








Modeling Human Actions in the Cart-Pole Game Using Cognitive and Deep Reinforcement Learning Approach

Aadhar Gupta^(✉) , Mahavir Dabas^(✉) , Shashank Uttrani^(✉) ,
Sakshi Sharma^(✉) , and Varun Dutt^(✉) 

Applied Cognitive Science Lab, Indian Institute of Technology Mandi, Kamand
175005, Himachal Pradesh, India
{aadhar.innovate,mahavirdabas18,shashankuttrani,sakshi28720}@gmail.com,
varun@iitmandi.ac.in

Abstract. Designing optimal controllers still poses a challenge for modern Artificial Intelligence systems. Prior research has explored reinforcement learning (RL) algorithms for benchmarking the cart-pole control problem. However, there is still a lack of investigation of cognitive decision-making models and their ensemble with the RL techniques in the context of such dynamical control tasks. The primary objective of this paper is to implement a Deep Q-Network (DQN), Instance-based Learning (IBL), and an ensemble model of DQN and IBL for the cart-pole environment and compare these models' ability to match human choices. Forty-two human participants were recruited to play the cart-pole game for ten training trials followed by a test trial, and the human experience information containing the situations, decisions taken, and the corresponding reward earned was recorded. The human experiences collected from the game-play were used to initialize the memory (buffer) for both the algorithms, DQN and IBL, rather than following the approach of learning from scratch through environmental interaction. The results indicated that the IBL algorithm initialized with human experience could be proposed as an alternative to the Q-learning initialized with human experience. It was also observed that the ensemble model could account for the human choices more accurately compared to the Q-learning and IBL models.

Keywords: Instance-Based Learning · Cognitive Modeling · Reinforcement Learning · Q-Learning · DQN · cart-pole · Ensemble

1 Introduction

Reinforcement Learning (RL) is a paradigm of machine learning where the agent learns by indirect supervision signal in the form of rewards [17]. Contrary to supervised learning, RL is used when the target outputs are unknown, so the

agent needs to interact with the environment to gather information [17]. The agent heads towards optimal behavior by exploring the rewards associated with various actions under various situations and exploiting the hence-gained knowledge of the goodness of actions to maximize the cumulative reward for an entire sequence of actions [17]. With the advent of Deep RL (DRL) [10], it has become possible to apply RL to complex problems, earlier considered to be intractable [2]. The recent success of RL in tasks like playing Atari games at a superhuman level [11] has demonstrated the capability and robustness of RL algorithms.

DRL suffers from certain shortcomings, such as reward shaping, sample inefficiency, and local optima [5]. Learning from human behavior offers an alternative to achieving intelligent behavior. Imitation Learning (IL) [14] is a branch of AI where the agent tries to mimic human behavior. Similarly, Cognitive Science is another branch of Artificial Intelligence (AI) that uses human behavior and aims at creating techniques as robust, insightful, and adaptive as human intelligence [7]. Prior research has contributed to more than a hundred cognitive architectures, including production rule-based, psychology-based, and a combination of neural networks with cognitive psychology, to mention a few [7]. Adaptive Control of Thought-Rational (ACT-R) [1] is a psychologically motivated cognitive model that combines AI, cognitive psychology, and some components of neurobiology. Many researchers have extended upon the principles of ACT-R yielding architectures avoiding the high complexity yet retaining the efficiency, such as Instance-Based Learning (IBL) [6].

The cart-pole problem [3, 9] provides a simple and cost-effective platform to test AI algorithms for control. It consists of a pole attached to a cart like an inverted pendulum, and the player needs to balance the pole by moving the cart. Prior research has investigated a wide range of techniques for the cart-pole problem [4, 8, 12, 13, 15, 16, 18, 19], with a major focus on RL and DRL [10, 11]. However, little is known about the capability of RL techniques to account for human choices in these games. The learning in RL techniques examined in the literature so far, with regard to control problems like cart-pole, is purely mathematical. It doesn't incorporate human intuition. To address this literature gap, we have made a two-fold attempt to give a human touch to RL: by building it over human behavior data and by developing a cognitive model to work in an ensemble with RL.

