Deep Reinforcement Learning: Q-Learning

Garima Lalwani Karan Ganju Unnat Jain

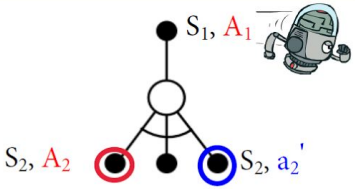
Today’s takeaways

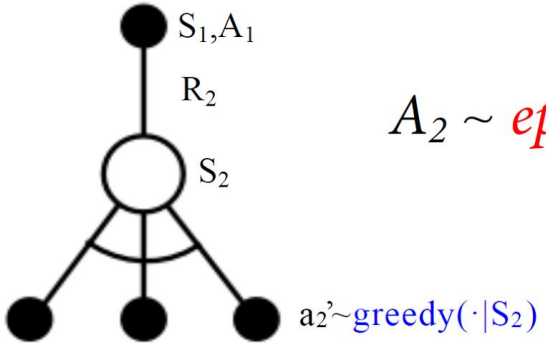
- Bonus RL recap
- Functional Approximation
- Deep Q Network
- Double Deep Q Network
- Dueling Networks
- Recurrent DQN
  - Solving “Doom”
- Hierarchical DQN

Today’s takeaways

- **Bonus RL recap**
- Functional Approximation
- Deep Q Network
- Double Deep Q Network
- Dueling Networks
- Recurrent DQN
  - Solving “Doom”
- Hierarchical DQN

Q-Learning





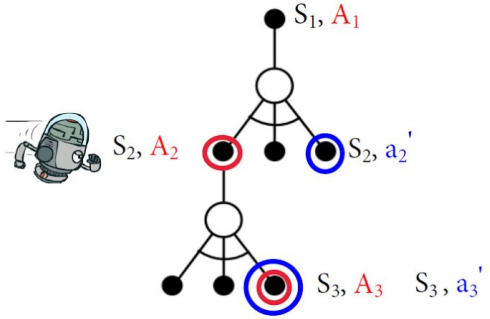
$A_2 \sim$ *epsilon greedy* $(\cdot|S_2)$

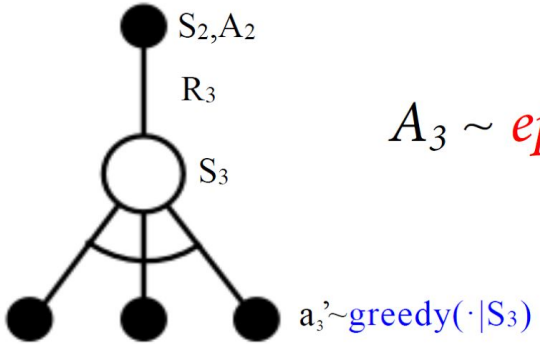
$$Q(S_1, A_1) := Q(S_1, A_1) + \alpha\left(R_2 + \gamma \max_{a_2'} Q(S_2, a_2') - Q(S_1, A_1)\right)$$

David Silver's Introduction to RL lectures

Peter Abbeel's Artificial Intelligence - Berkeley (Spring 2015)

Q-Learning





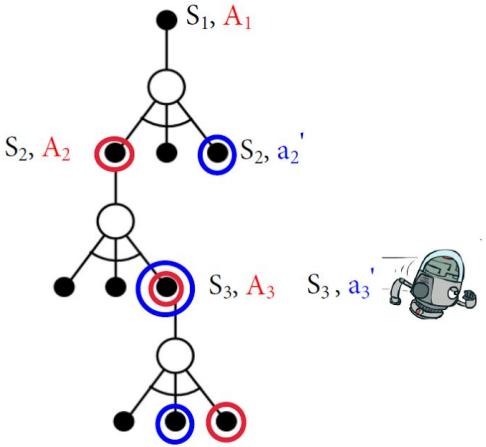
$A_3 \sim$ *epsilon greedy* $(\cdot|S_3)$

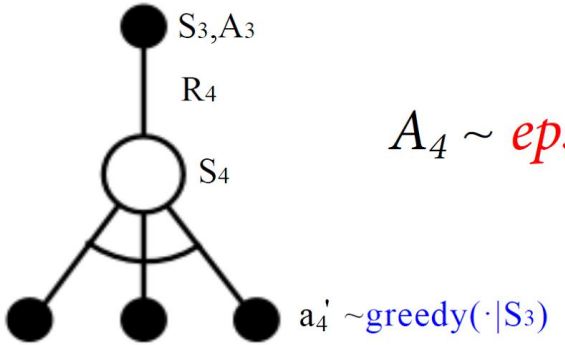
$$Q(S_2, A_2) := Q(S_2, A_2) + \alpha\left(R_3 + \gamma \max_{a_3'} Q(S_3, a_3') - Q(S_2, A_2)\right)$$

David Silver's Introduction to RL lectures

Peter Abbeel's Artificial Intelligence - Berkeley (Spring 2015)

Q-Learning





$A_4 \sim$ *epsilon greedy* $(\cdot|S_3)$

$$Q(S_3, A_3) := Q(S_3, A_3) + \alpha\left(R_4 + \gamma \max_{a_4'} Q(S_4, a_4') - Q(S_3, A_3)\right)$$

David Silver's Introduction to RL lectures

Peter Abbeel's Artificial Intelligence - Berkeley (Spring 2015)

Today’s takeaways

- Bonus RL recap
- **Functional Approximation**
- Deep Q Network
- Double Deep Q Network
- Dueling Networks
- Recurrent DQN
  - Solving “Doom”
- Hierarchical DQN

Function Approximation - Why?

- Value functions
  - Every state **s** has an entry **V(s)**
  - Every state-action pair **(s, a)** has an entry **Q(s, a)**
- How to get Q(s,a) → Table lookup
- What about large MDPs?
  - Estimate value function with **function approximation**

$$\hat{v}(s, \mathbf{w}) \approx v_\pi(s)$$

$$\text{or } \hat{q}(s, a, \mathbf{w}) \approx q_\pi(s, a)$$

  - Generalise from seen states to unseen states

Function Approximation - How?

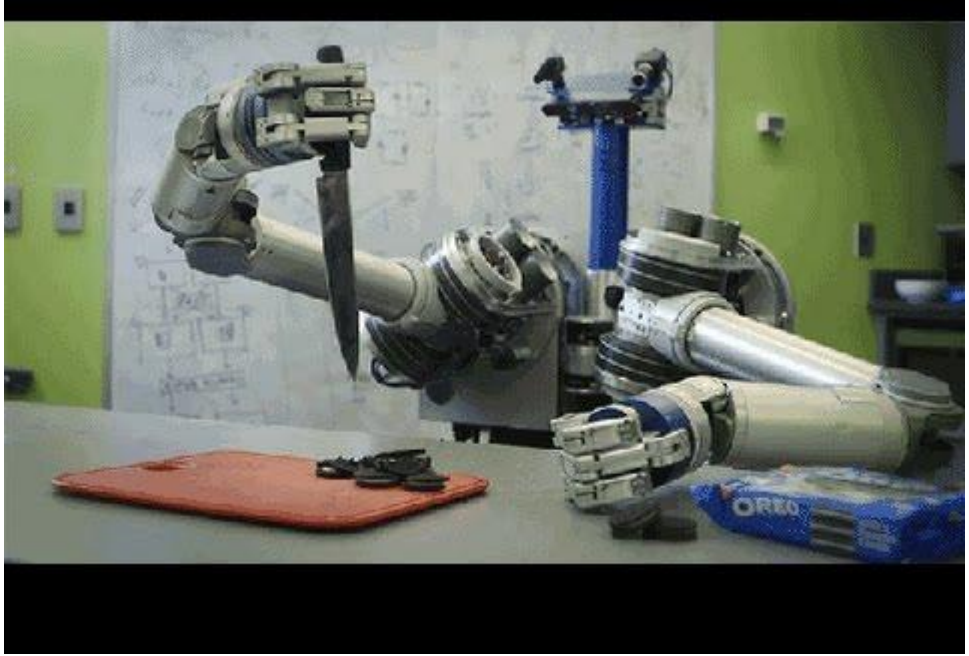
- Why Q?

$$\pi_{\text{new}}(a|s) = \begin{cases} 1 & \text{if } a = \underset{a \in \mathcal{A}}{\text{argmax}}\ q_{\text{old}}(s,a) \\ 0 & \textit{otherwise} \end{cases}$$

- How to approximate?
  - Features for state s | s,a → **x**(s,a)
  - Linear model | Q(s,a) = $\mathbf{w}^T\mathbf{x}$(s,a)
  - Deep Neural Nets - CS598 | Q(s,a) = **NN**(s,a)

$$\hat{v}(s, \mathbf{w}) \approx v_\pi(s)$$

$$\text{or } \hat{q}(s, a, \mathbf{w}) \approx q_\pi(s, a)$$



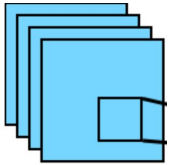
Function Approximation - Demo

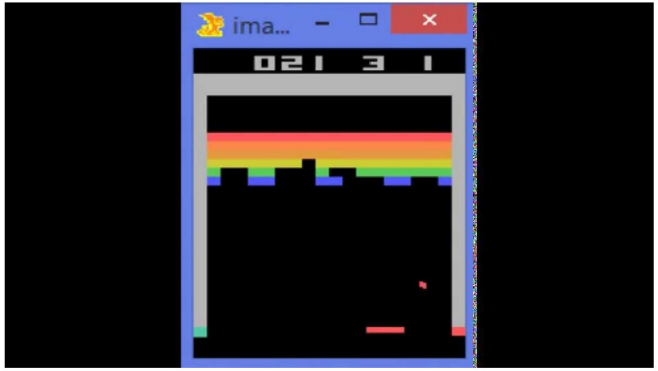
Today’s takeaways

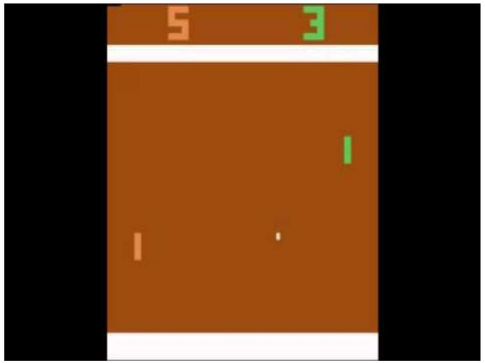
- Bonus RL recap
- Functional Approximation
- **Deep Q Network**
- Double Deep Q Network
- Dueling Networks
- Recurrent DQN
  - Solving “Doom”
- Hierarchical DQN

