate capturing human’s intrinsic reactions as implicit (and natural) feedback through EEG in the form of error-related potentials (ErrP), providing a natural and direct way for humans to improve the RL agent’s learning. As such, the human intelligence can be integrated via implicit feedback with RL algorithms to accelerate the learning of RL agent. We develop three reasonably complex 2D discrete navigational games to experimentally evaluate the overall performance of the proposed work. Major contributions of our work are as follows, (i) we propose and experimentally validate the zero-shot learning of ErrPs, where the ErrPs can be learned for one game, and transferred to other unseen games, (ii) we propose a novel RL framework for integrating implicit human feedbacks via ErrPs with RL agent, improving the label efficiency and robustness to human mistakes, and (iii) compared to prior works, we scale the application of ErrPs to reasonably complex environments, and demonstrate the significance of our approach for accelerated learning through real user experiments.

I. INTRODUCTION

AI systems are increasingly applied to real-world tasks that involve interaction with humans. And humans are often in the loop of the RL agent’s learning process. Self-driving cars learn with humans ready to intervene in dangerous situations. Facebook’s algorithm for recommending trending news stories has humans filtering out inappropriate content. Therefore RL with human-in-the-loop has inspired several research efforts where either an alternative (or supplementary) feedback is obtained from the human participant, such as human rankings or ratings [18], human robot interaction and rehabilitation engineering for the disabled [32], [36], or the learning is performed through human demonstrations [41]. Such approaches with explicit human input despite being highly effective, severely burdens the human interacting with RL agent. Further, it is difficult or even impossible to obtain the explicit human feedback in various situations, e.g., autonomous driving, disabled users, etc.

In this work, we investigate an alternative paradigm to obtain the human feedback in an implicit manner (by tapping directly into the intrinsic brainwaves) that substantially increases the richness of the reward functions, while not severely burdening the human-in-the-loop. We study the use of electroencephalogram (EEG) based brain waves of the human-in-the-loop to generate the auxiliary reward functions to augment the learning of RL agent. Such a model will benefit from the natural rich activity of a powerful sensor (the human brain), but at the same time not burden the human since the activity being relied upon is intrinsic. This paradigm is inspired by a high-level error-processing system in humans that generates error-related potential (ErrP) [50], [6], a negative deflection in the ongoing EEG signals. When a human recognizes an error made by an agent, the elicited ErrP can be captured through EEG to inform agent about the sub-optimality of the taken action in the particular state. Human feedback obtained in this manner is direct and fast while being natural and easy for humans. This widens the applicability of such RL-human interactive systems where the RL agents are deployed in the real-world environment, and increased latency of human feedback could create unwanted situations. Further, obtaining large amount of explicit feedback is infeasible due to the increased cognitive load [45]. Additionally, EEG-based feedback allows disabled users to provide the feedback, where explicit communication pathway is not available.

Previous works have [14], [48] demonstrated the benefit of error-potentials in a very simple setting (i.e., very small state-space, and two actions), and used ErrPs as the only reward. As a baseline contribution, we scale the feasibility of capturing error-potentials (of a human observer watching an agent learning to play games) to reasonably complex environments, and then experimentally show that decoded ErrPs can be appropriately used as an auxiliary reward function to a RL agent. Specifically, we show that the full access approach, inquiring human feedback on every state-action pair visited by RL agent, can significantly speedup the learning of RL agent. However, we argue that while obtaining such implicit human feedback through EEG is less burdensome, it is still a time-intensive task for the subject and the experimenter alike. This, combined with the noisy EEG signals and stochasticity in inferring error-potentials, raises significant challenges in terms of the practicality of the solution.

In this context, we first argue that the definition of ErrPs can be learned in a zero-shot manner across different environments. We experimentally validate that ErrPs of an observer can be learned to decode for a specific game, and the
definition can be used as-is for another game without requiring re-learning of the ErrP. This is notably different from previous approaches [14], [48], where the labeled ErrPs are obtained in the same environment the RL agent is trying to solve. We contend that previous approaches are not practical, since ErrP decoder cannot be trained and tested in the same environment.

We develop a framework to integrate deep RL (DRL) model with the implicit human feedback mechanism (via ErrP) in a practical, sample-efficient manner. Our proposed framework allows humans to provide their feedback implicitly prior to the agent training, reducing the cognitive load on humans, and hence the cost of human supervision. In the presented framework, prior to the training of RL agent, randomly generated demonstrations are presented to human for giving feedback (implicitly via ErrP), and an auxiliary reward function is learned to reflect the human decision and intelligence hidden behind ErrP labels. This auxiliary reward is then passed to the RL agent to accelerate the learning process in sparse-reward environments. Similar previous work lies in the streamline of human-agent interaction via reward shaping [8], [10], [34], [63], [65], [60]. However, their methods didn’t treat errors in the human feedback specifically, so that they cannot be robust to wrong ErrP labels, generated due to the randomness of collecting and decoding of ErrP data. Thus, we learn an auxiliary reward of human feedback in a way of being robust to wrong ErrP labels. We first model the human policy as a soft-Q policy [28] and learn the human Q function via maximum likelihood based on collected ErrP labels. Then in order to make the learned Q function more compatible with the state space, a baseline function is introduced to smoothen that. Finally, at the RL agent side, the received reward is the combination of environmental reward (sparse) and the auxiliary reward learned from human feedback.

We present results of real ErrP experiments to evaluate the acceleration in learning, and sample efficiency, of the proposed frameworks. We show that such implicit feedback approach can accelerate the training of RL agent by 2.25x, while reducing the number of queries required by 75.56%. In summary, the novel contributions of our work are,

1) We demonstrate the zero-shot learning of error-potentials over various visual-based RL problems (discrete grid-based navigation games, studied in this work), enabling the estimation of implicit human feedback in new and unseen environments without re-training of ErrP decoder.

2) In order to reduce the sample complexity of ErrP labels, we propose a new framework of integrating human feedback into RL via reward shaping. It is a novel approach specifically considering robustness against mistakes in human feedback. We first generate a set of random trajectories by Monte Carlo Tree Search (MCTS), balancing exploration and exploitation. Then ErrP labels are collected in experiments by demonstrating these trajectories to human observers. By learning human Q function with decoded labels, we derive an auxiliary reward function to augment the learning of following RL agent.