The upcoming sections include the background on the cart-pole problem, followed by the detailed methodology of this study. Next, the results are presented, followed by the conclusion of our findings with a brief analysis.

2 Background

The cart-pole problem seems to be introduced in [9] and popularized by [3]. Since then, literature has witnessed a plethora of experimentation on this problem, mostly focused on the RL techniques [4, 8, 12, 13, 15, 16, 18, 19]. The algorithms of Q-Learning [20] and deep Q-learning [10] have been thoroughly investigated, along with a few others. [12] examined Q-Learning and SARSA for playing cart-pole and found that both performed quite well. [13] examined a variety of algorithms, including Policy Gradient (PG), Temporal difference (TD), and DQN,

and found TD to perform the best while PG displayed better stabilization and faster convergence than Q-Learning. While [18] examined Deep Q-Learning over cart-pole, [16] examined the Baseline PG and the Reinforce PG and found Reinforce PG to outperform in cumulative reward while Baseline PG outperformed in episode speed. [19] proposed novel variants of DQN and other advanced algorithms and found the rewards to increase with a reduced need for training. [4, 8] examined various advanced algorithms, including DQN, and found PER with DQN to perform remarkably well. [15] investigated the difference in the performance of Q-learning and DQN over cart-pole but didn't observe any significant difference. However, the Q-learning algorithm was found to train the agent significantly faster than DQN. However, the aforementioned techniques investigated on cart-pole don't take into consideration the human aspect of decision making. Moreover, prior research has also lacked an investigation of ensemble techniques that combine the RL and cognitive paradigms.

In this study, we began by developing a virtual cart-pole game, followed by the implementation of a DRL and a cognitive algorithm for the agent to balance the pole. Among the DRL techniques, we developed a deep Q-learning network (DQN), and the cognitive model was based on IBL. The goal of the agent trained on DQN, IBL, and the ensemble of these two algorithms was to keep balancing the pole by moving the cart left or right. The study began with collecting the game-play data of human participants, with multiple trials in the training phase and a single trial in the testing phase. Next, the DQN and IBL were applied to the agent to play the cart-pole game in the same way human players did. Furthermore, an ensemble model was developed to combine the IBL cognitive architecture and DQN. Finally, the results for IBL, DQN, and their ensemble were observed and compared.

3 Methodology

3.1 Game Design

A cart-pole game was developed. The task was to balance the pole on the cart for as long as possible. The cart-pole system dynamics were completely governed by pre-defined equations [3, 9] for the horizontal motion of the cart and the angular displacement of the pole. The cart and the pole were assigned a virtual weight of 1 kg each, and a left or right action exerted a force of 10N on the cart. Hence a keypress in either direction caused the cart to accelerate, either increasing the speed in that direction or reducing the speed if the cart was moving in the reverse direction. On initialization of the game, the cart appeared vertically above the horizontal center of the platform. The initial angle of the pole with the vertical was obtained randomly between 0.05 rad (approximately 2.86°) to the left and 0.05 rad to the right side of the vertical. There were two terminating conditions for the game: the angle of the pole with the vertical axis exceeding a threshold of 30° and the cart falling off the platform. A reward of 0.1 and -5 was given for non-terminating and terminating actions, respectively. The game-play was divided into two phases: the training phase, with ten trials per player, and the

testing phase, with a single trial per player. The situation of the cart-pole was defined by four values: cart position, cart velocity, pole angle to the vertical axis, and pole angular velocity. There were two possible actions for a participant to be taken in the game: move left, and move right.

3.2 Participants

42 participants were enlisted from the Indian Institute of Technology, Mandi, to collect human data after approval from the ethics committee. There were 76.18% males and 23.82% females (mean = 25, sd = 3). 93% of the participants belonged to STEM, and the rest belonged to the humanities.