Deep Q Network

1) Input:
4 images = current frame + 3 previous







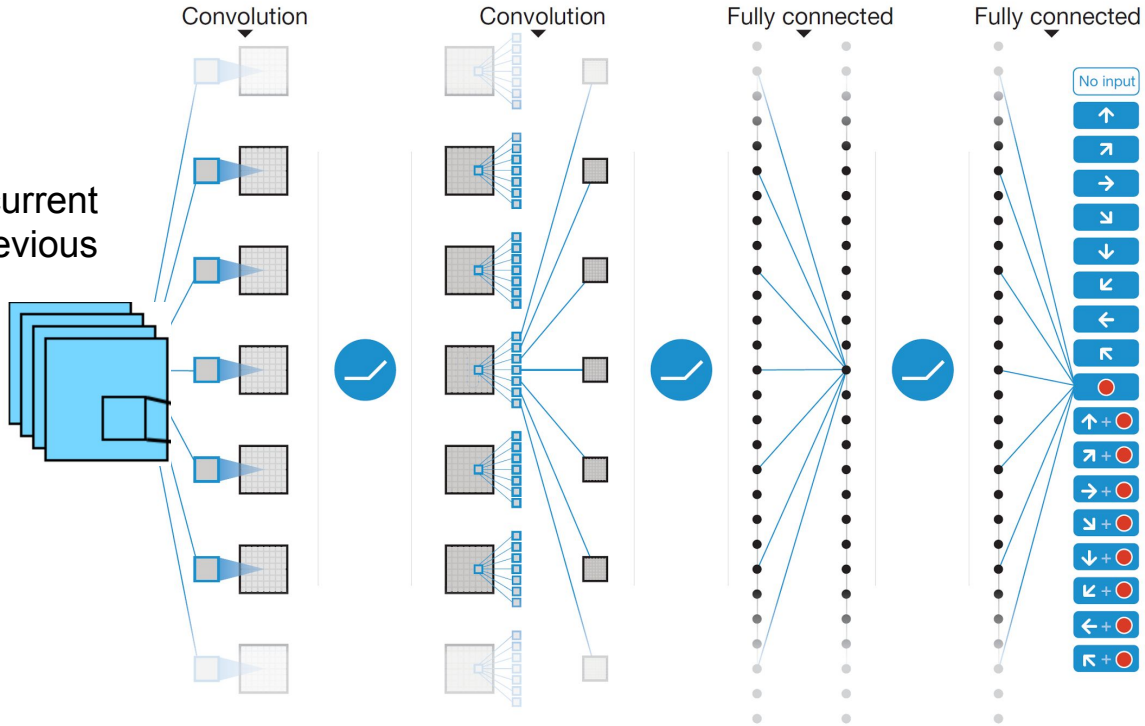
2) Output: $Q(s,a_i)$

$Q(s,a_1)$
$Q(s,a_2)$
$Q(s,a_3)$
.
.
.
.
.
.
.
.
.
$Q(s,a_{18})$

Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." *Nature* 518.7540 (2015): 529-533.

Deep Q Network

1) Input:
4 images = current frame + 3 previous



2) Output: $Q(s,a_i)$

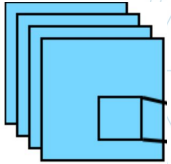
$Q(s,a_1)$
$Q(s,a_2)$
$Q(s,a_3)$
.
.
.
.
.
.
.
.
.
$Q(s,a_{18})$

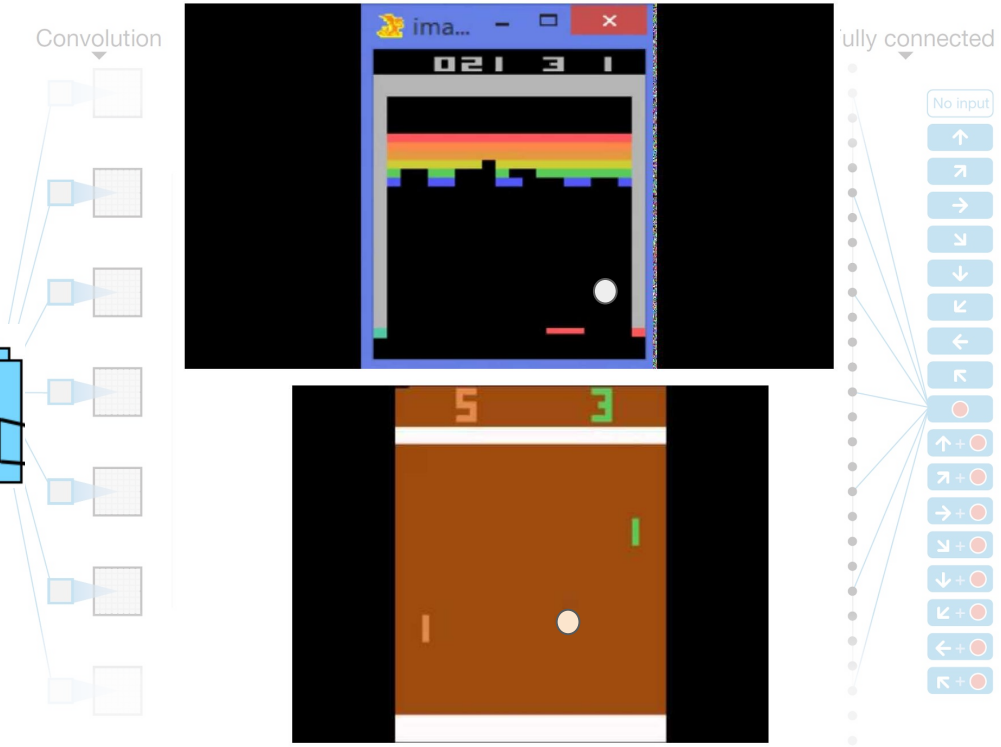
Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." *Nature* 518.7540 (2015): 529-533.

Deep Q Network

1) Input:
4 images = current frame + 3 previous

?





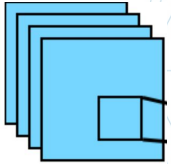
2) Output: $Q(s,a_i)$
$Q(s,a_1)$
$Q(s,a_2)$
$Q(s,a_3)$
.
.
.
.
.
.
.
.
.
$Q(s,a_{18})$

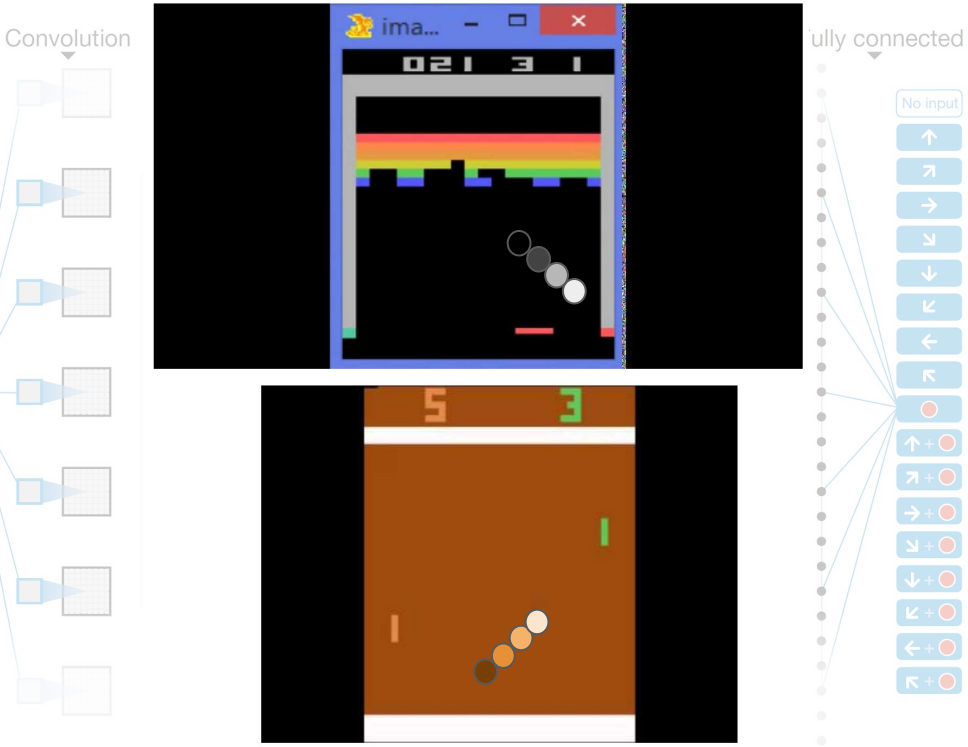
Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." *Nature* 518.7540 (2015): 529-533.

Deep Q Network

1) Input:
4 images = current frame + 3 previous

$\boldsymbol{\phi}(\mathbf{s})$





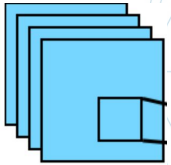
2) Output: $Q(s,a_i)$
$Q(s,a_1)$
$Q(s,a_2)$
$Q(s,a_3)$
.
.
.
.
.
.
.
.
.
$Q(s,a_{18})$

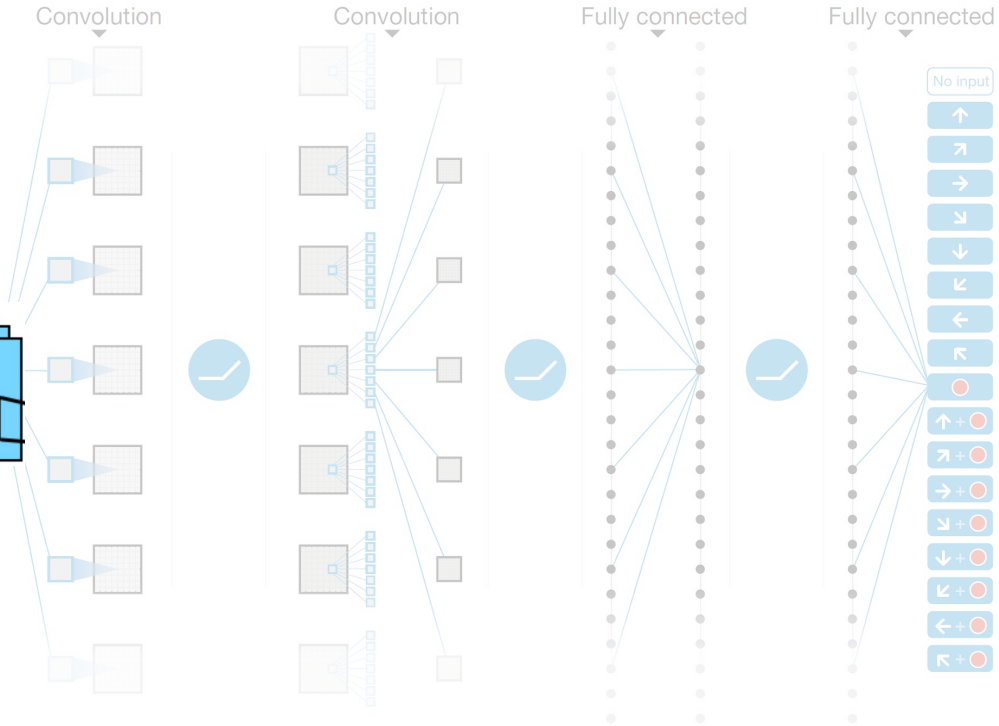
Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." *Nature* 518.7540 (2015): 529-533.

