Harnessing the wisdom of an unreliable crowd for autonomous decision making

Tamlin Love, Ritesh Ajoodha and Benjamin Rosman
School of Computer Science and Applied Mathematics
University of the Witwatersrand
Johannesburg, South Africa
1438243@students.wits.com, ritesh.ajoodha@wits.ac.za and benjamin.rosman1@wits.ac.za

Abstract

In Reinforcement Learning there is often a need for greater sample efficiency when learning an optimal policy, whether due to the complexity of the problem or the difficulty in obtaining data. One approach to tackling this problem is to introduce external information to the agent in the form of domain expert advice. Indeed, it has been shown that giving an agent advice in the form of state-action pairs during learning can greatly improve the rate at which the agent converges to an optimal policy. These approaches typically assume a single, infallible expert. However, it may be desirable to collect advice from multiple experts to further improve sample efficiency. This may introduce the problem of multiple experts offering conflicting advice. In general, experts (especially humans) can give incorrect advice. The problem of incorporating advice from multiple, potentially unreliable experts is considered an open problem in the field of Assisted Reinforcement Learning.

Contextual bandits are an important class of problems with a broad range of applications such as in medicine, finance and recommendation systems. To address the problem of learning with expert advice from multiple, unreliable experts, we present CLUE (Cautiously Learning with Unreliable Experts), a framework which allows any contextual bandit algorithm to benefit from incorporating expert advice into its decision making. It does so by modelling the unreliability of each expert, and using this model to pool advice together to determine the probability of each action being optimal.

We perform a number of experiments with simulated experts over randomly generated environments. Our results show that CLUE benefits from improved sample efficiency when advised by reliable experts, but is robust to the presence of unreliable experts, and is able to benefit from multiple experts. This research provides an approach to incorporating the advice of humans of varying levels of expertise in the learning process.

Keywords: Assisted Reinforcement Learning, Interactive Reinforcement Learning, Agent Teaching, Contextual Bandits, Expert Advice

1 Introduction

Sample efficiency is often an issue of great concern in Reinforcement Learning (RL). This is often due to the complexity of a problem, which may consist of a large number of states and actions. It may also be due to the difficulty in acquiring data. The Assisted Reinforcement Learning (ARL) framework seeks to improve sample efficiency by incorporating external information in the learning process [1]. For example, a domain expert could advise an expert on which action it should perform in a given state. This advice could be given throughout the learning process, in response to the agent's behaviour; an approach termed Interactive Reinforcement Learning (IRL). The types of advice offered by an expert may differ between approaches. In this work, we consider policy-shaping advice in the form of a single action offered for a state, as it is often easier to elicit from a domain expert and is more robust to infrequent and inconsistent feedback [7].

It is often assumed that advice is coming from a single, infallible expert. These assumptions are not always practical, however. Experts, especially humans, can give suboptimal advice due to misunderstandings (e.g. advice is given for the wrong state), erroneous domain knowledge or on purpose with the intent of sabotaging the agent. Restricting advice to a single expert also limits the amount of information the agent can receive. Multiple experts can potentially have different perspectives or areas of expertise, and the contradictions and consensus between experts may reveal additional information.

Our aim is to build on the IRL approach in the specific context of contextual bandits (CBs), which are in essence RL problems whose episodes are a single timestep in length, where we attempt to tackle the open problem of incorporating the advice of multiple, potentially unreliable experts in policy-learning. Our main contribution in this regard is CLUE (Cautiously Learning with Unreliable Experts). This framework uses a model of the reliability of the experts to augment any CB action-selection algorithm with the ability to incorporate advice from multiple experts.

Related Work: There have been some approaches to tackling the problems of multiple experts and of unreliable experts, though the experts in these approaches often provide other forms of advice than the action advice we consider. Gimelfarb, Sanner, and Lee [6] combine reward-shaping advice from multiple experts as a weighted sum of potential functions, where the weights are updated as the agent learns. The decision-making rule in Section 2.2 is directly inspired by this Bayesian combination of advice. Griffith et al. [7] account for incorrect advice by modelling the probability of an expert giving correct advice with a single, static parameter $C \in (0, 1)$. Such a model of reliability is expanded on in Section 2.1. Other approaches include the adversarial bandit algorithms EXP4 and EXP4.P, whose experts provide advice in the form of probability vectors [10], and the probabilistic policy reuse algorithm, in which experts' entire policies are transferred and weighted against the agent's own policy using the reward [5].