3) We scale the implicit human feedback (via ErrP) based RL to reasonably complex environments and demonstrate the significance of our approach through experiments on various human subjects.

Our work demonstrates the potential of intuitive human robot interaction, facilitating robotic control by implicit human feedback in the form of ErrPs. We believe the contribution presented in this work, i.e., zero-shot learning of ErrPs and RL framework to reduce the human cognitive load, would inspire such implicit human feedback system to be deployed in practical robotic applications, such as autonomous driving or end-user applications for disabled, where explicit human feedback is not available.

II. Related Work

The impact of feedback provided by a human to an agent in RL settings has been investigated by multiple researchers. A survey of recent research in using human guidance for deep RL tasks is presented in [62]. We summarize related work in some of these techniques that are most relevant to us. In addition to rewards from environment, reward shaping learns an auxiliary reward function to accelerate the learning process of the agent [12], [13], [54]. A framework called TAMER (Training an Agent Manually via Evaluative Reinforcement) that enabled shaping (interactively training an agent via an external signal provided by a human) was presented in [34]. Then the author extended this work to enable human feedback to augment an RL agent that learned using an MDP reward signal [35], [56]. Recently an architecture called Deep-TAMER [57] has extended the TAMER framework to environments with high-dimensional state spaces. DQN-TAMER [1] modeled other characteristics of human observers, such as facial expressions, from which human reward was inferred.

Human preference [15], [59] is another approach to communicate complex goals to allow systems to interact with real-world environments in a meaningful way. This allowed the RL agent to directly learn from expert preferences. However, this approach is limited by assumptions on the existence of a (total) order among the set of trajectories. The author proposed a framework called Human-Agent Transfer (HAT) [53]. It directly used demonstrations provided by a human operator to synthesize a baseline policy, which is to guide the learning of the agent. CHAT [56] extended HAT to consider uncertainty in summarizing demonstrations and further improve the performance.

‘Potential functions’ is also used in Potential-based reward shaping (PBRS) methods to accelerate the learning process, while preserving the identity of optimal policies [41], [58], [61]. The potential function was designed to encode ‘rules’ of the environment of the RL agent. However, potential functions will typically need to be pre-specified. This has restricted the use of PBRS to tabular / low-dimensional state spaces.

In previous reward-shaping work mentioned above, human feedback is explicit, requiring active human labeling or attention, and the mistakes in human feedback are not tackled. Here we propose to read implicit human feedback from error-potential hidden in human brain waves, and deal with wrong feedback in a robust approach. Recently, there is a long line of papers studying reinforcement learning from human feedback, such as [17], [18], [55], [59], [15]. However, they
are only about explicit human feedback or labeling, and they all assume human feedback is noiseless. In this work, we use reward function learned by imitation learning to augment the following RL agent.

Numerous works [11, 31, 50] have studied a high-level error-processing system in humans generating the error-related potential/negativity (ErrP or ERN).

Interaction, response, and feedback ErrPs have been heavily investigated in the domain of choice reaction tasks, where human is actively interacting with the system [49, 7, 44, 22, 23] and the error is made either by the human or by the machine. 53 demonstrated the use of ErrP signals in an interactive RL task, when the human is actively interacting with the machine system. 22 explored the ErrPs when human is silently observing the machine actions (and does not actively interact). Works at the intersection of ErrP and RL [14], [48] demonstrate the benefit of ErrPs in a very simple setting (i.e., very small state-space), and use ErrP-based feedback as the only reward. Moreover, in all of these works, the ErrP decoder is trained on a similar game (or robotic task), essentially using the knowledge that is supposed to be unknown in the RL task. In our work, we use labeled ErrPs examples of very simple and known environments to train the ErrP decoder, and integrate ErrP with DRL in a sample-efficient manner for reasonably complex environments.

III. PRELIMINARIES AND SETUP

Definitions: Consider a Markov Decision Process (MDP) problem \( M \), as a tuple \(< \mathcal{X}, \mathcal{A}, P, P_0, R, \gamma >\), with state-space \( \mathcal{X} \), action-space \( \mathcal{A} \), transition kernel \( P \), initial state distribution \( P_0 \), accompanied with reward function \( R \), and discounting factor \( 0 \leq \gamma \leq 1 \). In this work, we only consider MDP with discrete actions and states. In model-free RL method, the central idea of most prominent approaches is to learn the Q-function by minimizing the Bellman residual, i.e., \( \mathcal{L}(Q) = \mathcal{E}_\pi [ (Q(x, a) - r - \gamma Q(x', a))^2 ] \), and temporal difference (TD) update where the transition tuple \((x, a, r, x')\) consists of a consecutive experience under behavior policy \( \pi \).

Bayesian Deep Q Network: The Q function model adopted in this paper is Bayesian DQN [2]. It is a neural architecture where the Q-function is approximated as a linear function, weighted by \( \omega_a, a \in \mathcal{A} \), of the feature representation of states \( \phi_q(x) \in \mathcal{R}^d \), parameterized by neural network with weights \( \theta \). The weights \( \omega_a \) follow the Gaussian distribution from Bayesian linear regression.

A. System Setup and Data Collection

We designed and developed an experimental protocol, where a machine agent plays a computer game, while a human silently observes (and assesses) the actions taken by the machine agent. These implicit human reactions are captured by placing raw electrodes on the scalp of the human brain in the form of EEG potentials. The electrode cap (BIOPAC CAP-100C) was attached with the OpenBCI Cyton platform, which was further connected to a desktop machine over the wireless channel. In the game design (developed on OpenAI Gym), we open a TCP port, and continuously transmit the current state-action pair using the TCP/IP protocol. We used OpenViBE software [46] to record the human EEG data. OpenViBE continuously listens to the TCP port (for state-action pairs), and timestamps the EEG data in a synchronized manner. A total of five human subjects were recruited (mean age 26.8 with standard deviation of 1.92, 1 female) using standard procedures with their consent. For each subject-game pair, the experimental duration was less than 15 minutes. The agent took action every 1.5 seconds during the experiment. All the research protocols for the user data collection were reviewed and approved by the University Institutional Review Board.

1) Game Environments: We have developed three discrete grid-based navigational games in OpenAI Gym framework, namely Wobble, Catch, and Maze (Fig. 1(a)).