Deep Q Network

1) Input:
4 images = current frame + 3 previous

$\phi(s)$





2) Output: $Q(s,a_i)$

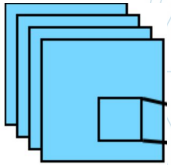
$Q(s,a_1)$
$Q(s,a_2)$
$Q(s,a_3)$
.
.
.
.
.
.
.
.
$Q(s,a_{18})$

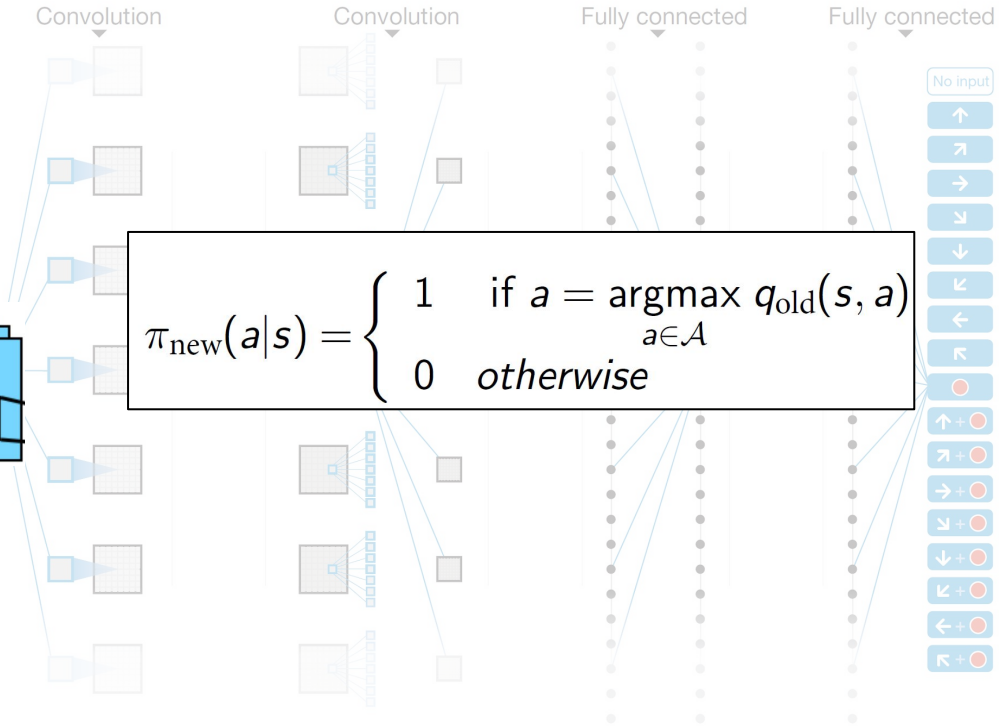
Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." *Nature* 518.7540 (2015): 529-533.

Deep Q Network

1) Input:
4 images = current frame + 3 previous

$\phi(s)$





$$\pi_{\text{new}}(a|s) = \begin{cases} 1 & \text{if } a = \underset{a \in \mathcal{A}}{\operatorname{argmax}}\, q_{\text{old}}(s, a) \\ 0 & \textit{otherwise} \end{cases}$$

2) Output: $Q(s,a_i)$
$Q(s,a_1)$
$Q(s,a_2)$
$Q(s,a_3)$
.
.
.
.
.
.
.
.
$Q(s,a_{18})$

Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." *Nature* 518.7540 (2015): 529-533.

Supervised SGD (lec2) vs Q-Learning SGD

- SGD update assuming supervision

$$J(\mathbf{w}) = \mathbb{E}_\pi \left[(q_\pi(S, A) - \hat{q}(S, A, \mathbf{w}))^2 \right]$$

$$\Delta \mathbf{w} = \alpha (q_\pi(S, A) - \hat{q}(S, A, \mathbf{w})) \nabla_{\mathbf{w}} \hat{q}(S, A, \mathbf{w})$$

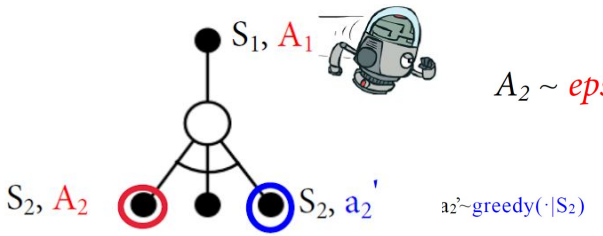
David Silver's Deep Learning Tutorial, ICML 2016

Supervised SGD (lec2) vs Q-Learning SGD

- SGD update assuming supervision

$$J(\mathbf{w}) = \mathbb{E}_\pi \left[(q_\pi(S, A) - \hat{q}(S, A, \mathbf{w}))^2 \right]$$

$$\Delta \mathbf{w} = \alpha (q_\pi(S, A) - \hat{q}(S, A, \mathbf{w})) \nabla_{\mathbf{w}} \hat{q}(S, A, \mathbf{w})$$



$A_2 \sim$ *epsilon greedy* $(\cdot|S_2)$

$$Q(S_1, A_1) := Q(S_1, A_1) + \alpha \left(R_2 + \gamma \max_{a_2'} Q(S_2, a_2') - Q(S_1, A_1) \right)$$

David Silver's Deep Learning Tutorial, ICML 2016

Supervised SGD (lec2) vs Q-Learning SGD

- SGD update assuming supervision

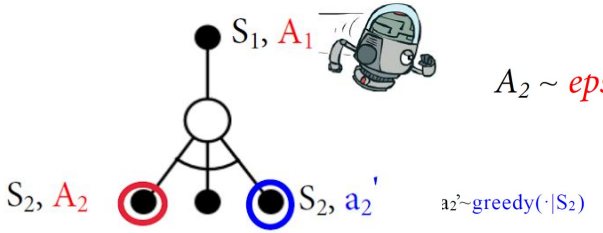
$$J(\mathbf{w}) = \mathbb{E}_\pi \left[(q_\pi(S, A) - \hat{q}(S, A, \mathbf{w}))^2 \right]$$

$$\Delta \mathbf{w} = \alpha (q_\pi(S, A) - \hat{q}(S, A, \mathbf{w})) \nabla_{\mathbf{w}} \hat{q}(S, A, \mathbf{w})$$

- SGD update for Q-Learning

$$J(w) = \mathbb{E} \left[\left(\underbrace{r + \gamma \max_{a'} Q(s', a', w)}_{\text{target}} - Q(s, a, w) \right)^2 \right]$$

$$\Delta \mathbf{w} = \left(r + \gamma \max_{a'} Q(s', a', w) - Q(s, a, w) \right) \frac{\partial Q(s, a, w)}{\partial w}$$



$A_2 \sim$ *epsilon greedy* $(\cdot | S_2)$

$$Q(S_1, A_1) := Q(S_1, A_1) + \alpha \left(R_2 + \gamma \max_{a_2'} Q(S_2, a_2') - Q(S_1, A_1) \right)$$

David Silver's Deep Learning Tutorial, ICML 2016

Training tricks

- Issues:
  a. Data is sequential
     - Successive samples are correlated, non-iid
     - An experience is visited only once in online learning
  b. Policy changes rapidly with slight changes to Q-values
     - Policy may oscillate

Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." *Nature* 518.7540 (2015): 529-533.

Training tricks

- Issues:
  a. Data is sequential
     - Successive samples are correlated, non-iid
     - An experience is visited only once in online learning

Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." *Nature* 518.7540 (2015): 529-533.

Training tricks

- Issues:
  a. Data is sequential
     - Successive samples are correlated, non-iid
     - An experience is visited only once in online learning
  - **Solution: *'Experience Replay'* : Work on a dataset - Sample randomly and repeatedly**
     - Build dataset
        - Take action a_t according to ε-greedy policy
        - Store transition/experience (s_t , a_t ,r_{t+1},s_{t+1}) in dataset D ('*replay memory*')
     - Sample randomly mini-batch (32 experiences) of (s, a, r, s')) from D

$$J(w) = \mathbb{E}_{s,a,r,s' \sim \mathcal{D}} \left[\left(r + \gamma \max_{a'} Q(s', a', w) - Q(s, a, w) \right)^2 \right]$$

Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." *Nature* 518.7540 (2015): 529-533.

Training tricks

- Issues:
  a. Data is sequential
     - Successive samples are correlated, non-iid
     - An experience is visited only once in online learning
  - **Solution: *'Experience Replay'* : Work on a dataset - Sample randomly and repeatedly**
     - Build dataset
        - Take action a_t according to ε-greedy policy
        - Store transition/experience (s_t , a_t ,r_{t+1},s_{t+1}) in dataset D ('*replay memory*')
     - Sample randomly mini-batch (32 experiences) of (s, a, r, s')) from D

$$J(w) = \mathbb{E}_{s,a,r,s' \sim \mathcal{D}} \left[\left(r + \gamma \max_{a'} Q(s', a', w) - Q(s, a, w) \right)^2 \right]$$

Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." *Nature* 518.7540 (2015): 529-533.

Training tricks

- Issues:
  a. Data is sequential ✓ **Experience replay**
     - Successive samples are correlated, non-iid
     - An experience is visited only once in online learning
  b. Policy changes rapidly with slight changes to Q-values
     - Policy may oscillate

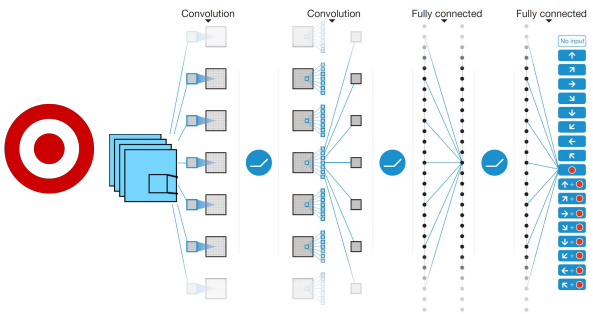
Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." *Nature* 518.7540 (2015): 529-533.