2 Methodology

In this section we describe the CLUE framework and the problem setting, which is composed of three actors: an environment, an agent and a panel E of one or more experts. The environment is a standard Contextual Bandit (CB) environment. For each trial t, it samples state s_t, accepts action a_t from the agent and returns reward r_t. At the end of the trial, each expert e in panel E receives $\langle s_t, a_t, r_t \rangle$ and may independently offer their own advice, $(s_t, a_t^{(e)})$ on what action the agent should have taken this trial. How and when an expert decides to offer advice may differ between experts. In our formulation of the problem, we assume each expert to be a domain expert with consistent reliability across the breadth of the problem. It is worth noting here that, although we choose to have the expert give advice at the end of the trial in this work, this can occur instead at the start of a trial without requiring any change to the CLUE algorithm.

The agent is composed of three components, the first of which is a learning algorithm, which uses the information $\langle s_t, a_t, r_t \rangle$ to learn a policy, such as the action-value update rule $Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha(r_t - Q(s_t, a_t))$, where $Q(s, a)$ is the action-value function and $\alpha \in [0, 1]$ is a step size parameter.

The second component, and one of the contributions of this work, is a model of the reliability of each expert (see Section 2.1). This model is necessary for learning which pieces of advice are to be followed and which are to be ignored. When an expert utters a piece of advice at the end of a trial, the agent uses its own information about the environment (such as an action-value function) to evaluate the advice and update the model. The third component, and another contribution of this work, is a decision making process which uses the information learned by the learning algorithm and the models of each expert to select an action for a state while exploring, given any advice it has previously received for that state (see Section 2.2).

2.1 Modelling Experts

Intuitively, we can think of an expert's reliability as the probability of them giving optimal advice [7]. We can therefore model it as $\rho \in [0, 1]$, where $\rho = 1$ is an infallible expert and $\rho = 0$ is an expert that always gives suboptimal advice. Given the range of values, a natural choice is to model the probability distribution $P(\rho)$ as a Beta distribution $Beta_\rho[\alpha, \beta]$,

where α, $\beta > 0$ can be thought of as counts recording the number of times the expert gave correct and incorrect advice respectively.

At the end of trial t, the agent must update this distribution for each expert that gave advice for s_t sometime in the past. To do this, the agent can evaluate the advice as either optimal or suboptimal, given its own information. In this work, we set $x_t = 1$ if $Q(s_t, a_t^{(e)}) = max_a Q(s, a)$, and $x_t = 0$ otherwise, where $a_t^{(e)}$ denotes the advice received from expert e. Let $x_t = 1$ denote an optimal evaluation, and $x_t = 0$ denote a suboptimal evaluation. In order to allow for inconsistent experts (e.g. an expert whose performance degrades over time), we update an estimate χ of the expected value $\mathbb{E}[\rho]$ using a recency-weighted moving average with weight parameter $\delta \in [0, 1]$,

$$\chi_{t+1} = (1 - \delta)\chi_t + \delta x_t, \tag{1}$$

where $\chi_0 = \mathbb{E}_0[\rho] = \frac{\alpha_0}{\alpha_0 + \beta_0}$, with prior counts α_0 and β_0.

2.2 Making Decisions

Suppose that, at the start of trial t, the agent observes state s_t and recalls any advice that some subset $E_t \subseteq E$ of experts offered for state s_t in trials $[0, ..., t - 1]$. The agent must now use its model of the reliability of each expert to decide which advice (if any) to follow. In order to allow the agent to surpass the performance of the experts advising it, we only allow the agent to consider expert advice when exploring. Determining when the agent is exploring depends on the underlying action-selection algorithm. As CLUE can augment any CB action-selection algorithm, we consider the Epsilon-Greedy, Adaptive Greedy, Explore-then-Exploit (ETE) and Upper Confidence Bound (UCB) algorithms, representing several families of CB algorithms [4]. For the first three algorithms, whether or not the agent is exploring is explicitly determined by the algorithms' parameters. For UCB, the agent can be said to be exploring if $argmax_a Q(s, a) \neq argmax_a(Q(s, a) + c\sqrt{\frac{2ln(t)}{N(s,a)}})$, where $N(s, a)$ counts the number of times action a has been selected for state s and c is a parameter that balances exploration and exploitation.