3.3 Procedure

The experiment began with instructing the participants on the game’s rules, along with a collection of the demographic details. No time limitation was set for either of the phases. The actions taken and the corresponding situation vector were recorded. The recorded data was fed into the DQN, IBL, and ensemble model of DQN and IBL (more details ahead), which were then made to act in the environment, and the observations were collected.

IBL Model

Conceptual Details. IBL [9] works similarly to how humans make judgments by gathering and refining memory experiences. The past experiences are stored as situation-decision-utility (SDU) tuples called instances. Given a situation, the most similar situations are retrieved and are used to compute the goodness score for each decision, called blended value (BV). The decision with maximum BV is executed. IBL uses the formulations of the Activation, Probability of retrieval (PR), and BV, given as:

$$A_{i,t} = \sigma \ln\left(\frac{\gamma_{i,t}}{1 - \gamma_{i,t}}\right) + \ln\left(\sum_{t_p=1}^{t-1} (t - t_p)^{-d}\right) + \mu(S) \quad (1)$$

where d , σ and γ represent the parameter for memory decay, cognitive noise, and a random draw from a uniform probability distribution, respectively. t_p and S for instance i , represent the timestamp and the similarity measure with the current test situation, respectively, while μ is the scaling factor.

$$P_{i,t} = \frac{e^{A_{i,t}/\tau}}{\sum_j e^{A_{j,t}/\tau}} \quad (2)$$

where τ represents the random noise and $A_{i,t}$ represents the activation of the instance i .

$$V_j = \sum_{i=1}^n p_i x_i \quad (3)$$

where j is the concerned action, while x_i and p_i are the utility and PR of the instance i .

Implementation. The IBL agent’s memory was initialized with the SDU instances of a human participant’s ten training trials of game-play. All the instances were timestamped with 0; hence, base activation was not used. The IBL agent was evaluated on situations from the participant’s test session data. The memory instances with cosine similarity greater than 0.85 for the current situation were shortlisted and used to compute the activation and PR, and BV. The model’s performance was measured by comparing predicted decisions with the human participants. Considering the class imbalance, the F1 score was opted to evaluate the model’s ability to mimic human decision making. Each of the 42 participant’s data was used to initialize an IBL model. Hence there were 42 distinct instances of the IBL model. The F1 score over all the model instances was averaged to give a generalized metric for IBL’s human behavior-mimicking ability.

Hyper-parameters. In the IBL model, the hyper-parameters used were the cognitive noise and the similarity threshold, as mentioned in Table 1a. Cognitive noise (CN) is added to capture the variability in decisions from one agent to another, while the similarity threshold controls the memory instances that are allowed to contribute to the decision-making.

DQN Model

Conceptual Details. Q-learning is a model-free RL algorithm [20] that uses a trial-and-error approach to learn via environmental interaction [20]. The algorithm aims to determine the State-Action values (Q-Value) and store it in a table called the Q-table [20]. The Q-values are updated using the Bellman equation, given as:

$$Q(s_t, a_t) = (1 - \alpha)Q(s_t, a_t) + \alpha(r_t + \gamma \max_a [Q(s_{t+1}, a)]) \quad (4)$$

where $Q(s_t, a_t)$ represents the Q-Value for state s_t and action a_t . s_t and s_{t+1} stand for the current and the next state, respectively, and r_t represents the reward on the transition from the state s_t to the state s_{t+1} on taking action a_t , α represents the learning rate that controls the amount of updation in the Q-values and γ represents the discounting factor.

Implementation. The experience replay buffer [20] of DQN was initialized with the quadruplets of ‘State, Decision, Feedback, and Next state’ to enable it to learn from human behavior rather than environmental interactions. The model

was trained for a maximum of 10 epochs with early stopping. The model predicted Q values corresponding to the actions left and right. The cosine distance between the predicted Q-values and the target Q-values, along with the validation loss, was computed. A distinct model instance was trained on each participant’s data. The overall performance of the DQN model was computed by averaging the F1 score between predicted and actual human decisions for all the model instances.

Network Architecture. The neural network architecture (16-32-2) comprised two fully connected hidden layers with 16 and 32 units and the output layer with two units corresponding to the two actions. Rectified linear activation followed by Dropout with a rate of 0.1 was used for both hidden layers.