Training tricks

- Issues:
  a. Data is sequential ✔ **Experience replay**
     - Successive samples are correlated, non-iid
     - An experience is visited only once in online learning
  b. Policy changes rapidly with slight changes to Q-values
     - Policy may oscillate
- **Solution: *'Target Network'* : Stale updates**
  - C step delay between update of Q and its use as targets

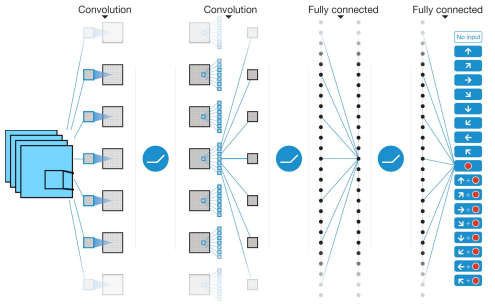
$$J(w) = \mathbb{E}_{s,a,r,s' \sim \mathcal{D}} \left[\left(r + \gamma \max_{a'} Q(s', a', w^{-}) - Q(s, a, w) \right)^2 \right]$$

Network 1



Tells me Q(s,a) targets (w_{i-1})

Network 2



Q values are updated every SGD step (w_i)

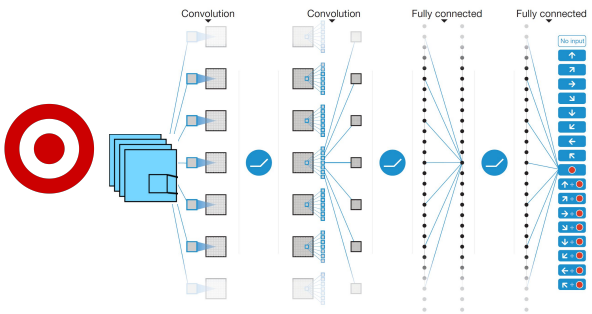
Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." *Nature* 518.7540 (2015): 529-533.

Training tricks

- Issues:
  a. Data is sequential ✓ **Experience replay**
     - Successive samples are correlated, non-iid
     - An experience is visited only once in online learning
  b. Policy changes rapidly with slight changes to Q-values
     - Policy may oscillate
- **Solution: *'Target Network'* : Stale updates**
  - C step delay between update of Q and its use as targets

$$J(w) = \mathbb{E}_{s,a,r,s' \sim \mathcal{D}} \left[\left(r + \gamma \max_{a'} Q(s', a', w^-) - Q(s, a, w) \right)^2 \right]$$

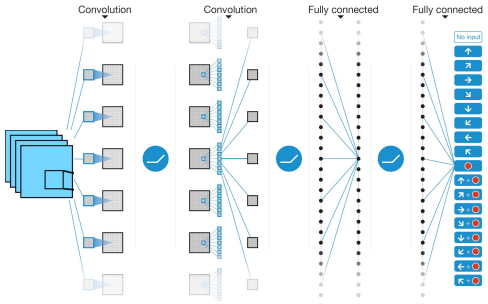
Network 1



Tells me Q(s,a) targets (w_i)

After 10,000 SGD updates

Network 2



Q values are updated every SGD step (w_{i+1})

Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." *Nature* 518.7540 (2015): 529-533.

Training tricks

- Issues:
  a. Data is sequential ✓ **Experience replay**
     - Successive samples are correlated, non-iid
     - An experience is visited only once in online learning
  b. Policy changes rapidly with slight changes to Q-values ✓ **Target Network**
     - Policy may oscillate

Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." *Nature* 518.7540 (2015): 529-533.

Training tricks

- Issues:
  a. Data is sequential ✓ **Experience replay**
     - Successive samples are correlated, non-iid
     - An experience is visited only once in online learning
  b. Policy changes rapidly with slight changes to Q-values ✓ **Target Network**
     - Policy may oscillate

Game	With replay, with target Q	With replay, without target Q	Without replay, with target Q	Without replay, without target Q
Breakout	316.8	240.7	10.2	3.2
Enduro	1006.3	831.4	141.9	29.1
River Raid	7446.6	4102.8	2867.7	1453.0
Seaquest	2894.4	822.6	1003.0	275.8
Space Invaders	1088.9	826.3	373.2	302.0

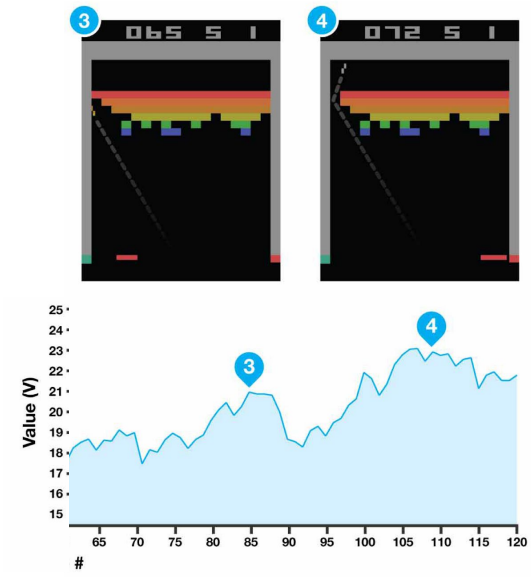
Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." *Nature* 518.7540 (2015): 529-533.

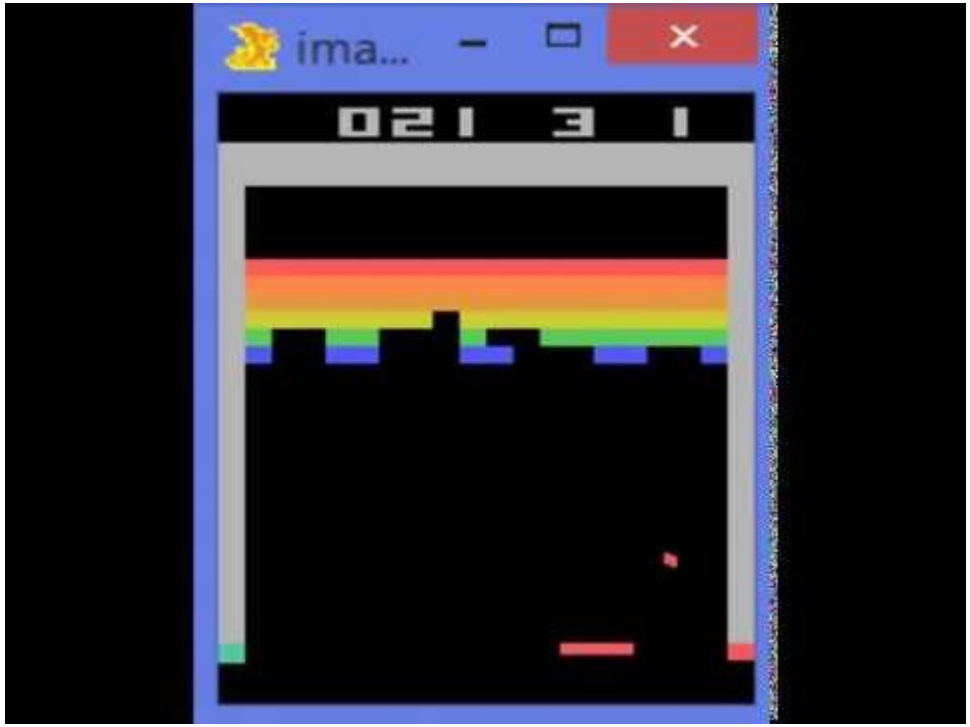
DQN: Results

Why not just use VGGNet features?

Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." *Nature* 518.7540 (2015): 529-533.

DQN: Results





Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." *Nature* 518.7540 (2015): 529-533.

Today’s takeaways

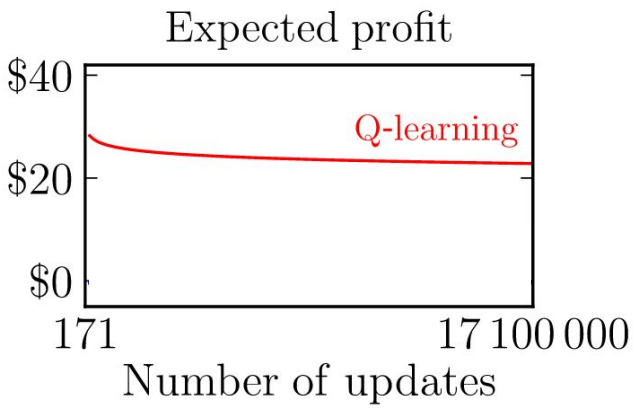
- Bonus RL recap
- Functional Approximation
- Deep Q Network
- **Double Deep Q Network**
- Dueling Networks
- Recurrent DQN
  - Solving “Doom”
- Hierarchical DQN

Q-Learning for Roulette

Hasselt, Hado V. "Double Q-learning." In *Advances in Neural Information Processing Systems*, pp. 2613-2621. 2010.

https://en.wikipedia.org/wiki/File:13-02-27-spielbank-wiesbaden-by-RalfR-094.jpg

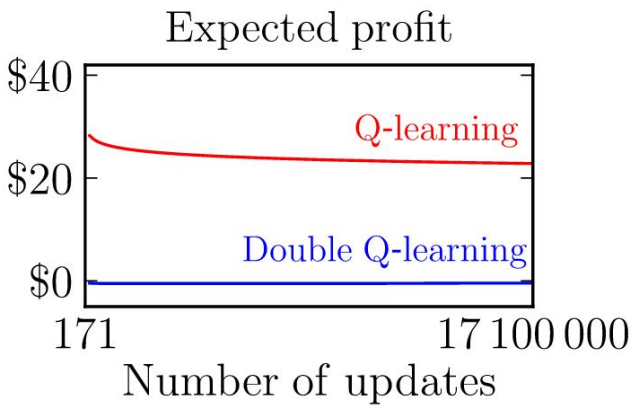
Q-Learning for Roulette



Hasselt, Hado V. "Double Q-learning." In *Advances in Neural Information Processing Systems*, pp. 2613-2621. 2010.