Wobble: Wobble is a simple 1-D cursor-target game, where the middle horizontal plane is divided into 20 discrete blocks. At the beginning of the game, the cursor appears at the center of the screen, and the target appears no more than three blocks away from the cursor position. The action space constitutes moving to the left or right. The game is finished when the cursor reaches the target. Once the game is finished, a new game is started with the cursor in place.

Catch: Catch is a simplistic version of Eggomania [2] (Atari 2600 benchmark), where we display a single egg on the screen at a time. The screen dimensions are divided into 10x10 grid space, where the egg and the cart, both occupies one block. The action space of the agent consists of NOOP (no operation), left and right. At the start of the game, the horizontal position of the egg is chosen randomly. At each time step, the egg falls one block in the vertical direction.

Maze: Maze is a 2-D navigational game, where the agent has to reach to a fixed target (shown with a plus symbol). The screen is divided into 10x10 square blocks. The action space consists of four directional movements. The only reward here is the result of the episode, i.e., win or lose. If an agent moves, but hits a wall, a quick blinking of the agent is displayed, to render the action taken by the agent.

B. Advantages of using error-potentials

In our work, relying on error-potentials provides two primary benefits:

(a) Provides a generalization notion of error-detection:
Error-potentials are elicited by incorrect feedback in a diverse set of tasks [25] implying that the error processing system is sensitive to a generalized notion of error detection. Error-potentials are observed across a wide variety of input modality (e.g., audio [19], visual [21], somatosensory [40], etc.). This generalized mechanism is one of the characteristic advantage in error-potentials, unlike other brain-potentials specific to a stimulus or modality. For instance, the P600, N300, P300, and N200 are elicited when a subject is presented with syntactic

1http://openbci.com

2https://en.wikipedia.org/wiki/Eggomania
of the impact of time pressure on mental comfort and accuracy. The first instance had a time delay of 1.5 seconds between successive actions of the agent while the second and the third had a delay of 1.0 and 0.5 seconds respectively. Subjects were required to play 3 trials in each instance (thus totaling to 9 trials overall) where the order of instances was randomized to avoid biasing the users to a particular order of the game. Once subjects finished playing 3 trials of an instance, they were redirected to a Qualtrics survey where they had to provide their subjective feedback about the experiment. Thus, there were 3 forms that each subject had to fill (one per instance). We used Amazon’s Mechanical Turk to request anonymous workers to complete this task. The study protocol was approved by the Institutional Review Board.

Results: We obtained a total of 281 responses (87, 91, and 103 unique user responses for the 1.5s, 1.0s, and 0.5s instances of the game respectively). On average, for the 1.5s instance of the game, we obtained a true positive rate of 56.6% and 41.5% for correct and incorrect actions of the maze agent respectively. We also obtained a feedback latency of 376ms and 540ms for correct and incorrect actions of the maze agent respectively. Note that correct and incorrect actions of the agent corresponds to the non-Errp and ErrP respectively, during EEG experiments. For the 1.0s instance of the game, we obtained a true positive rate of 49.8% and 38.8% for correct and incorrect actions of the maze agent respectively. We also obtained a feedback latency of 288ms and 456ms for correct and incorrect actions of the maze agent respectively. For the 0.5s instance of the game, we obtained a true positive rate of 34.9% and 14.1% for correct and incorrect actions of the maze agent respectively. We also obtained a feedback latency of 179ms and 207ms for correct and incorrect actions of the maze agent respectively. However some trials in these experiments had very poor labeling rate (some subjects only labeled less than 50% of the actions available). In order to prevent the results from being swayed by inert participants, we decided to separate them from the operational participants.

We decided to remove the trials for the users which had less than 50% feedback rate. In other words, we removed the trials where participants failed to provide the feedback for at least 50% of all the actions. On average, for the 1.5s instance of the game, we obtained a true positive rate of 74.1% and 53.4% for correct and incorrect actions of the maze agent respectively. We also obtained a feedback latency of 364ms and 539ms for correct and incorrect actions of the maze agent respectively. For the 1.0s instance of the game, we obtained a true positive rate of 69.8% and 52.6% for correct and incorrect actions of the maze agent respectively. We also obtained a feedback latency of 290ms and 451ms for correct and incorrect actions of the maze agent respectively. For the 0.5s instance of the game, we obtained a true positive rate of 56.4% and 21.6% for correct and incorrect actions of the maze agent respectively. We also obtained a feedback latency of 177ms and 203ms for correct and incorrect actions of the maze agent respectively. These values are summarized in Table I

Insights: As we can clearly see, the accuracy values for correct and incorrect actions both decrease with decrease in time interval. The labeling accuracy for correct actions seems anomalies in sentences [43], semantically inconsistent word and picture pairs [39], interruption of a stimulus with another divergent stimulus [52], and detection of mismatch in a stimulus [24] respectively.

(b) Evolutionary Significance: Error-potentials in primates are strong and universal (exhibiting similar behaviours across individuals) as they have an evolutionary significance due to their importance in cognition, learning, and survival. Error-potentials enable the learning process via the administration of rewards and punishments in Anterior Cingulate Cortex (ACC) [29]. In monkeys, error-potentials were generated in anterior cingulate sulcus, when monkeys made errors in a simple response task. The are the are linked to the. [42] found error-recognition units in monkeys’ anterior cingulate sulcus that were activated when the animals received negative feedback in the form of absence of an expected reward. Similarly, [26] found that when monkeys made errors in a simple response task, error-related potentials were generated in the anterior cingulate sulcus thereby advocating that ErrPs link human and non human primates on the basis of error monitoring.

C. Motivation: Using intrinsic error-potentials over manual labeling

Experimental Methodology: We conducted an experiment in which subjects were asked to label the actions of an AI agent in a maze. If the agent took a correct action, they needed to press a certain key and if the agent performed a wrong action, they were supposed to press another key. We conducted this experiment to find differences between manual labeling and labeling using EEG experiments in terms of user comfort and labeling accuracy. The methodology of this experiment was simple. We designed a maze game and generated 3 instances of it where each instance got progressively faster (to study
to be more than that of incorrect actions. Both, the accuracy for correct as well as incorrect actions decreases as the time latency is decreased (thereby increasing time pressure). Even the best possible accuracy for incorrect actions is about 53.4% (only marginally better than random labeling). This performs rather poorly compared to the labeling accuracy using ErrP.

