Deep TAMER: Interactive Agent Shaping in High-Dimensional State Spaces

By Garrett Warnell, Nicholas Waytowich, Vernon Lawhern, and Peter Stone

Presenter: Alex Tsolovikos

November 2, 2021

Human Input in Reinforcement Learning

Autonomous agents need a **policy** for *sequential decision making*

Hand-Coded

Learning from Demonstrations
Inverse RL

Reward Shaping
TAMER

Reinforcement Learning

Expert Human
Designer

Expert Human
Demonstrator

Non-Expert Human
Critic

No Human
Input

Expert demonstrations
are needed

No demonstrations

The TAMER Framework

Training an **A**gent **M**anually via **E**valuative **R**einforcement

- Learn by interacting with a non-expert human
- Non-expert human observes system performance and “critiques” how good or bad it is via a **scalar feedback**
- No demonstrations – only scalar critique
- Useful for tasks that are hard for a human to demonstrate but easy to critique

The TAMER Framework

- S: set of states
- A: set of actions
- **Agent**: selects actions $(a_1, a_2, \dots)$ that lead to a sequence of states $(s_0, s_1, s_2, \dots)$
- **Human**: observes $(s_0, s_1, s_2, \dots)$ and periodically provides scalar feedback $(h_0, h_1, \dots)$
- Large h: good behavior
- Implicit human reward function: $H(\cdot,\cdot) : S \times A \rightarrow \mathrm{R}$

Learning:

Estimate $\hat{\mathrm{H}}$

→

Greedy Policy:

$\pi(s) = argmax_a \, \hat{H}(s, a)$

Implicit Human Function vs Q-Function

Q-Function

- Associated with a state-action reward
- In general, no labels available
- A value for each policy and state-action pair

$$Q(\cdot,\cdot) : S \times A \rightarrow \mathrm{R}$$

$$Q(s,a) = r(s,a) + \gamma \max_{a'} Q(s',a')$$

Implicit Human Reward Function

- No predefined reward
- Labeled by human feedback
- Provided at unknown intervals

$$H(\cdot,\cdot) : S \times A \rightarrow \mathrm{R}$$

$$H(s,a) = ?$$

Deep TAMER as Supervised Learning

Agent observes:

- State – action pairs:

$$x_i = (s_i, a_i, \underbrace{t_i^{start}, t_i^{end}}_{\text{Time spent in state}})$$

- Human feedback:

$$y_j = (h_j, t_j^{feedback})$$

Time feedback is given

Which x_i does the feedback correspond to?

- No one-to-one correspondence

$$\{x_i\} \rightarrow y_j$$

$$\{x_i, x_{i+1}, x_{i+2}\} \rightarrow y_j$$

$$\{x_i, x_{i+1}\} \rightarrow \{0\}$$

- Assumptions:
  - $t^{feedback} < t^{start}$: no correspondence
  - $t^{start} \leq t^{feedback} \leq t^{end}$: correspondence
  - $t^{feedback} \gg t^{end}$: correspondence goes to zero

Deep TAMER as Online Supervised Learning

Loss Function:

$$\ell(\hat{H}; x_i, y_j) = \underbrace{w\left(t_i^{start}, t_i^{end}, t_j^{feedback}\right)}_{\text{Weight}}\left[\hat{H}(s_i, a_i) - h_j\right]^2$$

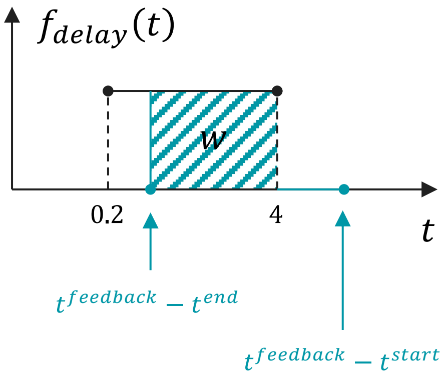
Weight:

$t^{feedback} < t^{start}$: $w = 0$

$t^{feedback} \gg t^{end}$: $w \to 0$

else: $w \neq 0$

Uniform distribution



Weight Function:

$$w\left(t_i^{start}, t_i^{end}, t_j^{feedback}\right) = \int_{t^{feedback} - t^{end}}^{t^{feedback} - t^{start}} f_{delay}(t)dt$$