https://en.wikipedia.org/wiki/File:13-02-27-spielbank-wiesbaden-by-RalfR-094.jpg

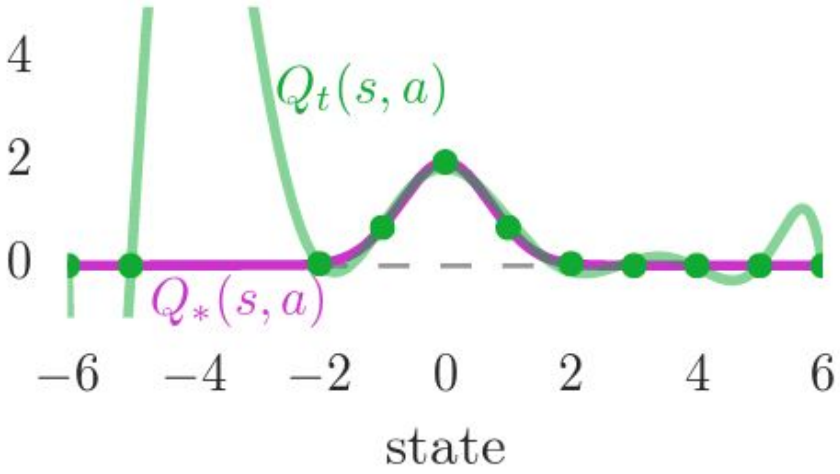
Q-Learning for Roulette



Hasselt, Hado V. "Double Q-learning." In *Advances in Neural Information Processing Systems*, pp. 2613-2621. 2010.

https://en.wikipedia.org/wiki/File:13-02-27-spielbank-wiesbaden-by-RalfR-094.jpg

Q-Learning Overestimation : Function Approximation

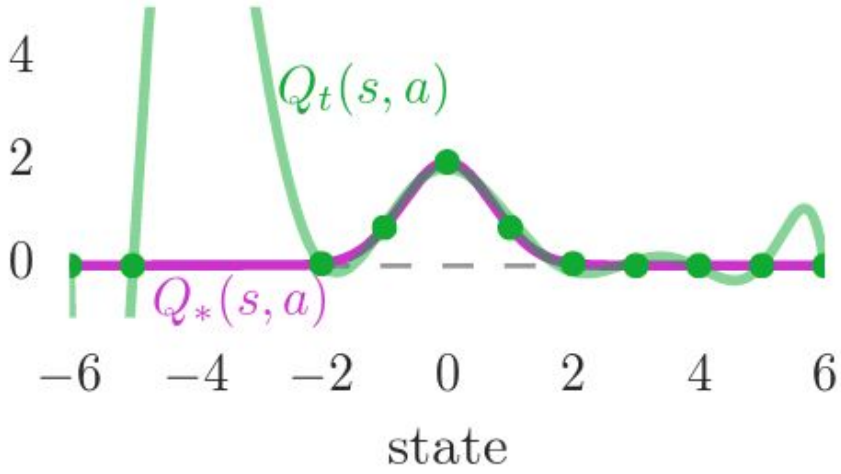


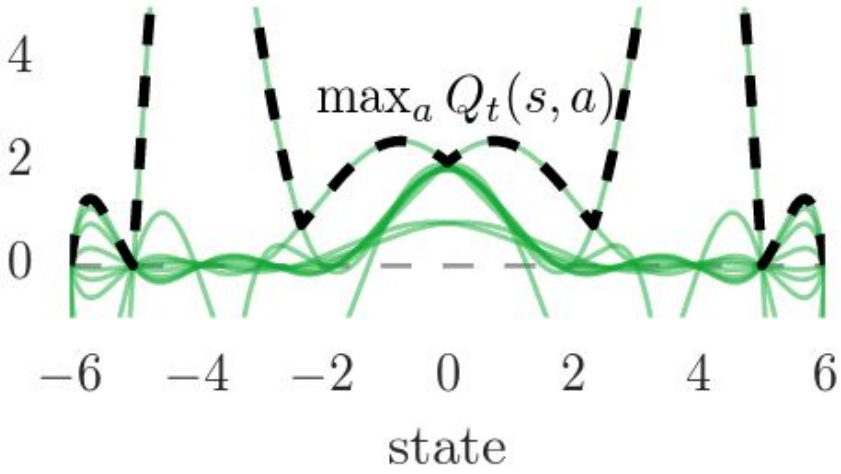
Q Estimate

Actual Q Value

Van Hasselt, Hado, Arthur Guez, and David Silver. "Deep Reinforcement Learning with Double Q-Learning." In *AAAI*, pp. 2094-2100. 2016.

Q-Learning Overestimation : Function Approximation

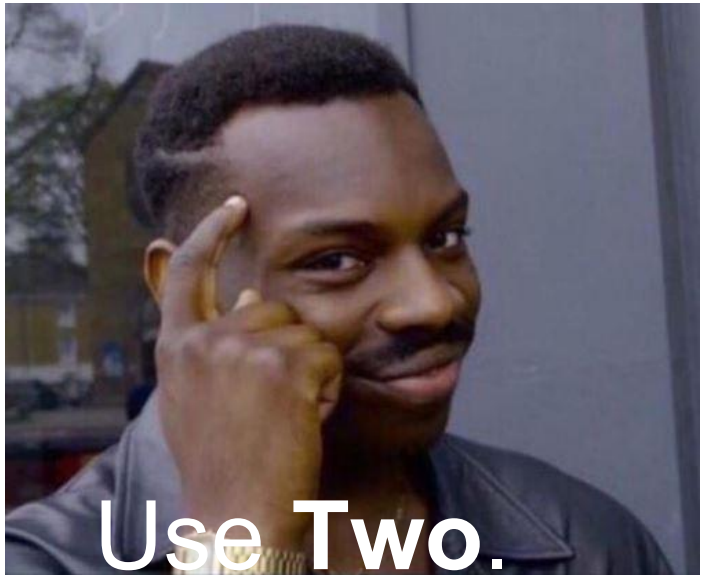




Van Hasselt, Hado, Arthur Guez, and David Silver. "Deep Reinforcement Learning with Double Q-Learning." In *AAAI*, pp. 2094-2100. 2016.

One (Estimator) Isn’t Good Enough?

One (Estimator) Isn’t Good Enough?



https://pbs.twimg.com/media/C5ymV2tVMAYtAev.jpg

Double Q-Learning

- Two estimators:
  - Estimator Q_1 : Obtain best action
  - Estimator Q_2 : Evaluate Q for the above action

- Chances of both estimators overestimating at same action is lesser

Van Hasselt, Hado, Arthur Guez, and David Silver. "Deep Reinforcement Learning with Double Q-Learning." In *AAAI*, pp. 2094-2100. 2016.

Double Q-Learning

- Two estimators:
  - Estimator Q_1 : Obtain best action
  - Estimator Q_2 : Evaluate Q for the above action

$$Q_1(s,a) \leftarrow Q_1(s,a) + \alpha(Target - Q_1(s,a))$$

Q Target

$$R + \max_{a'} Q_1(s',a')$$

Van Hasselt, Hado, Arthur Guez, and David Silver. "Deep Reinforcement Learning with Double Q-Learning." In *AAAI*, pp. 2094-2100. 2016.

Double Q-Learning

- Two estimators:
  - Estimator Q_1 : Obtain best action
  - Estimator Q_2 : Evaluate Q for the above action

$$Q_1(s,a) \leftarrow Q_1(s,a) + \alpha(Target - Q_1(s,a))$$

Q Target

$$R + \max_{a'} Q_1(s',a')$$

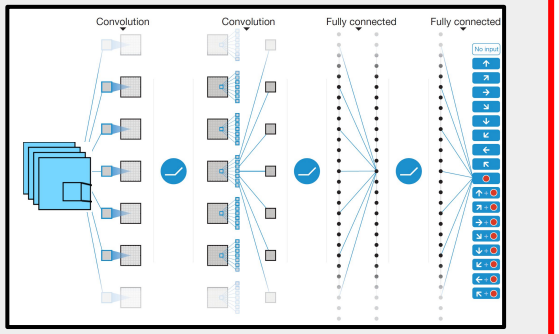
Double Q Target

$$R + Q_2(s', argmax_{a'}(Q_1(s',a')))$$

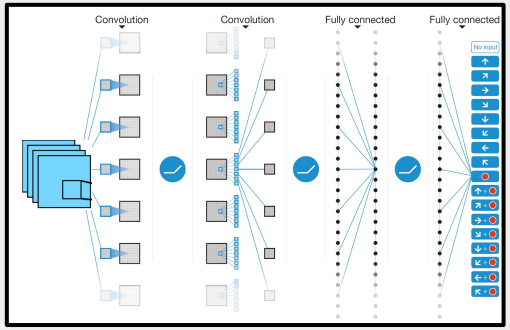
Van Hasselt, Hado, Arthur Guez, and David Silver. "Deep Reinforcement Learning with Double Q-Learning." In *AAAI*, pp. 2094-2100. 2016.

Double Q-Learning

- Two estimators:
  - Estimator Q_1 : Obtain best action
  - Estimator Q_2 : Evaluate Q for the above action