If exploring, the agent must choose between the action suggested by the underlying action-selection algorithm or between following advice it has received for s, in which case it must choose which advice to follow. If $E_t = \varnothing$, no advice has been offered, such as may happen at the beginning of the learning process, and the agent must act without advice according to its underlying action-selection algorithm. If $|E_t| \geq 1$, at least one expert has offered advice. In order to take advantage of the information provided by consensus and contradiction among experts, we employ a Bayesian method of pooling advice, inspired by similar approaches in potential-based reward shaping [6] and in crowd-sourced data labelling [2]. Let a^* denote the optimal action for state s_t and $v_t^{(e)}$ denote the advice utterance given by expert e for s_t, with V_t denoting the set $\{v_t^{(e)} | e \in E_t\}$. Our aim, therefore, is to calculate $P(a_j = a^* | V_t)$ for each $a_j \in A$. To do this, we employ Bayes' rule, coupled with the assumptions that each expert gives advice independently of every other expert and that each action has a uniform prior probability of being optimal,

$$P(a_j = a^* | V_t) = \frac{\prod_{e \in E_t} P(v_t^{(e)} | a_j = a^*)}{\sum_{k=0}^{|A|} \prod_{e \in E_t} P(v_t^{(e)} | a_k = a^*)}. \tag{2}$$

Note that, if for a particular domain one can reasonably assume a non-uniform prior distribution of $P(a = a^*)$, this distribution can be incorporated into Equation 2 without fundamentally changing this decision-making process.

All that remains is to calculate $P(v_t^{(e)} | a_j = a^*)$. Recalling that the probability of the advice being correct is estimated by $\chi^{(e)} \approx \mathbb{E}[\rho^{(e)}]$ and assuming that, if the advice is incorrect, the expert is equally likely to advise any suboptimal action, then $P(v_t^{(e)} | a_k = a^*) = \chi^{(e)}$ if the expert advised a_k and $P(v_t^{(e)} | a_k = a^*) = \frac{1 - \chi^{(e)}}{|A| - 1}$ otherwise. Substituting this into Equation 2, we can calculate the probability of each action in A being optimal, and can set $a_{best} = \arg\max_a P(a = a^* | V_t)$. In an approach reminiscent of both Epsilon Greedy and probabilistic policy reuse [5], the agent selects action a_{best} with probability $P(a_{best} = a^* | V_t)$, and otherwise acts as if $E_t = \varnothing$. This allows for a trade-off between following advice and exploring as normal, where the former is more likely if the agent is confident that a_{best} is optimal.

In the above formulations, we have assumed that the estimated $\chi^{(e)}$ accurately represents the underlying reliability of the expert e. Early in the learning process however, this will not be the case. Erring on the side of caution, we can compensate for the over-estimation of the reliability of particularly bad experts by introducing a threshold parameter $T \in [0, 1]$, such that if $P(a_{best} = a^* | V_t) < T$, the agent acts without advice. This approach ensures that the agent will only follow advice if it is sufficiently confident that the advice is correct. 379

3 Experiments and Results

In this section, we present a number of experiments to demonstrate that CLUE benefits from improved sample efficiency when being advised by a reliable expert but is robust to the presence of suboptimal advice. Furthermore, we aim to show that CLUE can benefit from advice from a panel of multiple experts with varying degrees of reliability.

These experiments are performed using a number of randomly generated Contextual Bandit environments, specified by Influence Diagrams with a great diversity of randomly generated graph structures and parameters [8]. Experts are simulated, and are limited in how much advice they can give, thus only giving advice if the agent is underperforming within some degree of tolerance [9]. In order to simulate reliability, each expert is controlled by a *true reliability parameter* ρ_{true}. When offering advice, the expert will advise the optimal action a^* with probability ρ_{true}, or else will randomly advise any other action. Thus an expert with $\rho_{true} = 1$ is reliable, while one with $\rho_{true} = 0$ never advises the optimal action.