Probability of correspondence

Deep TAMER as Online Supervised Learning

Optimization Goal:

$$\hat{H}^* = \arg\min_{\hat{H}} E_{x,y}\left[\ell\left(\hat{H}; x, y\right)\right]$$

- Minimize loss in the statistical sense
- Online supervised learning: treat observations as realizations of a random variable

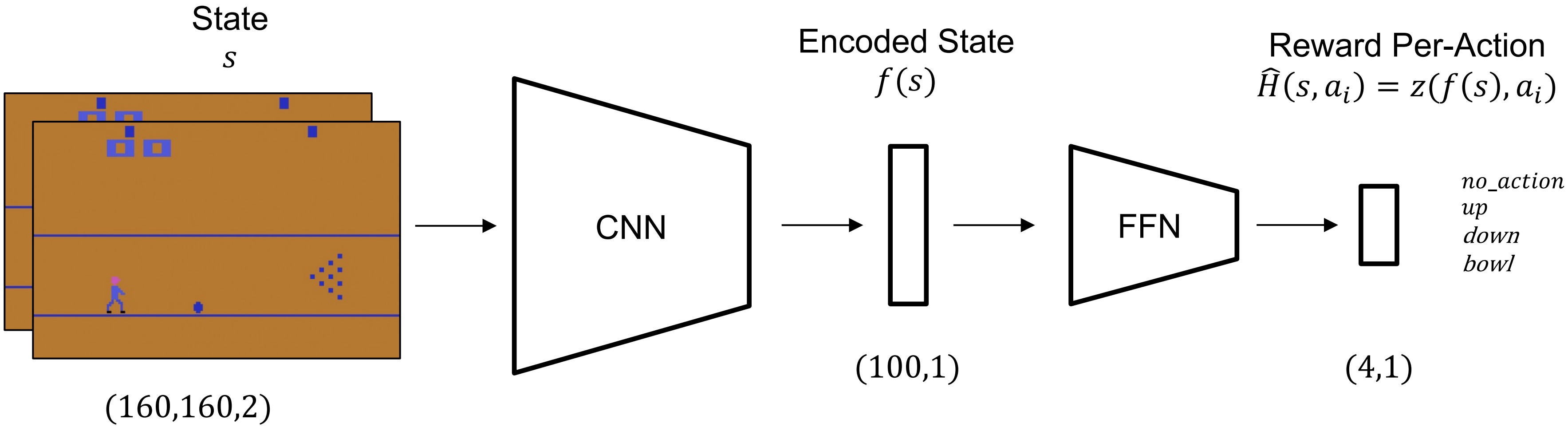
Stochastic Gradient Descent Updates:

$$\hat{H}_{k+1} = \hat{H}_k - \eta_k \nabla_{\hat{H}} \ell\left(\hat{H}_k; \underbrace{x_{i_k}, y_{j_k}}\right)$$

Sampled from experience $(x_1, x_2, \ldots)$ and feedback $(y_1, y_2, \ldots)$
for pairs with $w \neq 0$