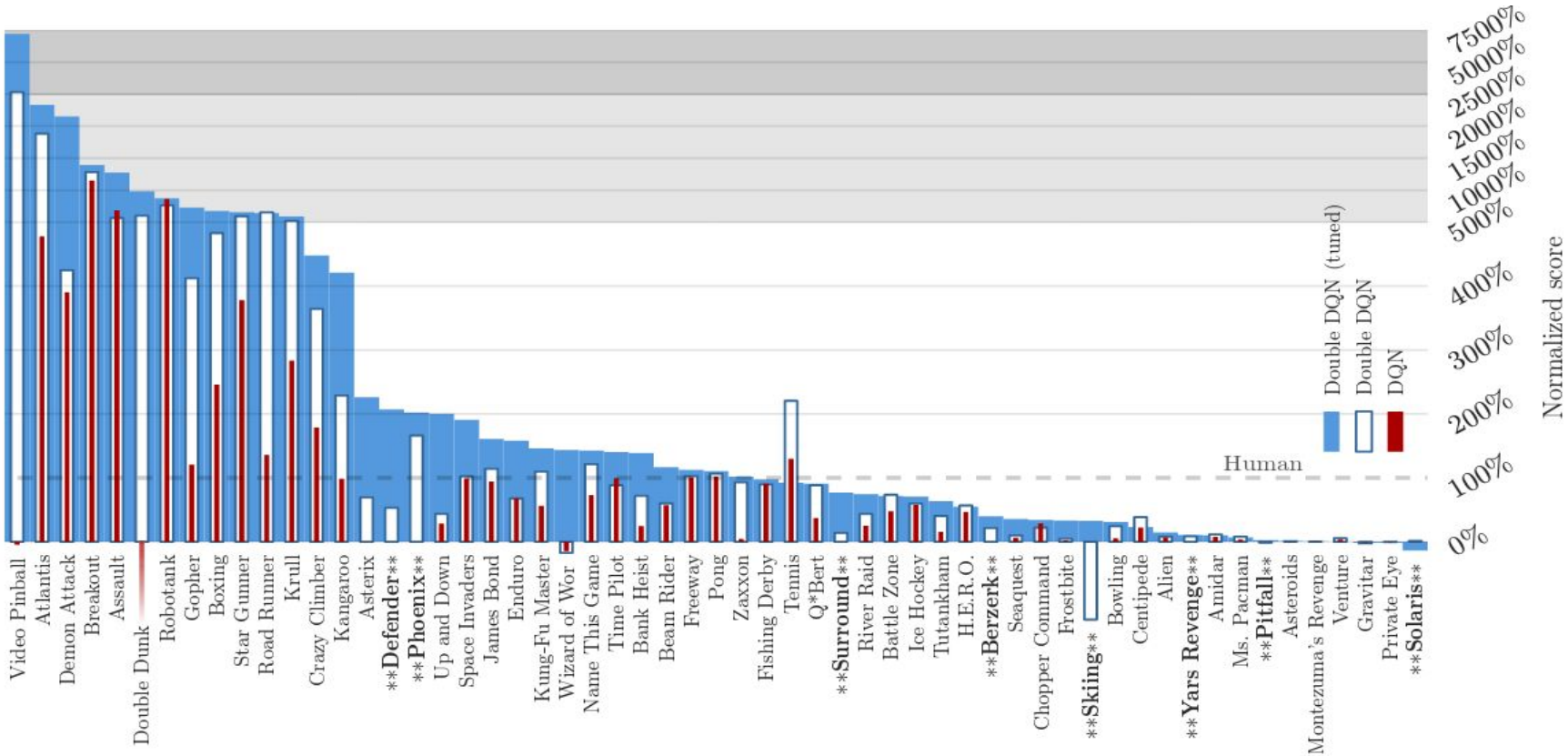
Q1



Q2

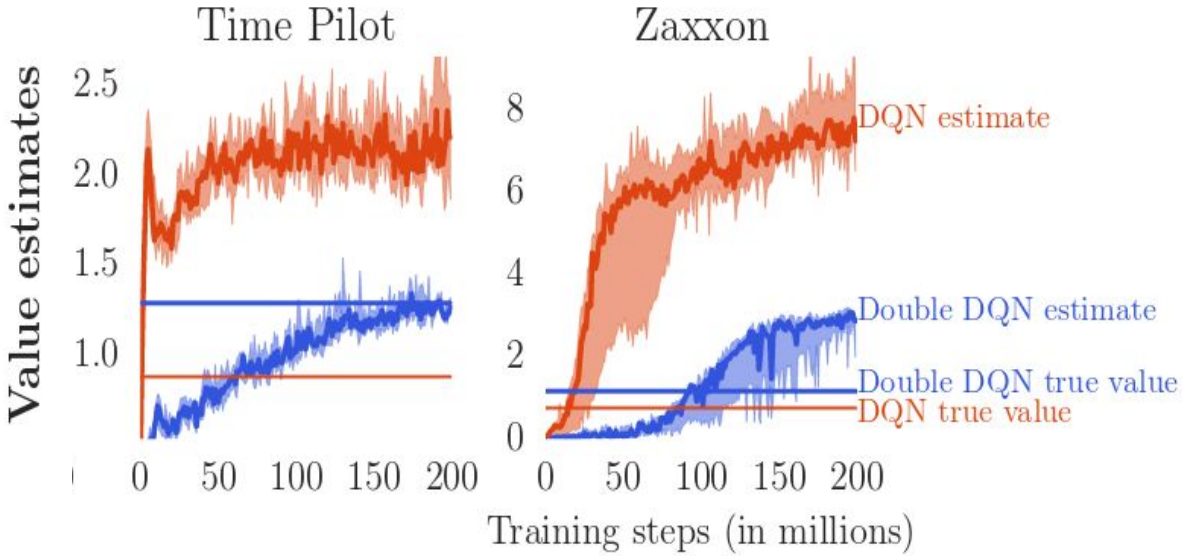
Van Hasselt, Hado, Arthur Guez, and David Silver. "Deep Reinforcement Learning with Double Q-Learning." In *AAAI*, pp. 2094-2100. 2016.

Results - All Atari Games



Van Hasselt, Hado, Arthur Guez, and David Silver. "Deep Reinforcement Learning with Double Q-Learning." In *AAAI*, pp. 2094-2100. 2016.

Results - Solves Overestimations

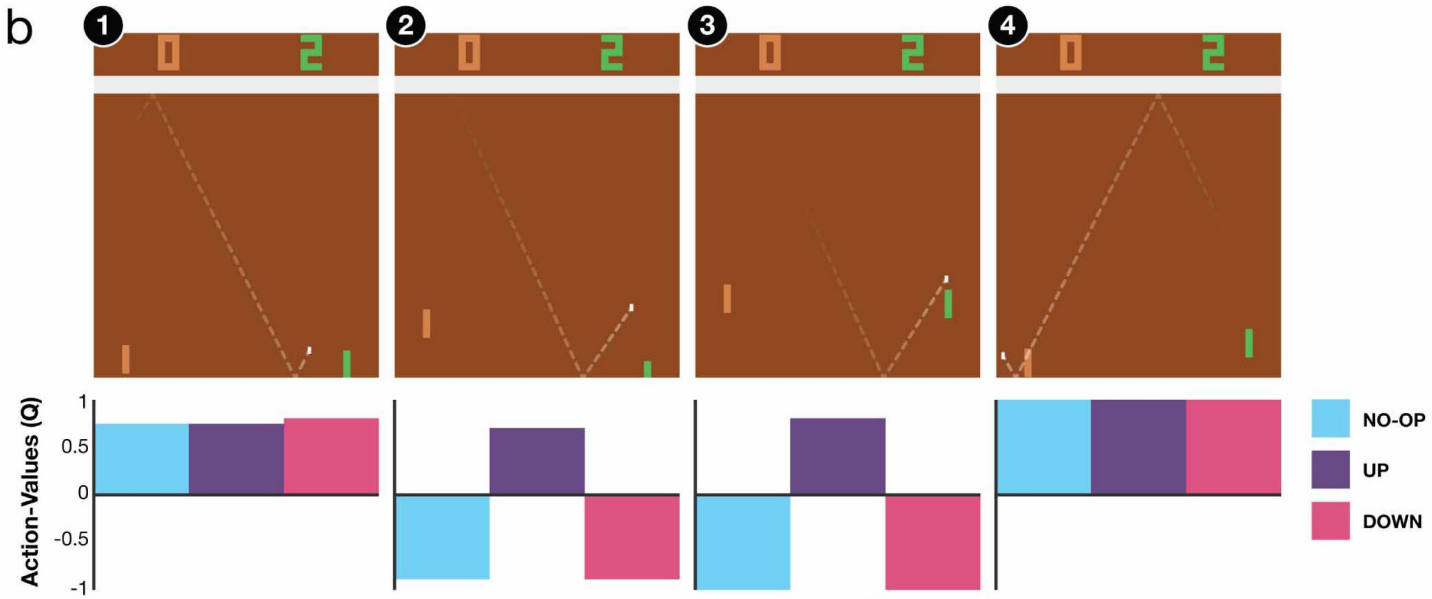


Van Hasselt, Hado, Arthur Guez, and David Silver. "Deep Reinforcement Learning with Double Q-Learning." In *AAAI*, pp. 2094-2100. 2016.

Today’s takeaways

- Bonus RL recap
- Functional Approximation
- Deep Q Network
- Double Deep Q Network
- **Dueling Networks**
- Recurrent DQN
  - Solving “Doom”
- Hierarchical DQN

Pong - Up or Down



Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." *Nature* 518.7540 (2015): 529-533.

Enduro - Left or Right?

http://img.vivaolinux.com.br/imagens/dicas/comunidade/Enduro.png

Enduro - Left or Right?

http://twolivesleft.com/Codea/User/enduro.png

Advantage Function

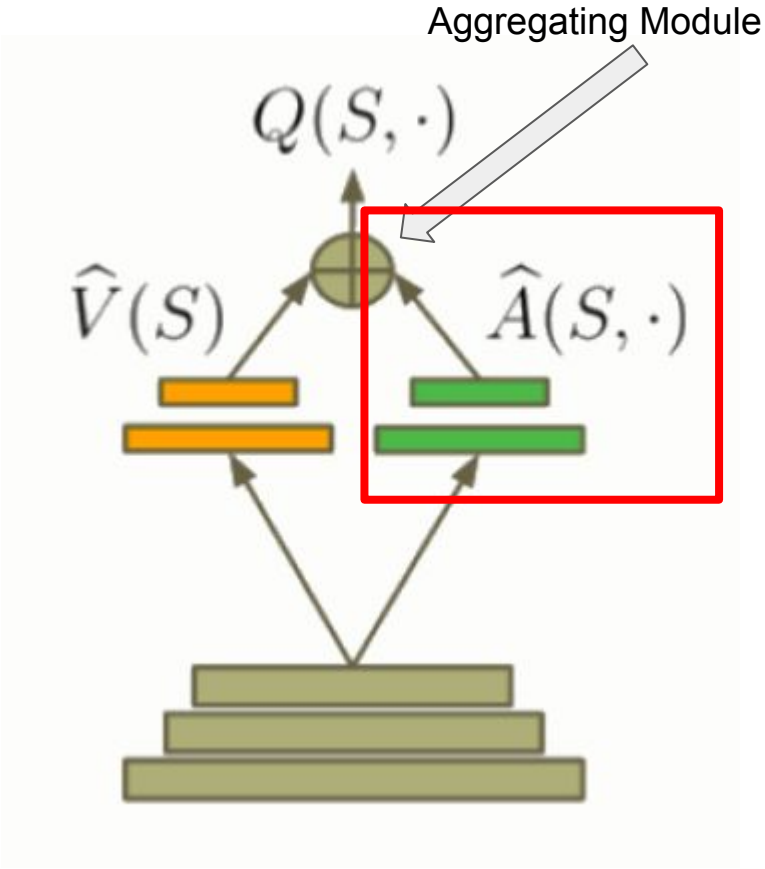
$$Q(s,a) = V(s) + A(s,a)$$

Learning action values ≈ Inherently learning both state values and relative value of the action in that state!

We can use this to help generalize learning for the state values.