Based on the qualitative survey responses, on a scale of 1 to 7, the users gave the 1.5s instance of the game an average comfort rating of 5.4 which declined to 4.9 and 3.9 for the 1.0s instance and 0.5s instance respectively. On being asked if they were able to mark all actions correctly, 40% of the subjects answered in the affirmative in the 1.5s instance of the game, which declined to 26% and 14% in the 1.0s and the 0.5s instance of the game. Across the board, the majority of the participants reported that the ideal time interval for them to correctly label all actions of the agent would be between 1.5s and 3.0s or larger. 64% of the participants in the 1.5s instance of the game reported that reducing the time interval of the game to 1.0s would decrease their labeling accuracy, and 69% of the participants reported that it would increase their mental burden. 52% of the participants in the 1.0s instance of the game reported that reducing the time interval of the game to 0.5s would decrease their labeling accuracy, and 60% of the participants reported that it would increase their mental burden. In contrast, 64% of the participants in the 1.0s instance of the game reported that increasing the time interval from 1.0s to 1.5s would increase their labeling accuracy and decrease their mental burden. 49% of the participants in the 0.5s instance of the game reported that reducing the time interval of the game further would decrease their labeling accuracy, and 53% of the participants reported that it would increase their mental burden. To summarize, the users felt increasing discomfort and cognitive burden as the time latency reduced from 1.5s to 1.0s and further to 0.5s. They also reported that the optimal time latency for comfortable manual labeling would be between 1.5s and 3s. This was also evident from the fact that more than 60% of the participants anticipated reduction in their accuracy if time latency was to be decreased from 1.5s.

### IV. Integrating RL with Implicit Human Feedback: A Naive Approach

In this section, we provide our baseline contribution, i.e., (i) we demonstrate the feasibility of capturing error-potentials of a human subject watching an RL agent learning to play several different games, and then decoding the human feedback (judgment) on the observed state-action pair appropriately, and (ii) using them as an auxiliary reward function to accelerate the learning of the RL agent.

### TABLE I: Accuracy and latency for maze game manual labeling

| Time Interval (s) | Subjects | Non-ErrP TPR (%) | ErrP TPR (%) | Non-ErrP latency (ms) | ErrP latency (ms) |
|-------------------|----------|------------------|-------------|-----------------------|------------------|
| 1.5               | 87       | 74.1             | 53.4        | 364                   | 539              |
| 1.0               | 91       | 69.8             | 52.6        | 290                   | 451              |
| 0.5               | 103      | 56.4             | 21.6        | 177                   | 203              |

### A. Obtaining the Implicit Human Feedback: Decoding ErrPs

We rely on the Riemannian Geometry framework for the classification of human’s intrinsic reaction (captured in the form of ErrPs) [5]. We consider the classification of error-related potentials as a binary classification task indicating the presence (i.e., action taken by the agent is incorrect) and absence of error (i.e., action taken by the agent is correct). The raw EEG data is bandpass filtered in [0.5, 40] Hz. Epochs of 800ms were extracted relative to pre-stimulus 200ms baseline, and were subjected to spatial filtering. In spatial filtering, prototype responses of each class, i.e., “correct” and “erroneous”, are computed by averaging all training trials in the corresponding classes (“xDAWN Spatial Filter” [47], [16]). “xDAWN filtering” projects the EEG signals from sensor space (i.e., electrode space) to the source space (i.e., a low-dimensional space constituted by the actual neuronal ensembles in brain firing coherently). The covariance matrix of each epoch is computed, and concatenated with the prototype responses of the class. Further, dimensionality reduction is achieved by selecting relevant channels through backward elimination [3].

The filtered signals are projected to the tangent space for feature extraction. The obtained feature vector is first normalized (using L1 norm) and fed to a regularized regression model. A threshold value is selected for the final decision by maximizing accuracy offline on the training set. We present the algorithm to decode the ErrP signals in Algorithm 1.

#### Algorithm 1: Riemannian Geometry based ErrP classification algorithm

```plaintext
Input : raw EEG signals EEG
1 Pre-process raw EEG signals ;
2 Spatial Filtering: xDAWN Spatial Filter (nfilter) ;
3 Electrode Selection: ElectrodeSelect (nelec, metric='riemann') ;
4 Tangent Space Projection : TangentSpace(metric = “logeuclid”) Normalize using L1 norm ;
5 Regression: ElasticNet ;
6 Select decision threshold by maximizing accuracy
```

### B. The Full Access Method

A naive approach to integrate the human feedback with RL models is reward shaping with full access. Human feedback is obtained on every visited state-action pair while RL agent is learning, and a negative penalty is added to the environmental reward in case an ErrP is detected. The evaluation result of this method based on real ErrP data are presented later in the evaluation section (section VI-A), validating that full access method can significantly accelerate the learning of the RL agent. However, obtaining the human feedback for every state-action pair is time-intensive and not practically feasible. In the next section, we provide our novel contributions to practically obtain and integrate the implicit feedback with the learning of RL agent.

Interaction, response, and feedback ErrPs have been heavily investigated in the domain of choice reaction tasks, where
human is actively interacting with the system \[49\], \[7\], \[44\], \[22\], \[23\] and the error is made either by the human or by the machine. \[33\] demonstrated the use of ErrP signals in an interactive RL task, when the human is actively interacting with the machine system. \[22\] explored the ErrPs when human is silently observing the machine actions (and does not actively interact). Works at the intersection of ErrP and RL \[14\], \[48\] demonstrate the benefit of ErrPs in a very simple setting (i.e., very small state-space), and use ErrP-based feedback as the only reward. Moreover, in all of these works, the ErrP decoder is trained on a similar game (or robotic task), essentially using the knowledge that is supposed to be unknown in the RL task. In our work, we use labeled ErrPs examples of very simple and known environments to train the ErrP decoder, and integrate ErrP with DRL in a sample-efficient manner for reasonably complex environments.

V. TOWARDS PRACTICAL INTEGRATION OF RL WITH IMPLICIT HUMAN FEEDBACK

In this section, we propose two approaches to enable the deployment of ErrP-augmented RL into practical systems. Firstly, we show that ErrPs of an observer can be learned in a zero-shot manner, i.e. ErrP decoder can be trained for a specific game, and the trained decoder can be used as-is for another game without re-training of the ErrP decoder. To combat with the practical issues with obtaining ErrP labels for every state-action pairs, we propose an RL framework (motivated by imitation learning approaches) allowing humans to provide their feedback on a few trajectories prior to the learning of the RL agent. This dramatically reduces the number of feedback labels required from the human observer.