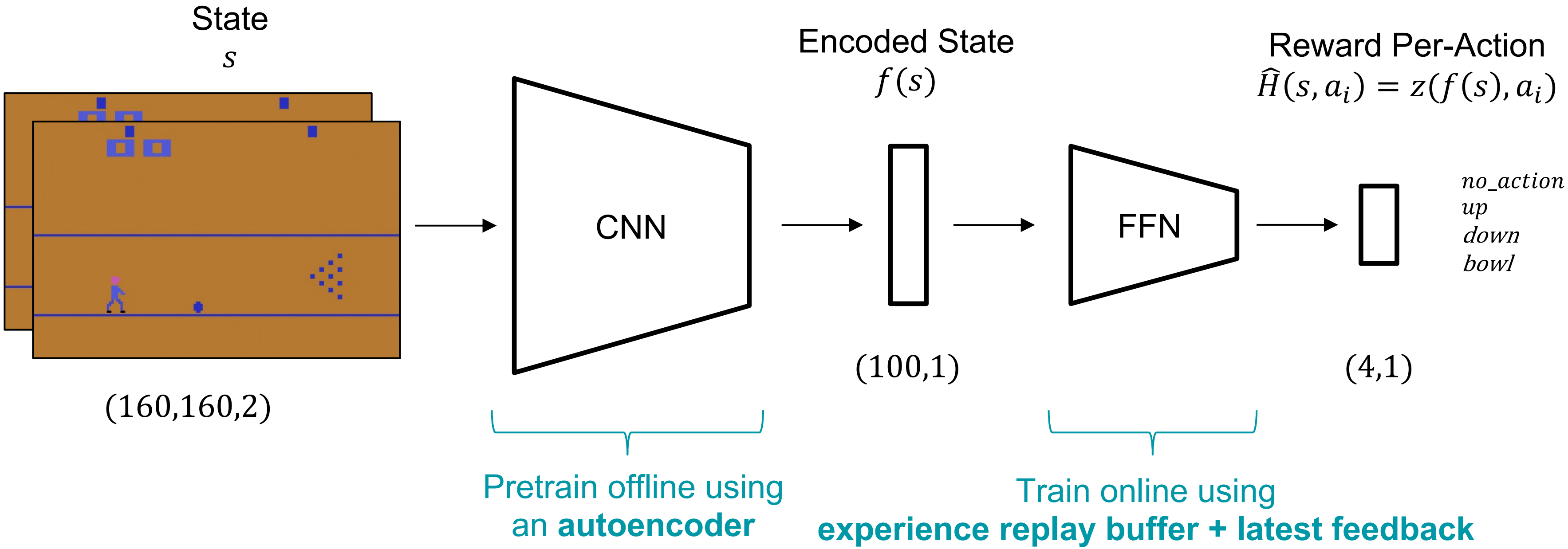
High-Dimensional Systems: Atari Bowling

Deep Reward Function for the Atari Bowling game:



High-Dimensional Systems: Atari Bowling

Deep Reward Function for the Atari Bowling game:



TAMER vs Deep TAMER

Original TAMER

- Human reward function $\widehat{H}$: **linear**
- Reward h_j applies to **all** state-action pairs (s_i, a_i)

$$\ell(\widehat{H}; \{\boldsymbol{x}\}, y_j) = \frac{1}{2}\left(h_j - \sum_i w\left(t_i^s, t_i^e, t_j^f\right)\widehat{H}(s_i, a_i)\right)^2$$

Deep TAMER

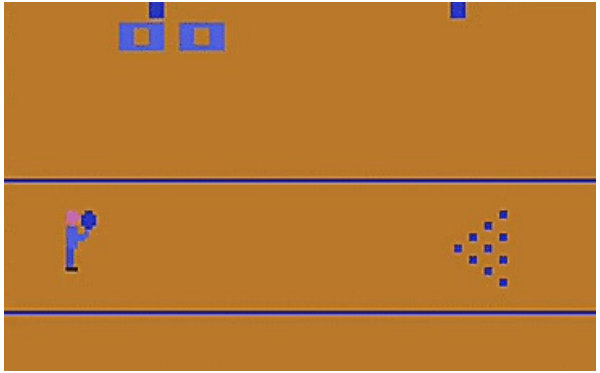
- Human reward function $\widehat{H}$: **deep CNN**
- Reward h_j applies to **one** state-action pair (s_i, a_i)

$$\ell(\widehat{H}; x_i, y_j) = w\left(t_i^s, t_i^e, t_j^f\right)\left[\widehat{H}(s_i, a_i) - h_j\right]^2$$

Intuition: human's feedback applies to individual state-action pair, not any state-action pair

Experiments: Atari Bowling

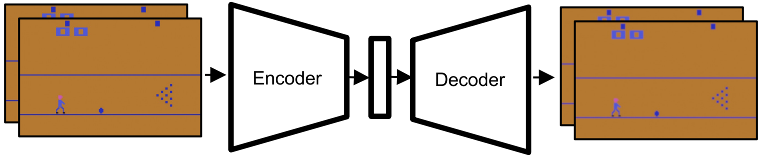
- Actions:
  - **no-action**
  - **up**
  - **down**
  - **bowl**
- State:
  - 2 most recent grayscale images $(160{,}160{,}2)$
  - 20 FPS
- 10 frames per game
- Metric: **score per game**
- Maximum score: **270**
- Implementation: **OpenAI Gym**