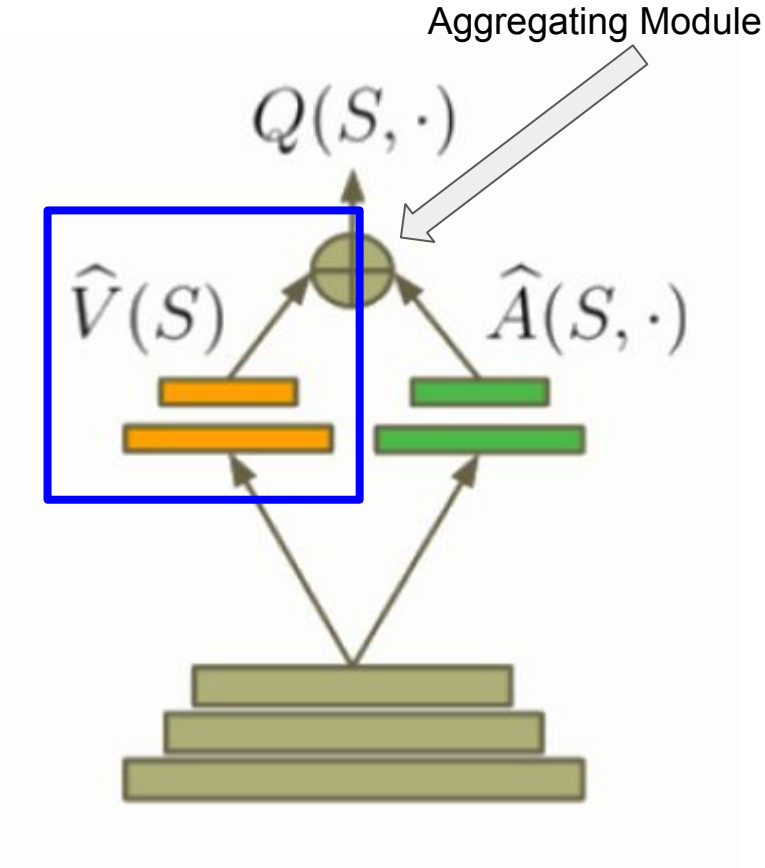
Wang, Ziyu, Tom Schaul, Matteo Hessel, Hado van Hasselt, Marc Lanctot, and Nando de Freitas. "Dueling network architectures for deep reinforcement learning." *arXiv preprint arXiv:1511.06581* (2015).

Dueling Architecture



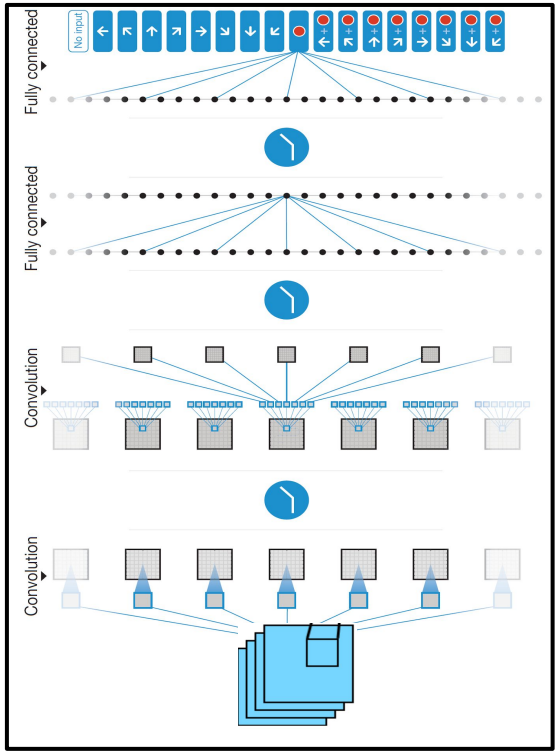
http://torch.ch/blog/2016/04/30/dueling_dqn.html

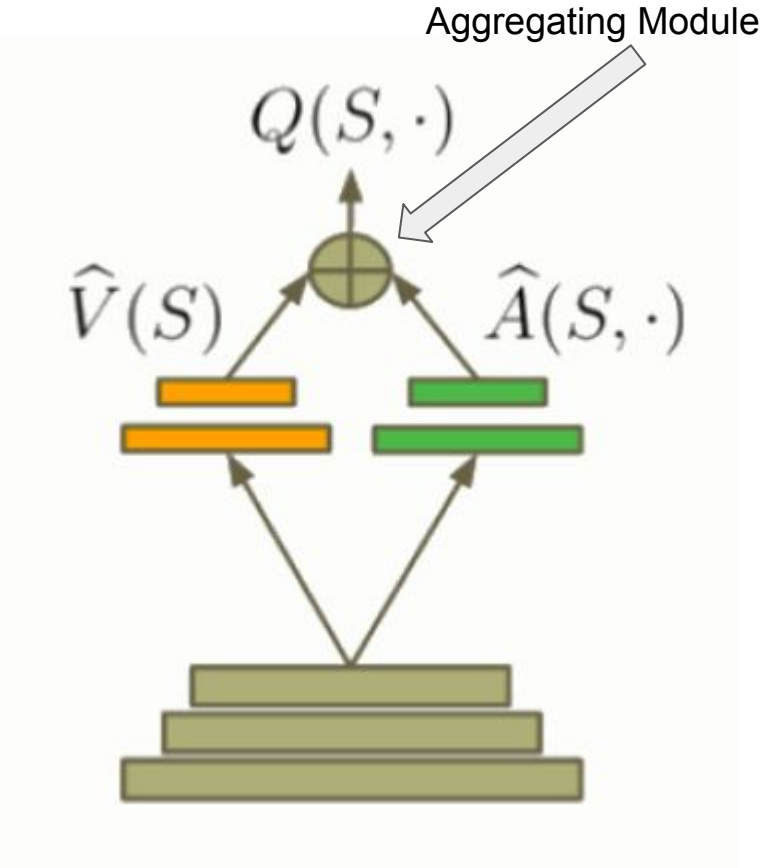
Dueling Architecture



http://torch.ch/blog/2016/04/30/dueling_dqn.html

Dueling Architecture





http://torch.ch/blog/2016/04/30/dueling_dqn.html

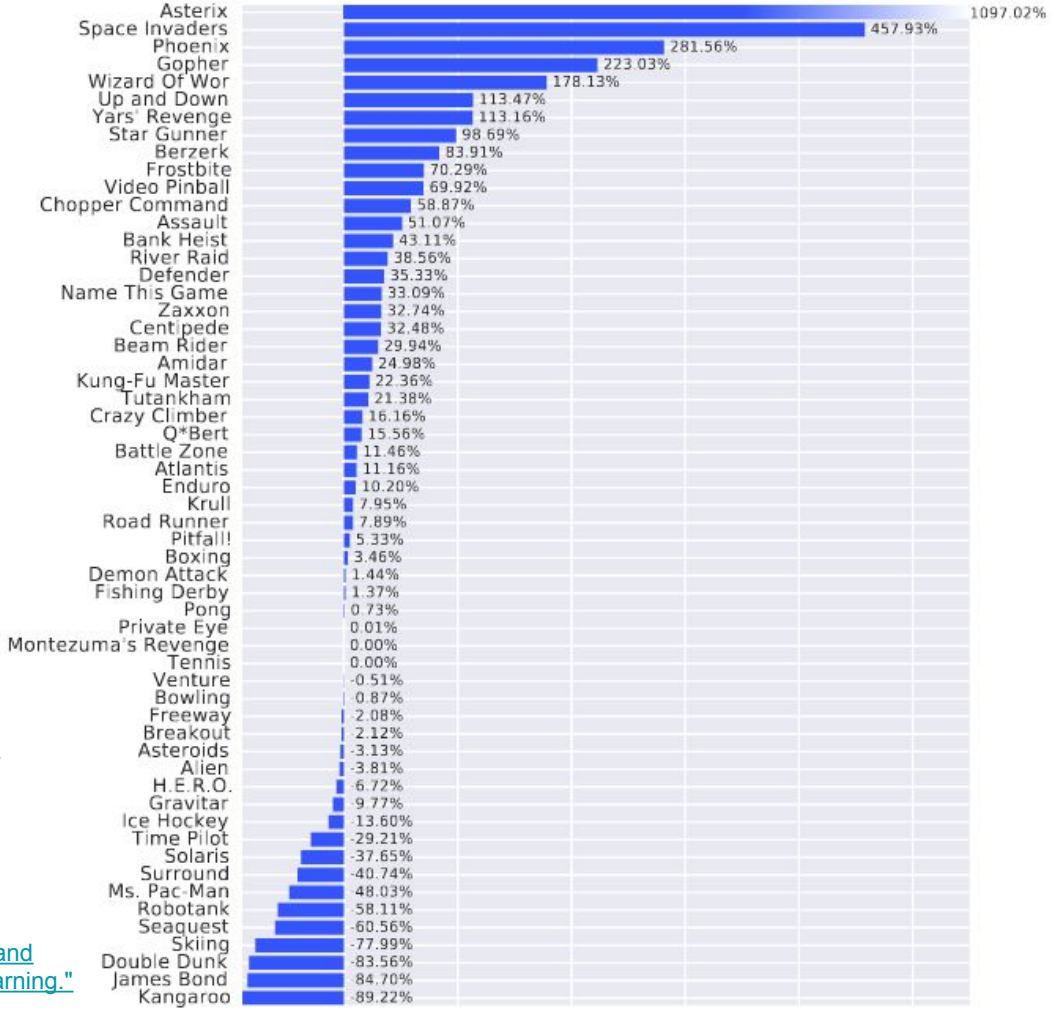
Results

Where does V(s) attend to?

Where does A(s,a) attend to?

Wang, Ziyu, Tom Schaul, Matteo Hessel, Hado van Hasselt, Marc Lanctot, and Nando de Freitas. "Dueling network architectures for deep reinforcement learning." *arXiv preprint arXiv:1511.06581* (2015).

Results



Improvements of dueling architecture over Prioritized DDQN baseline measured by metric above over 57 Atari games

[Wang, Ziyu, Tom Schaul, Matteo Hessel, Hado van Hasselt, Marc Lanctot, and Nando de Freitas. "Dueling network architectures for deep reinforcement learning." *arXiv preprint arXiv:1511.06581* (2015).]