A. Zero-shot learning of ErrPs

Error-potentials in the EEG signals are studied under two major paradigms in human-machine interaction tasks, (i) feedback and response ErrPs: error made by human \[11\], \[20\], \[2\], \[44\], (ii) interaction ErrPs: error made by machine in interpreting human intent \[22\]. Another interesting paradigm is observation ErrPs, when human is watching (and silently assessing) the machine performing a specific task \[14\]. The manifestation of these potentials across these paradigms were found quite similar in terms of their general shape, negative and positive peak latency, and frequency characteristics\[22\], \[14\]. This prompts us to explore the consistency of the error-potentials across different environments (i.e., games, in our case) within the observation ErrPs. In Figure \[2\] we plot the grand average waveforms across three environments (Maze, Catch and Wobble), to visually validate the consistency of potentials. We can see that the shape of negativity, and the peak latency is quite consistent across the three game environments. Further, in evaluation section \[VI-B1\] we experimentally demonstrate the zero-shot learning of error-potentials.

B. Robust Reward Shaping using Human Feedback

RL algorithms deployed in the environment with sparse rewards demand heavy explorations (require a large number of trial-and-errors) during the initial stages of training. Previous work on reward shaping with human feedback \[10\], \[34\], \[53\], \[57\], \[60\] build a specific model to generalize human feedback in state space, without tackling wrong feedback. Inspired by soft Q policy \[28\], we develop a novel framework of learning the auxiliary reward from human feedback to accelerate the training of the RL agent, with robustness to mistakes in ErrP labeling.

In this framework, implicit human feedback is required on all state-action pairs along trajectories (demonstrations) randomly generated initially. Before RL agent starts learning, human subjects are asked to observe a number of trajectories, and their implicit feedback in the form of ErrP on corresponding state-action pair are recorded in a dataset. Then the auxiliary reward function \(r_a(\cdot, \cdot)\) is learned from these trajectories labeled by human feedback. It also discovers the human decision or intention hidden behind the implicit feedback, in the form of ErrPs. During the RL training, the learned reward function acts as a proxy for the human feedback, compensating the very sparse reward of environment. The flowchart of the proposed learning framework is shown in Figure \[3\] Different from the naive baseline full access method, in this approach, queries for human feedback (ErrP labeling) are required only on trajectories generated initially, instead of querying every learning step during the training. Hence, the total number of ErrP queries can be reduced significantly, further reducing the cognitive load for the human-in-the-loop on the implicit feedback.

Trajectory Generation: Constraint by the coherence requirement in EEG experiments, the trajectories for ErrP labeling have to be complete, containing every state-action pair from the beginning to the end of the game. However, the selected trajectories have to cover state space as much as possible, and cannot be too far away from the optimal solutions. This is essentially the trade-off between exploitation and exploration. So we propose to use Monte Carlo Tree Search (MCTS) \[9\], \[37\]. \[51\] to generate random trajectories for ErrP experiments. It is to tackle exploration-exploitation trade-off by Upper Confidence Bound (UCB) method \[37\], and doesn’t require to know the optimal solution a prior. MCTS is a general game playing technique with recent success in discrete, turn-based, and non-deterministic game domains. We choose MCTS as a trajectory sampling algorithm for its proven high-level performance, domain generality, and variable computational bounds.

There is one node in the tree for each state \(s\), containing a value \(Q(s,a)\) and a visit count \(N(s,a)\) for each action \(a\), and an overall count \(N(s) = \sum_a N(s,a)\). Each node is initialised to \(Q(s,a) = 0, N(s,a) = 0\). The value is estimated by the mean return from \(s\) in all simulations where action \(a\) was selected from state \(s\), and the only reward here is the result of the game, win or lose. At each state \(s\) of the trajectory, the action is selected to be the maximizer of the objective \(Q(s,a) + c \sqrt{\log N(s,a) / N(s)}\), where \(c\) is to trade off between reaching the target and exploring more state space \[37\]. By the end of generating each trajectory, the return is back-propagated into \(Q\) values along the trajectory, i.e., \(Q(s_t,a_t) := r + \gamma Q(s_{t+1},a_{t+1})\). Only first \(K\) generated
trjectories are used in ErrP experiments.

In experiments for collecting ErrPs, the human subject provides implicit feedback (via ErrP) over all the generated trajectories, labeling every state-action pair as a positive or negative sample corresponding to its correctness according to human intelligence. With decoded ErrP labels over trajectories as input, we propose a novel reward shaping method to incorporate ErrP labels into the following reinforcement learning process. It specifically tackles the problem of robustness against wrong ErrP labels, with details explained in the following section.

**Reward Learning** Since implicit human feedback via ErrP is noisy, different from previous work, we don’t use a neural network to approximate the human feedback directly. In order to improve the robustness to errors, we first model the human policy as soft-Q policy, i.e., a popular energy-based policy [27], [28], and learn the human Q function instead by solving a classification problem. Following the principle of maximum entropy [63], given human Q function $Q_h(\cdot, \cdot)$, the human policy distribution and value function can be expressed as follows,

$$
\pi_h(a|s) = \exp((Q_h(s, a) - V_h(s))/\alpha),
$$

$$
V_h(s) = \alpha \log \sum_a \exp(Q_h(s, a)/\alpha)
$$

(1)

where $\alpha$ is a free parameter, tuned empirically. The likelihood of positive and negative state-action pair are denoted as $\pi_h(a|s)$ and $1 - \pi_h(a|s)$. When trajectories and corresponding human feedback (ErrP labels) are ready, we learn the human Q function by maximizing the likelihood of both positive and negative state-action pairs in the trajectories, which is to maximize the objective [3], where the binary variable ErrP($s,a$) denotes the ErrP labels obtained in previous step. It is essentially a classification problem on states with ErrP as labels and Q function as logits. Hence, a naive choice for auxiliary reward for the following RL agent is the Bellman difference of human Q function, i.e., $Q_h(s, a) - \gamma \max_{a'} Q_h(s', a')$. However, due to the noise on ErrP decoding and lack of ErrP labels, the function $Q_h$ learned by maximum likelihood doesn’t have the shape compatible with the state dynamics of the target MDP (environments in experiments). And the generated reward function can make the learning process of RL agent unstable and even divergent.