Training

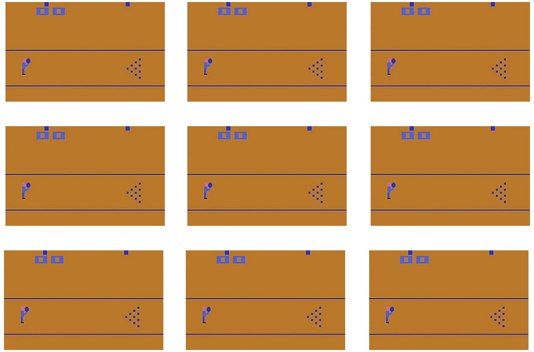
Offline

- CNN pretrained with an **autoencoder**

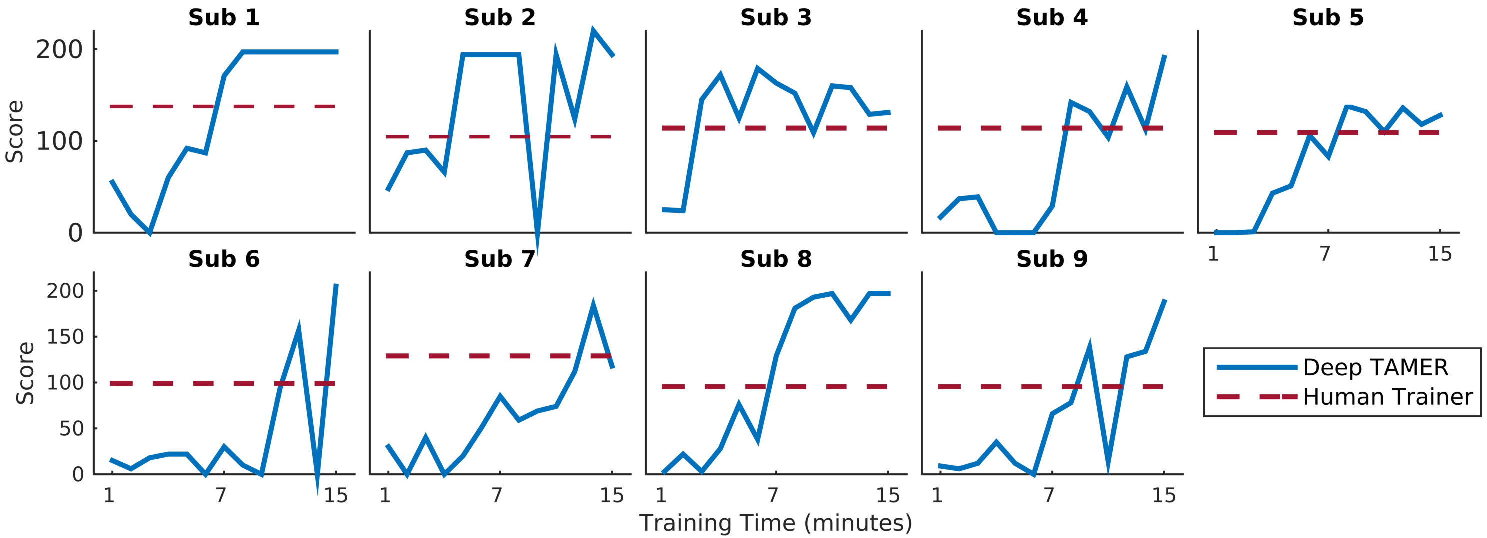


Online

- FNN trained online
- **9 trainers** do the following:
  - **Record human performance**: 2 games
  - **Familiarize with giving feedback:** 10 minutes
  - **Train using Deep TAMER:** 15 minutes
  - **Train using Original TAMER:** 15 minutes



Trainers vs Trained Agent



Agents perform **better than their trainers** after ~7 minutes

Evaluation

Deep TAMER is compared with:

- **Original TAMER:** linear rewards model; global loss function
- **Double Deep Q-Learning:** online, off-policy
- **A3C:** Asynchronous advantage actor-critic; uses 16 parallel actor learners
- **Human Trainers** performance
- **Expert Human** performance (Mnih et al. 2015)
- **Learning from Demonstrations** (Hester et al. 2017)

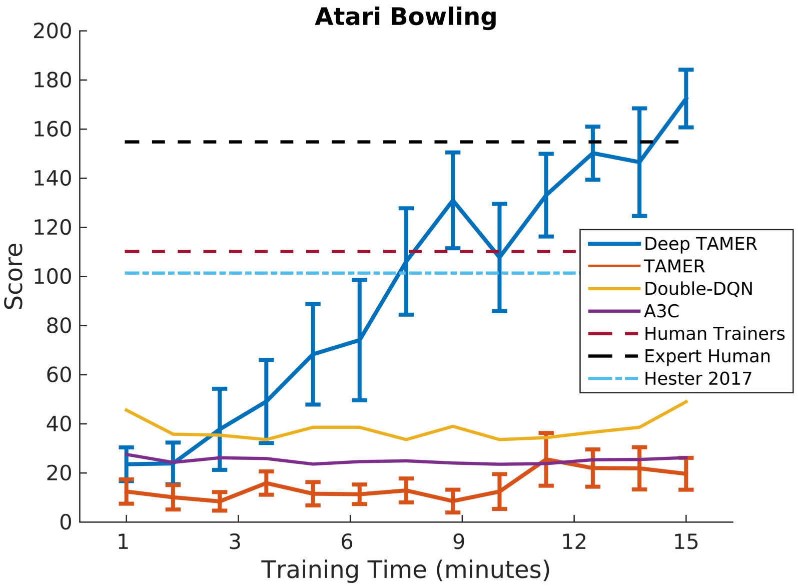
Evaluation

Other Methods:

- **Double-DQN and A3C:** Fail to learn in 15 minutes or **even with 50-100 million** training steps
- **Original TAMER:** Fails, since it uses a **linear model** for the human reward function
- **Demonstrations (Hester et al. 2017):** Also fails; **task is hard for a human to demonstrate**

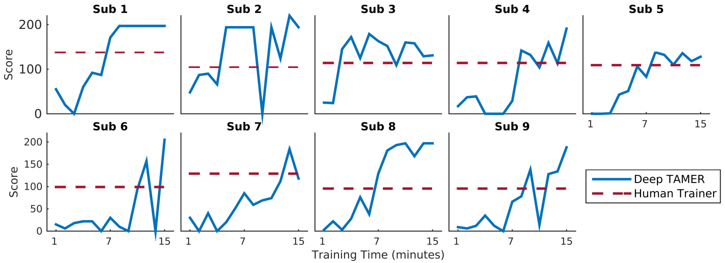
Deep TAMER:

- Better than trainers after 7 minutes of training; **task is easy to critique**
- Better than human expert (Mnih et al. 2015)



Limitations / Open Issues

- Performance increase can be **noisy** due to stochastic optimization



- Difficult to do hyperparameter search because obtaining more human interaction data is difficult
- **Does not** seek to maximize discounted sum of future rewards; only short-term human feedback h
- No reward is directly considered

Summary

- Deep learning of policies by interacting in real-time with a non-expert human
- Non-expert human observes system performance and “critiques” how good or bad it is
- Extension of the original TAMER to high-dimensional systems
- Agents are able to learn the Atari Bowling in just 15 minutes of interactions with human critics
- Outperforms human trainers, human experts, and related RL methods in the Atari Bowling game
- **Useful when a task is hard for a human to demonstrate but easy to critique**

Extended Readings

- **Deep COACH (similar idea, but with actor-critic):** Arumugam, Dilip, et al. "Deep reinforcement learning from policy-dependent human feedback." (2019).
- **Combining Deep TAMER with distant rewards:** Arakawa, Riku, et al. "DQN-TAMER: Human-in-the-loop reinforcement learning with intractable feedback." (2018).
- **Combining demonstrations and human preference:** Ibarz, Borja, et al. "Reward learning from human preferences and demonstrations in Atari." (2018).
- **Survey on human guidance in deep RL:** Zhang, Ruohan, et al. "Leveraging human guidance for deep reinforcement learning tasks." (2019).