3.1 Panel Comparisons

In this set of experiments, we compare the reward obtained in $80,000$ trials, averaged across 100 random environments ($|S| = 1024$, $|A| = 8$). LOWESS smoothing is employed for legibility [3], with the standard deviation represented by the shaded areas. We compare the performance of each agent with three panels of experts. The first, a *Single Reliable Expert*, consists of one expert that always gives correct advice ($\rho_{true} = 1$). The second, a *Single Unreliable Expert*, consists of one expert that always gives incorrect advice ($\rho_{true} = 0$). The third, a *Varied Panel*, consists of seven experts with varying degrees of unreliability ($P_{true} = \{0, 0.1, 0.25, 0.5, 0.75, 0.9, 1\}$). Agents tested include an unassisted *Baseline Agent*, a *Naïve Advice Follower* (NAF), which follows any advice it has received for a state (choosing randomly between contradicting advice) otherwise acting as the Baseline Agent, and CLUE ($\alpha_0 = 1 = \beta_0$, $T = \frac{2}{|A|}$, $\delta = 0.5$), which augments the Baseline Agent. The four tested baselines are Epsilon Greedy (ϵ decays from 1 to 0 across 80% of trials), Adaptive Greedy (z decays from 1 to -1 across 80% of trials), Explore-then-Exploit (ETE, threshold of $20,000$) and Upper Confidence Bound (UCB, $c = 0.25$), all of which employ the Q update rule in Section 2 ($Q_0 = 0, \alpha = \frac{1}{k(s,a)}$). Results are shown in Figure 1.

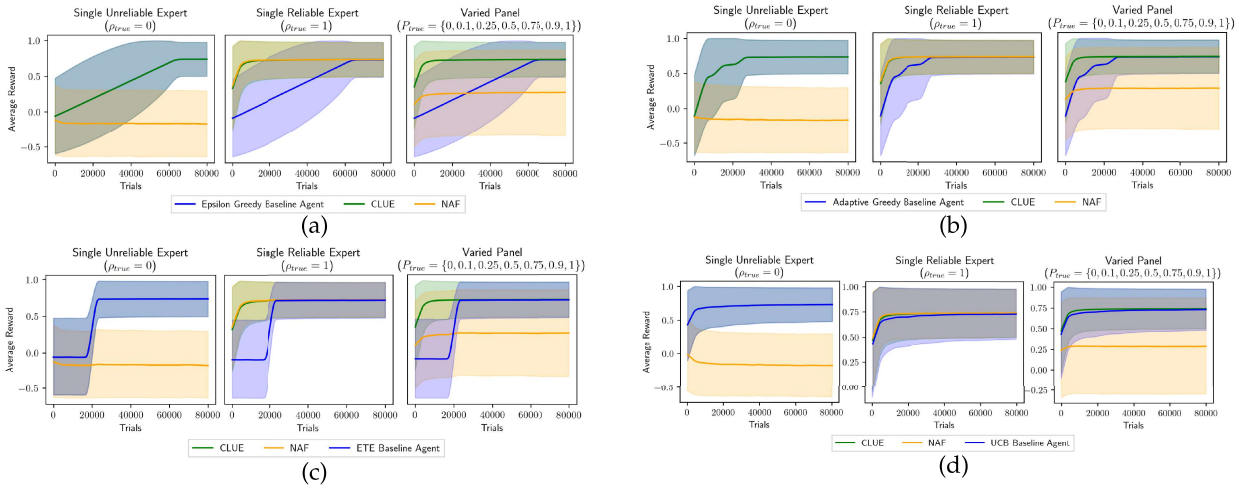

Figure 1: Panel comparisons for (a) Epsilon Greedy, (b) Adaptive Greedy, (c) ETE and (d) UCB. Note that CLUE and the Baseline are nearly identical for $\rho_{true} = 0$.

For $\rho_{true} = 1$, both CLUE and NAF converge faster than all Baselines as they quickly benefit from the optimal advice provided by the reliable expert. A demonstration of the robustness of CLUE comes when $\rho_{true} = 0$. In this scenario, NAF exclusively follows sub-optimal advice and thus is unable to converge to the optimal policy. CLUE on the other hand is able to identify that the expert is unreliable and defaults to its underlying action-selection algorithm, performing identically to the Baselines. For the varied panel, the performance of NAF lies somewhere between the two single expert cases, as it receives a mix of advice including optimal and suboptimal actions, and cannot discern which advice is advantageous to follow. CLUE is able to differentiate between reliable and unreliable experts and benefits from the former despite the presence of the latter. In all cases, CLUE either converges faster than the Baseline when good advice is available, or otherwise converges at the same rate as the Baseline.

3.2 Reliability Estimates

To investigate the results obtained in Section 3.1, we plot the value of $\chi^{(e)}$ over time for the same panels of experts and with an Epsilon Greedy Baseline. Results are plotted in Figure 2.