Hyper-parameters. Hyper-parameters can play a significant role in the learning of a neural network. The hyper-parameters used for DQN in this study are presented in Table 1b.

Ensemble Model

Conceptual Details. The Ensemble model was obtained by performing weighted addition of the cognitive model’s BVs and the DQN model’s State-Actions Values (SAV) for the corresponding decisions. The BV of the IBL model represents the experienced utility (experienced reward) for the current action, and the State-Action value predicted by the DQN approximates the cumulative future rewards. With the aim of attaining more informed decision making, these two values were brought together. The values were normalized to bring the BV and the SAV to the same scale. A weight variable was used to determine the contribution of each approach in the ensemble value corresponding to each decision alternative. The decision against the higher ensemble value was chosen. For the weight value, ‘x’ multiplied by the IBL BV V_j , a weight of ‘1-x’ was multiplied with the DQN SAV $Q(s_t, j)$, before adding, given as:

$$EnsembleValue = x * V_j + (1 - x) * Q(s_t, j) \quad (5)$$

Hyper-parameter. The weight combinations for IBL and DQN varied from 0.1 to 0.9 in steps of 0.1. In this case, as well, a distinct ensemble model was created, corresponding to each participant, with the IBL and DQN model initialized with that participant’s data. For each weight combination in the ensemble model, the generalized performance was computed by averaging over the F1 score for the ensemble model for each participant’s data.

4 Results

Figure 1 shows the total number of human choices for the left and the right action in the cart-pole game for the ten training trials and the single test trial

for the 42 human participants. Figure 2a and b shows the confusion matrix for the IBL and the DQN model, respectively, taking into consideration the actions predicted by all the model instances (each uniquely trained on the data of one participant, where the number of model instances equaled the number of human participants).

As shown in Table 2, the average F1 score for the DQN model and the IBL model was observed to be 0.829 and 0.809, respectively. Table 3 shows the F1 score of the Ensemble model for each weight combination of the IBL and DQN model. The highest F1 score was achieved for the weight of 0.2 and 0.8 for IBL and DQN, respectively. Notably, the F1 scores of all the weights combinations for the Ensemble exceeded the F1 scores of the individual DQN and IBL models. However, as shown in Table 3, the F1 score is found to decline with the increase in weight of IBL beyond the value of 0.2, indicating the inclination of optimal decision-making toward the long-sighted RL approach.

graph total actions 42 all train session test session nowhere avg.png

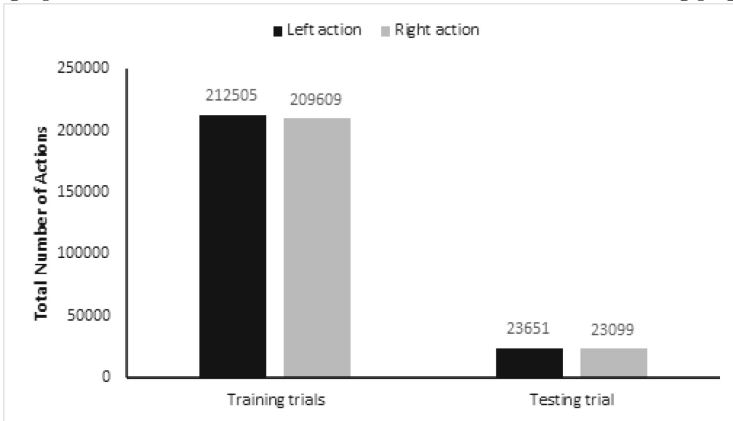


Fig. 1. The total number of human choices for the left and the right action by the 42 players in the cart-pole game, corresponding to all 10 training trials and the single testing trial.