In order to refine the reward shape and attenuate the gradient variance, we introduce another baseline function $t(s)$ only in terms of state, to incorporate the state transition information. Hence, the Q function becomes $Q_B(s, a) := Q_h(s, a) + t(s)$. It can be proved that $Q_B(\cdot, \cdot)$ and $Q_h(\cdot, \cdot)$ induce the same optimal policy [41]. The baseline function $t(\cdot)$ can be learned by optimizing $t^* = \arg \min_t J_t(t)$, whose objective is defined as [4], where the loss function $l(\cdot)$ is chosen to be $l_1$-norm via empirical evaluations.

As shown in Figure 3 for learning the reward function, in addition to the demonstration $D$, we incorporate another set of demonstrations $D_R$, containing transitions randomly sampled from environment without reward information. The set $D_R$ is to help the function $t(\cdot)$ to efficiently learn the state dynamics, and does not require any human labeling. After reward learning, i.e., learning both $Q_h(\cdot, \cdot)$ and $t(\cdot)$, for any transition tuple $(s, a, s')$, the learned auxiliary reward function can be represented as

$$
r_a(s, a) = Q_h(s, a) + t(s) - \gamma \max_{a'}[Q_h(s', a') + t(s')]
$$

(2)

We then use this $r_a$ to augment the following RL agent. In order to further attenuate the negative influence of wrong ErrP labels, when combining environmental reward $r_e$ and auxiliary reward $r_a$, we propose a coefficient $\beta(\epsilon)$, exponentially decreasing in terms of learning episodes $e$, i.e., $\beta(\epsilon) := ae^{-\epsilon/b}$. Finally, the reward received by RL agent is $r_e(s_t, a_t) + \beta(\epsilon)r_a(s_t, a_t)$. Empirically, the best coefficient function is $\beta(\epsilon) = 3e^{-\epsilon/90}$ in experiments.

**VI. Evaluation**

**A. Baseline results: Naive Approach**

We first validate the feasibility of decoding error-potentials using a 10-fold cross-validation scheme for each game. In this scheme, we split the state-action pairs of a game in 10-folds for training and testing of the ErrP decoder. In Figure 4(a), we show the performance of three games in terms of Area Under Curve (AUC) score, sensitivity and specificity, averaged over 5 subjects. The Maze game has the highest AUC score (0.89 ± 0.05) followed by Catch (0.83 ± 0.08) and Wobble (0.77 ± 0.09).

As discussed in section [IV-B], the full access method is the most preliminary approach to incorporate implicit human feedback (in the form of decoded error-potentials) into the DRL model. It asks the external oracle (human) for the implicit feedback in every training step, reaching the maximum number of possible queries. Hence it has the fastest
B. Evaluation of the Proposed Solution

In this subsection, we evaluate the performance of proposed approaches to practically integrate the implicit human feedback (via EEG) into the DRL algorithms.

1) Zero-shot learning of ErrPs: To evaluate the zero-shot learning capability of error-potentials and the decoding algorithm, we train on the samples collected from the Catch game and test on the Maze game. As Catch is a simple game, we assume the optimal action for each state is already known (providing the labeled examples to train the ErrP decoder). However, the Maze game need to be solved, hence, we do not make any assumptions about the optimality of the actions. In Figure 5(a) we provide the zero-shot learning performance and compare it against the 10-fold Cross-Validation (CV) scheme shown in section VI-A. Further, we present the AUC score of zero-shot learning performance over all training and testing combinations in Figure 5(b). We use the Area Under Curve (AUC) as the performance metric for the decoding of error-potentials. We can see that the ErrPs recorded for Catch game, are able to capture more than 80% of the variability in the ErrPs for Maze game. Averaged over 5 subjects, the decoder performs with an AUC score of 0.693 (±0.034) when trained using False labels. Similarly, Catch and Wobble performs with an average AUC score of 0.790 (±0.018) and 0.680 (±0.018) respectively, when trained on labels obtained through the Maze environment. These experiments validate that the error-potentials can be learned in a zero-shot manner to avoid re-training of the human feedback (via EEG) decoder.
2) Evaluation of Robust Reward Shaping with Human ErrP: For the evaluation of Algorithm 2, stochastic trajectories for the Maze game were generated by random walking in the state space, as long as most part of state space is covered. Before training the RL agent, each human subject provided implicit feedback (via ErrP) as explained in the experimental protocol (section III-A) on every state-action pair along these trajectories. The performance of the proposed approach was evaluated with 10 and 20 initial trajectories, each for 5 subjects. We use the Bayesian DQN as the DRL model.

The acceleration of human feedback is shown in Figure 6(a) for 10 trajectories, where the base model is Bayesian DQN. We can see the significant acceleration in training convergence in Figure 6(a) in terms of the success rate for 5 subjects and compared against the case of No ErrP, i.e., no human feedback. Subject 01 has the highest fidelity for error-potentials, and hence, RL algorithm converges at much faster rate when relies upon the feedback obtained by Subject 01. It is evident from the results that the error-potential decoding performance is sufficient to achieve around 2x improvement in training time (in terms of the number of episodes required). Similarly, Figure 6(b) shows the success rate and convergence curve for training to complete, for 20 trajectories. Comparing Figure 6(a) and (b), we can see that the training converges at much faster rate when the number of initial trajectories are increased. Further, the learning variance is also decreased with more trajectories. The comparison between Figure 6(a) and Figure 6(b) shows that the proposed framework learns faster than No-ErrP case, while outperforming the full access case, even though full access requires significantly larger amount of queries. We also compare the number of ErrP queries for full access and proposed method in Table II, according to the statistics on experiments with 20 trajectories. On an average for 5 subjects, the proposed approach makes 75.56% less queries as compared to the full access. As full access queries for feedback label at every learning step, while the proposed framework queries only on the trajectories given initially, the total number of queries made are significantly reduced.