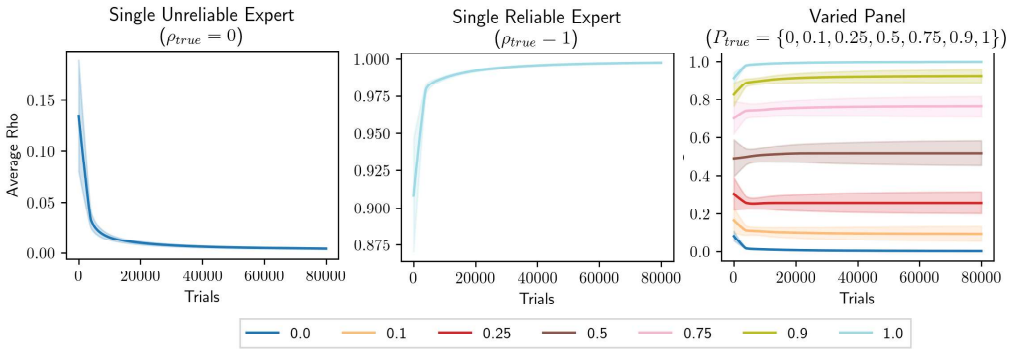

Figure 2: A comparison of $\chi^{(e)}$ for each panel with an Epsilon Greedy Baseline. The Legend denotes the value of ρ_{true}

For the single expert cases, the value of χ converges towards the correct value of ρ_{true} (1 and 0 respectively), with the final estimates being $\chi = 0.995$ for the single reliable expert and $\chi = 0.005$ for the single unreliable expert. For the varied panel, each expert is correctly ranked according to their reliability and the value of $\chi^{(e)}$ for each expert e correctly converges towards the true value of $\rho^{(e)}_{true}$, even faster than the single expert cases. This accuracy in the estimates of reliability explains the performance obtained in Section 3.1. As is to be expected, the variance in the final estimate is larger for experts that randomly choose between suboptimal and optimal advice ($\rho_{true} = 0.5$) than for experts that more consistently offer one or the other.

4 Conclusion

Our results show that CLUE is able to incorporate expert advice in such a way that it benefits from improved sample efficiency when advised by a reliable expert, but is robust to advice from unreliable experts. Furthermore, by modelling the reliability of the experts, CLUE is able to incorporate advice from multiple experts, even when these experts contradict each other. When multiple experts are present, CLUE is able to rank them by their reliability and exploit the information revealed by consensus and contradiction between experts. This work may allow for easier integration of external information in the learning process, ultimately contributing towards tackling more complex problems with greater sample efficiency.

References

[1] Adam Bignold et al. "A conceptual framework for externally-influenced agents: an assisted reinforcement learning review". In: *Journal of Ambient Intelligence and Humanized Computing* (2021), pp. 1–24.

[2] Pierce Burke and Richard Klein. "Confident in the Crowd: Bayesian Inference to Improve Data Labelling in Crowdsourcing". In: *2020 International SAUPEC/RobMech/PRASA Conference*. IEEE. 2020, pp. 1–6.

[3] William S Cleveland. "LOWESS: A program for smoothing scatterplots by robust locally weighted regression". In: *American Statistician* 35.1 (1981), p. 54.

[4] David Cortes. "Adapting multi-armed bandits policies to contextual bandits scenarios". In: *arXiv preprint arXiv:1811.04383* (2018).

[5] Fernando Fernández and Manuela Veloso. "Probabilistic policy reuse in a reinforcement learning agent". In: *Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems*. 2006, pp. 720–727.

[6] Michael Gimelfarb, Scott Sanner, and Chi-Guhn Lee. "Reinforcement learning with multiple experts: A Bayesian model combination approach". In: *Advances in Neural Information Processing Systems* 31 (2018), pp. 9528–9538.

[7] Shane Griffith et al. "Policy shaping: Integrating human feedback with reinforcement learning". In: Georgia Institute of Technology. 2013.

[8] Ronald A Howard and James E Matheson. "Influence diagrams". In: *Decision Analysis* 2.3 (2005), pp. 127–143.

[9] Craig Innes and Alex Lascarides. "Learning Structured Decision Problems with Unawareness". In: *International Conference on Machine Learning*. 2019, pp. 2941–2950.

[10] Li Zhou. "A survey on contextual multi-armed bandits". In: *arXiv preprint arXiv:1508.03326* (2015).