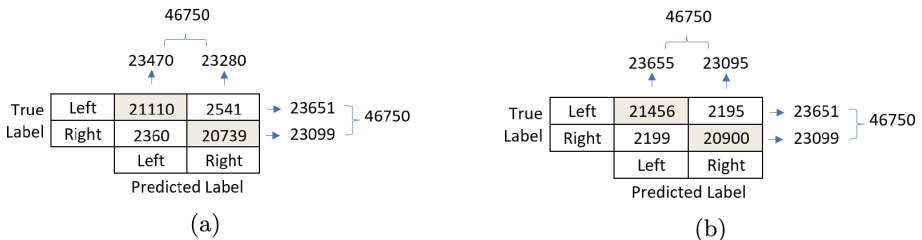


Fig. 2. The confusion matrix for the total actions taken by all the 42 model instances (each uniquely trained on the data of one participant) for a) the IBL model and, b) the DQN model.

Table 1. Hyper-parameters used for the IBL and DQN model.

Name	Value	Name	Value
Cognitive Noise	0.25	Weight initialization	Xavier Uniform
Similarity Threshold	0.85	Batch size	32
(a) IBL		Optimizer	Adam
		Learning rate	0.001
		Discounting factor	0.95
		Dropout rate	0.1
		(b) DQN	

Table 2. Average F1 score of the DQN and IBL model.

Model	Average F1 score
DQN	0.8228
IBL	0.8091

Table 3. F1 score of the Ensemble model, averaged over the model instances corresponding to all human participants, for each weighted combination of the IBL and DQN

Weight IBL (w)	Weight DQN (w)	Mean F1 score
0.1	0.9	0.859
0.2	0.8	0.860
0.3	0.7	0.859
0.4	0.6	0.858
0.5	0.5	0.859
0.6	0.4	0.858
0.7	0.3	0.855
0.8	0.2	0.848
0.9	0.1	0.836

5 Discussion and Conclusion

In this study, we modeled human behavior in the cart-pole game via an IBL cognitive model, a DQN model, and their ensemble. The IBL and DQN models were initialized with human behavior data via memory pre-population and experience replay initialization, respectively. The IBL model performed moderately well in matching human choices with an F1 score of 80%. A possible explanation of why the model fell short of a 100% F1 score might be that the model could recognize the frequent states but not the rarely occurring states. The DQN model outperformed the IBL model in matching human choices, the likely reason being that the DQN approach of maximizing the cumulative reward

fits human decision-making better than the IBL approach of maximizing the current reward. However, the F1 scores of both the models lay in the interval of 80-82%, indicating their inability to model a significant portion of the human decisions.

The ensemble of the IBL and the DQN models was developed to bring the principles of the cognitive and RL approaches under one roof. Results revealed that the ensemble models could predict human choices with greater accuracy than the standalone cognitive and RL models. The likely reason could be that the human decision-making process is based on a trade-off between immediate short-term and long-term goals, more accurately modeled by the ensemble. The optimal weights of IBL and DQN were found to be 0.2 and 0.8, respectively. This points to the trade-off being inclined towards the far-sighted approach.

The limitation of this research is that in the process of data extraction from the recorded human game-play, only two actions, left and right, were considered, but for a human, another outcome of ‘no action’ occurred for some situations while switching between the left and right action key. Dropping the ‘no action’ action might prevent capturing actual human behavior. Additionally, a very simple architecture was used for DQN, and there may be a possibility to push the DQN results a little further through more complex networks.

There is a broad scope of future work based on this study. Various modifications could be done to the IBL models by importing concepts from other cognitive mechanisms [7], which could increase the match with human behavior. More complex network architectures could be examined for improving DQN performance. It would be interesting to observe the models’ performance if different rewards are associated with a win or loss in the episode, and each action’s reward is obtained via discounting, unlike predefined rewards for each step, as in this study. Apart from DQN, other more advanced state-of-the-art algorithms could also be investigated. Moreover, the approaches possible to achieve an ensemble of these two techniques from such different paradigms are limited only by one’s imagination. Instead of addition, multiplication of the corresponding action scores to give a final measure of an action’s goodness, weighted multiplication, and a hybrid mechanism to merge the working principle of the two algorithms, to mention a few.