3) Analysis of the dependence and subjectivity of errors: In this section, we analyze the detection accuracy of error-potentials for the Maze game, to develop insights into the characteristics of error-potential based on the users and provided stimulations. The EEG samples recorded for the Maze experiment can be presented along two independent dimensions, (i) users and (ii) state-action pair of the agent (i.e., stimulation). Within the state-action pairs, if the action is correct, it is called non-errP, otherwise errP.

- **Experiment 1: Subjectivity over correct and incorrect actions.** For each user, we divide the EEG trials into two categories (a) errP and (b) non-errP. For each user and category, we compute the mean and standard deviation of classification accuracy of EEG trials, and present in Table III. We can observe that the per user standard deviations for ErrPs is roughly double the standard deviations for non-ErrPs. The aggregate per user standard deviation across SAPs is 0.28 for ErrPs and 0.14 for non-ErrPs. This difference in per user standard deviations is statistically significant (p<0.001). We also calculate the standard deviations across our user accuracy vectors for both ErrP and non-ErrP case and find that the standard deviations for the per user accuracy vectors are 0.06 and 0.08 respectively.

- **Experiment 2: Subjectivity over users.** In this experiment, for each unique state-action pair, we average the performance of EEG trials of all users. We achieved a mean and standard deviation of 0.75 and 0.13, and 0.75 and 0.07 for errP and non-errP respectively. We use Levene’s test [38] to conclude that the difference in variance between these two population samples is statistically significant (p = 0.023 < 0.05).

- **Experiment 3: Subjectivity over states.** For each unique state in Maze game, we plot the mean and standard deviation of EEG trial performance in Fig. 7. We plot the classifier accuracy for ErrP and non-ErrP SAPs respectively based on the initial state on the maze. We can visualize that the plot corresponding the standard deviation for non-ErrPs is darker (indicating lower standard deviation) compared to the plot corresponding to the deviations for ErrPs. We can also see that within a plot, there is also a gradation in the accuracy (indicated by different shades of green) implying that there is some dissimilarity among erroneous states and hence subjectivity on the user’s part and diminishing the argument that erroneous vs non-

### Table II: Average Number of Queries on Maze Game

| Subject     | 01 | 02 | 03 | 04 | 05 |
|-------------|----|----|----|----|----|
| Full access | 1879.4 | 2072.1 | 2293.7 | 1975.4 | 2130.1 |
| Proposed method | 505.7 | 394.7 | 587.1 | 681.4 | 361.3 |

### Table III: Accuracy and standard deviations per subject

| Subject | ErrP mean | ErrP std dev | non-ErrP mean | non-ErrP std dev |
|---------|-----------|--------------|---------------|------------------|
| S12     | 0.79      | 0.27         | 0.75          | 0.17             |
| S07     | 0.8       | 0.3          | 0.85          | 0.16             |
| S02     | 0.73      | 0.29         | 0.77          | 0.15             |
| S08     | 0.6       | 0.25         | 0.56          | 0.14             |
| S01     | 0.8       | 0.25         | 0.77          | 0.16             |
| S04     | 0.78      | 0.25         | 0.63          | 0.16             |
| S16     | 0.73      | 0.3          | 0.78          | 0.14             |
| S03     | 0.65      | 0.25         | 0.61          | 0.13             |
| S06     | 0.73      | 0.3          | 0.64          | 0.17             |
| S05     | 0.75      | 0.3          | 0.72          | 0.13             |
| S09     | 0.71      | 0.27         | 0.66          | 0.13             |
| S15     | 0.67      | 0.31         | 0.65          | 0.1              |
| Average | 0.73      | 0.28         | 0.70          | 0.14             |
erroneous scenarios are purely binary.

- **Experiment 4: Errors of commission and omission.**

In this experiment, we consider only the erroneous actions and split the EEG trials into two categories, (i) commission errors and (ii) omission errors. Commission error is defined as an agent making an incorrect move to a new cell, while omission error refers to the incorrect action of agent by staying in the same cell grid. The total state-actions pairs for commission and omission are distributed fairly (out of 71 unique state-action pairs, 34 correspond to errors of omission and the remaining 37 correspond to errors of commission). 7 correspond to errors of commission. However we observe that among the state-action pairs which had very high accuracies, state-action pairs corresponding to errors of commission are disproportionately represented. Out of the top 5 state-action pairs that have the highest accuracy, all of them represent errors of commission and out of the top 10 state-action pairs that have the highest accuracy, 9 of them signify errors of commission. This was also indicated in the fact that errors of omission had a mean accuracy of 72% whereas errors of commission had a much higher mean accuracy of 77%. This implies that the error scenarios that are the easiest to detect are likely to be errors of commission. This has certain implications that bolster the hypothesis that certain errors are indeed more "valuable" to a user than others and hence generate a far more noticeable response in the brain.

These 4 experiments collectively lead us to 2 main insights.

(a) Per subject, owing to the differences in variances, there is less variation in the non-ErrP accuracies compared to the ErrP accuracies implying that erroneous scenarios lead to more variation in the classifier accuracy and by extension, in the brain’s response, than non-erroneous scenarios. This further implies that there is a gradation in error detection unlike it being a binary phenomenon which makes certain errors easier to detect and certain others more difficult to detect.

(b) The differences in variations in classifier accuracy between ErrP and non-ErrP SAPs diminishes when we average the accuracies over the SAPs and represent them as a function of users. This implies that the variation in the accuracy of ErrP vs non-ErrP is impacted more by differences in SAPs compared to the differences in users.

4) **Robustness Evaluation:** Because the generation process and decoding of brain signal are stochastic. The robustness to wrong ErrP labels is important when incorporating human feedback (via EEG) into reward shaping method. We are going to show that modeling the human policy as soft Q policy, as we did in (4), can make the learned auxiliary function $r_\theta$ resist to wrong human feedback. In the comparison on robustness, the baseline method, called "simple", is to simply use a bootstrap neural network to generalize the binary ErrP labels across the state space, same as (60). Both simple benchmark and the proposed robust reward shaping are trained on the same set of trajectories and human labels. The neural network in both methods is MLP, having two hidden layers of 64 units. And the number of bootstrap head in "simple" benchmark is set to 5. We evaluate both simple and the proposed methods on subject 02 and subject 07, whose accuracy are 0.71 and 0.78 respectively. The comparison result is shown in Figure 9. We can see that the proposed method performs better in both subjects with different initial trajectories. That is because the proposed method treats the human feedback in a probabilistic way, and the baseline function $t$ can incorporate the state transition information to attenuate the influence of wrong human feedback. Moreover, the comparison of all cases shows that the performance gain of simple benchmark over no-ErrP method is decreased when the error probability of human label increases.