References

1. Anderson, J., Bothell, D., Byrne, M., Douglass, S., Lebiere, C., Qin, Y.: An integrated theory of mind. *Psychol. Rev.* **111**, 1036–1060 (2004)
2. Arulkumaran, K., Deisenroth, M.P., Brundage, M., Bharath, A.A.: A brief survey of deep reinforcement learning. arXiv preprint [arXiv:1708.05866](https://arxiv.org/abs/1708.05866) (2017)
3. Barto, A.G., Sutton, R.S., Anderson, C.W.: Neuronlike adaptive elements that can solve difficult learning control problems. *IEEE Trans. Syst. Man Cybern. SMC-* **13**(5), 834–846 (1983). <https://doi.org/10.1109/TSMC.1983.6313077>

4. Duarte, F.F., Lau, N., Pereira, A., Reis, L.P.: Benchmarking deep and non-deep reinforcement learning algorithms for discrete environments. In: Silva, M.F., Luís Lima, J., Reis, L.P., Sanfeliu, A., Tardioli, D. (eds.) ROBOT 2019. AISC, vol. 1093, pp. 263–275. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-36150-1_22
5. Fgadaleta: top 4 reasons why reinforcement learning sucks (ep. 83) (2019). <https://datascienceathome.com/what-is-wrong-with-reinforcement-learning/>
6. Gonzalez, C., Dutt, V.: Instance-based learning models of training. In: Proceedings of the human factors and ergonomics society annual meeting, vol. 54, pp. 2319–2323. SAGE Publications Sage CA: Los Angeles, CA (2010)
7. Kotseruba, I., Tsotsos, J.K.: A review of 40 years of cognitive architecture research: core cognitive abilities and practical applications. arXiv preprint [arXiv:1610.08602](https://arxiv.org/abs/1610.08602) (2016)
8. Kumar, S.: Balancing a cartpole system with reinforcement learning—a tutorial. arXiv preprint [arXiv:2006.04938](https://arxiv.org/abs/2006.04938) (2020)
9. Meltzer, B., Michie, D.: Machine intelligence 4 (1970)
10. Mnih, V., et al.: Playing Atari with deep reinforcement learning. arXiv preprint [arXiv:1312.5602](https://arxiv.org/abs/1312.5602) (2013)
11. Mnih, V., et al.: Human-level control through deep reinforcement learning. *Nature* **518**(7540), 529–533 (2015)
12. Mothanna, Y., Hewahi, N.: Review on reinforcement learning in cartpole game. In: 2022 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT), pp. 344–349. IEEE (2022)
13. Nagendra, S., Podila, N., Ugarakhod, R., George, K.: Comparison of reinforcement learning algorithms applied to the Cart-Pole problem. In: 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 26–32. IEEE (2017)
14. Schaal, S.: Is imitation learning the route to humanoid robots? *Trends Cogn. Sci.* **3**(6), 233–242 (1999)
15. Sunden, P.: Q-learning and deep Q-learning in OpenAI gym cartpole classic control environment (2022)
16. Surriani, A., Wahyunggoro, O., Cahyadi, A.I.: Reinforcement learning for cart pole inverted pendulum system. In: 2021 IEEE Industrial Electronics and Applications Conference (IEACon), pp. 297–301. IEEE (2021)
17. Sutton, R.S., Barto, A.G.: Reinforcement learning: an introduction MIT press. Cambridge, MA 22447 (1998)
18. Tan, Z., Karakose, M.: Optimized deep reinforcement learning approach for dynamic system. In: 2020 IEEE International Symposium on Systems Engineering (ISSE), pp. 1–4. IEEE (2020)
19. Wang, X., Gu, Y., Cheng, Y., Liu, A., Chen, C.P.: Approximate policy-based accelerated deep reinforcement learning. *IEEE Trans. Neural Netw. Learn. Syst.* **31**(6), 1820–1830 (2019)
20. Watkins, C.J.C.H., Dayan, P.: Q-learning. *Mach. Learn.* **8**, 279–292 (1992). <https://doi.org/10.1007/BF00992698>