5) **Ablation Study:** In this section, we conduct ablation study on the proposed robust reward shaping with human feedback. We first specifically evaluate the effect of baseline function $t$ learned from (4). Because the human feedback labels in the initial trajectories can not cover the whole state space and some labels are wrong, the learned Q function of human $Q_h(\cdot, \cdot)$ may not be compatible with the state dynamics of the environment. Thus we introduce a baseline function only in terms of state to smoothen the learned Q function. Here the ablation evaluation on baseline function is still on subject 02 and 07, same as the section above. The baseline is the proposed reward shaping method without $t$ in (2), where
the auxiliary reward function is only the Bellman difference between adjacent states. The comparison result is shown in Figure 9. We can see that the baseline function can improve the convergence speed in all cases, and it can even do better than simple method in some cases, showing the importance of the baseline function here.

In addition, we also conduct the ablation study on the combining coefficient $\beta(\cdot)$. The benchmark method is to directly sum auxiliary reward $r_a$ and environmental reward $r_e$ together. The coefficient in the proposed method is set to $\beta(\epsilon) = 3e^{-\epsilon/80}$. Comparison result is shown in Figure 9. We can see this exponentially decreasing coefficient can stabilize the training process significantly, and hence improve the convergence speed.

VII. CONCLUSIONS AND FUTURE WORK

In this work, we investigate an interesting paradigm to obtain and integrate the implicit human feedback with RL algorithms. We first demonstrate the feasibility of obtaining implicit human feedback by capturing error-potentials of a human observer watching an agent learning to play several different visual-based games, and then decoding the signals appropriately and using them as an auxiliary reward function to help an RL agent. Then we argue that the definition of ErrPs can be learned in a zero-shot manner across different environments, eliminating the need of re-training over new and unseen environments. We validate the acceleration in learning of games through augmenting the RL agent by ErrP feedback using a naive approach, i.e., full access method. We then propose a novel RL framework, improving the label efficiency and reducing human cognitive load. We experimentally show that the proposed RL framework can accelerate the training of RL agent by $2.25x$, while reducing the number of queries required by $75.56\%$.

Scope and Future work: The scope of our work in limited to the visual-based RL problems with discrete state and action spaces. action. Moreover, the demonstration of the zero-shot learning of error-potentials is limited across the environments presented in the paper. We have considered discrete grid-based reasonably complex navigational games in our work. Further studies have to be done to explore if such an approach could be extended to Atari and Robotic environments environments with very large state-space and continuous action-space. We plan to test our framework over robotic environments, and evaluate the zero-shot learning capabilities of error-potentials between virtual and physical worlds.

REFERENCES

[1] Riku Arakawa, Susuke Kobayashi, Yuya Unno, Yuta Tsuboi, and Shinichi Maeda. Dqn-tamer: Human-in-the-loop reinforcement learning with intractable feedback. arXiv preprint arXiv:1810.11748, 2018.

[2] Kamyar Azizzadenesheli, Emma Brunskill, and Animashree Anandkumar. Efficient exploration through bayesian deep q-networks. In 2018 Information Theory and Applications Workshop (ITA), pages 1–9. IEEE, 2018.

[3] 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.

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

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

[6] Shilomo Bentin, Gregory McCarthy, and Charles C Wood. Event-related potentials, lexical decision and semantic priming. Electroencephalography and clinical Neurophysiology, 60(4):343–355, 1985.

[7] Benjamin Blankertz, Guido Dornhege, Christin Schafer, Roman Krepki, Jens Kohlmorgen, K-R Muller, Volker Kunzmann, Florian Losch, and Gabriel Curio. Boosting bit rates and error detection for the classification of fast-paced motor commands based on single-trial eeg analysis. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 11(2):127–131, 2003.

[8] Daniel S Brown, Wonjoon Goo, Prabhat Nagarajan, and Scott Nieckum. Extrapolating beyond suboptimal demonstrations via inverse reinforcement learning from observations. arXiv preprint arXiv:1904.06357, 2019.

[9] Cameron B Browne, Edward Powley, Daniel Whitehouse, Simon M Lucas, Peter I Cowling, Philipp Rohlfshagen, Stephen Tavener, Diego Perez, Spyridon Samothrakis, and Simon Colton. A survey of monte carlo tree search methods. IEEE Transactions on Computational Intelligence and AI in games, 4(1):1–43, 2012.

[10] Tim Brys, Anna Harutyunyan, Halit Bener Suay, Sonia Chernova, Matthew E Taylor, and Ann Nowé. Reinforcement learning from demonstration through shaping. In Twenty-Fourth International Joint Conference on Artificial Intelligence, 2015.

[11] 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.

[12] Hyeong Soo Chang. Reinforcement learning with supervision by combining multiple learnings and expert advices. In 2006 American Control Conference, pages 6–pp. IEEE, 2006.

[13] LI Charles and RS Christian. A social reinforcement learning agent. In Proceedings of the fifth international conference on Autonomous agents, 2001.

[14] Ricardo Chavarriaga and José del R Millán. Learning from eeg error-related potentials in noninvasive brain-computer interfaces. IEEE transactions on neural systems and rehabilitation engineering, 18(4):381–388, 2010.

[15] Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. In Advances in Neural Information Processing Systems, pages 4299–4307, 2017.

[16] Marco Congedo, Alexandre Barachant, and Anton Andreev. A new generation of brain-computer interface based on riemannian geometry. arXiv preprint arXiv:1310.8115, 2013.
advice for stochastic policy learning. In 2019 IEEE 58th Conference on Decision and Control (CDC), pages 1842–1849. IEEE, 2019.

[62] Ruohan Zhang, Faraz Torabi, Lin Guan, Dana H Ballard, and Peter Stone. Leveraging human guidance for deep reinforcement learning tasks. In Proceedings of the 28th International Joint Conference on Artificial Intelligence, pages 6339–6346. AAAI Press, 2019.

[63] Brian D Ziebart. Modeling purposeful adaptive behavior with the principle of maximum causal entropy. PhD thesis, figshare, 2010.