Today’s takeaways

- Bonus RL recap
- Functional Approximation
- Deep Q Network
- Double Deep Q Network
- Dueling Networks
- **Recurrent DQN**
  - Solving “Doom”
- Hierarchical DQN

Moving to more General and Complex Games

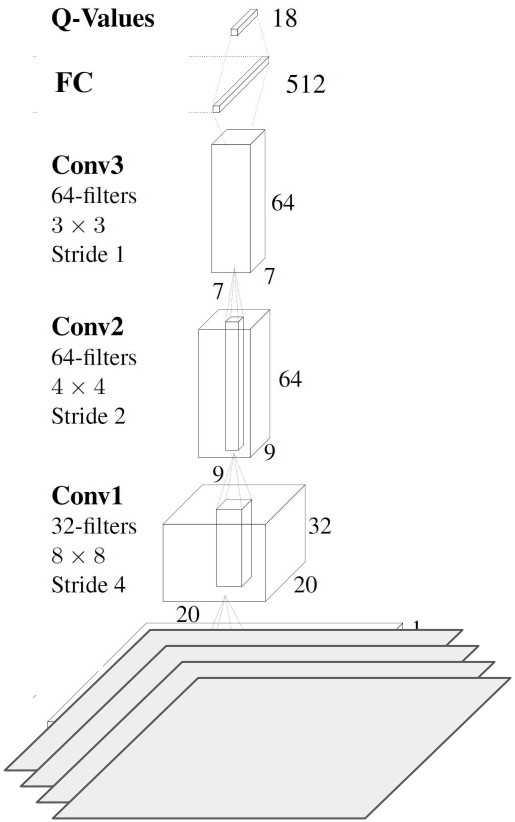
- All games may not be representable using MDPs; some may be POMDPs
  - FPS shooter games
  - Scrabble
  - Even Atari games
- Entire history a solution?

Moving to more General and Complex Games

- All games may not be representable using MDPs; some may be POMDPs
  - FPS shooter games
  - Scrabble
  - Even Atari games

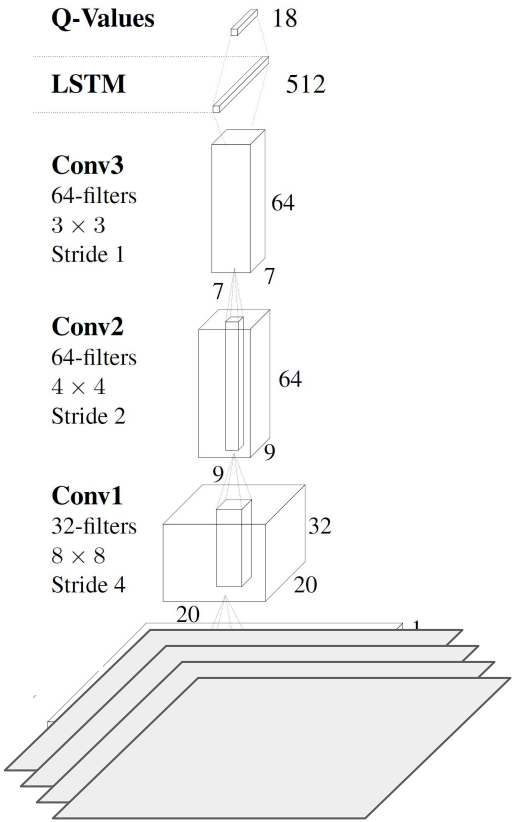
- Entire history a solution?
  - **LSTMs !**

Deep Recurrent Q-Learning



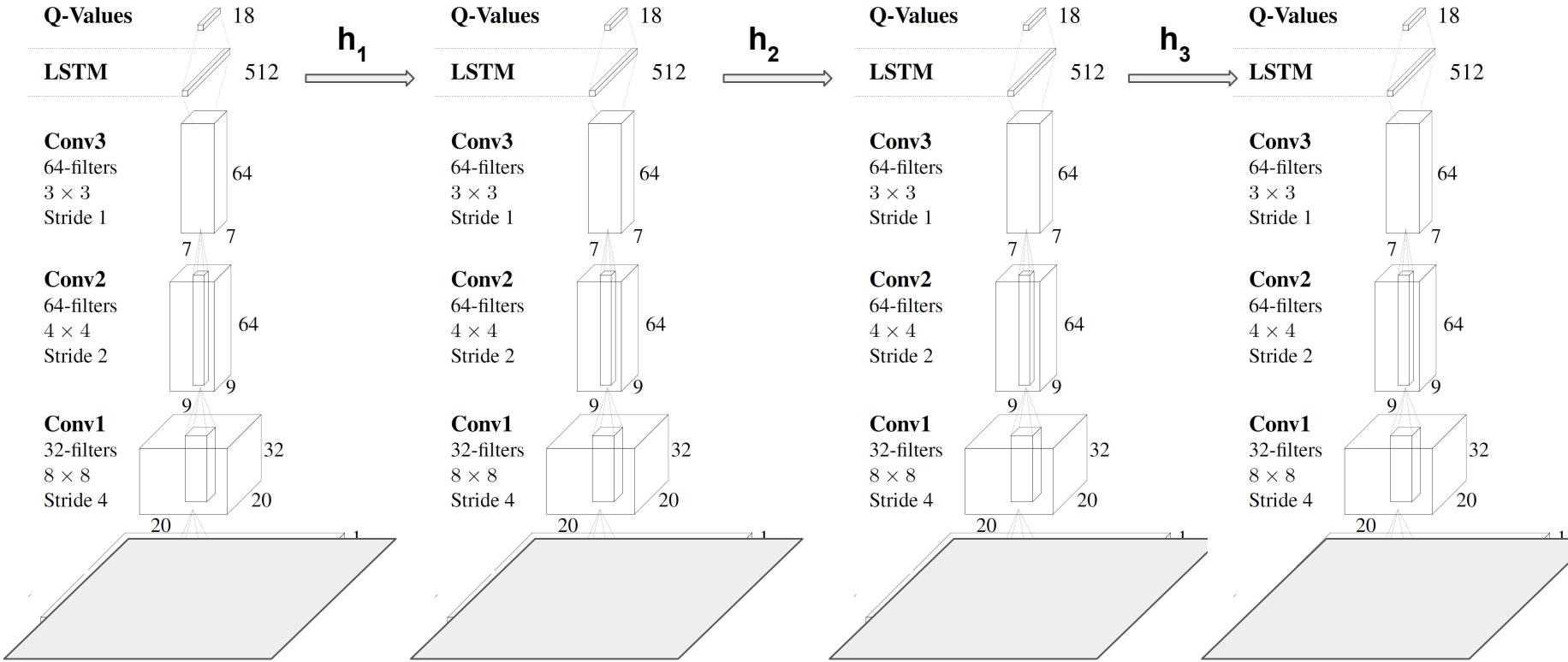
Hausknecht, Matthew, and Peter Stone. "Deep recurrent q-learning for partially observable mdps." *arXiv preprint arXiv:1507.06527* (2015).

Deep Recurrent Q-Learning



Hausknecht, Matthew, and Peter Stone. "Deep recurrent q-learning for partially observable mdps." *arXiv preprint arXiv:1507.06527* (2015).

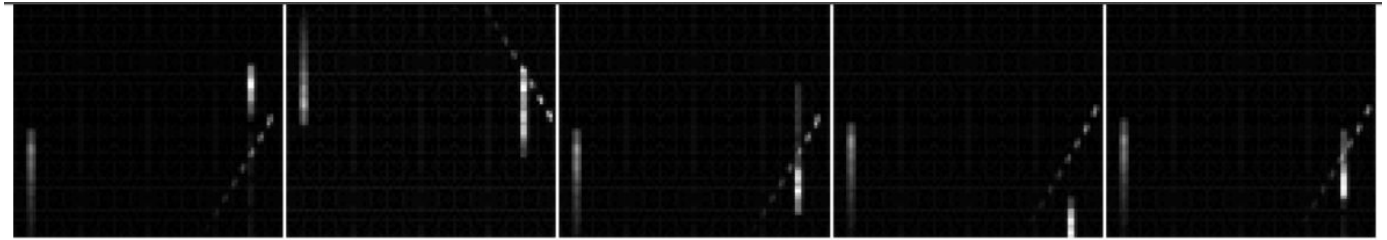
Deep Recurrent Q-Learning



Hausknecht, Matthew, and Peter Stone. "Deep recurrent q-learning for partially observable mdps." *arXiv preprint arXiv:1507.06527* (2015).

DRQN Results

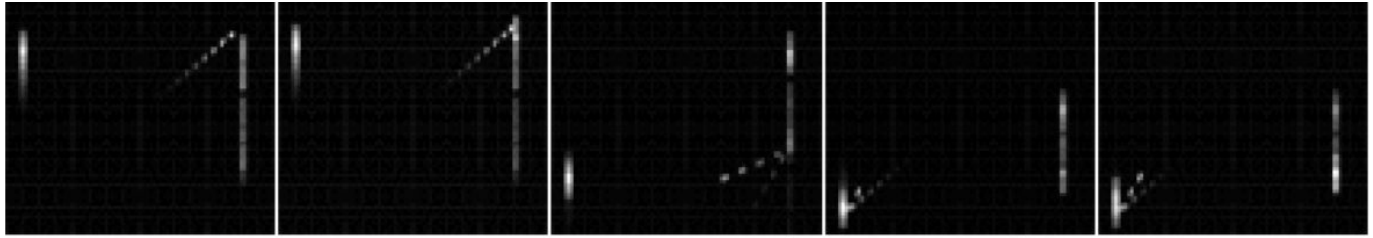
- Misses



Hausknecht, Matthew, and Peter Stone. "Deep recurrent q-learning for partially observable mdps." *arXiv preprint arXiv:1507.06527* (2015).

DRQN Results

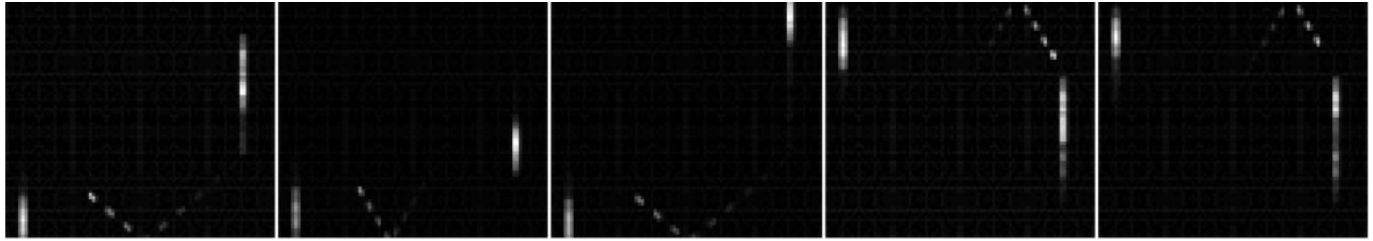
- Paddle Deflection



Hausknecht, Matthew, and Peter Stone. "Deep recurrent q-learning for partially observable mdps." *arXiv preprint arXiv:1507.06527* (2015).

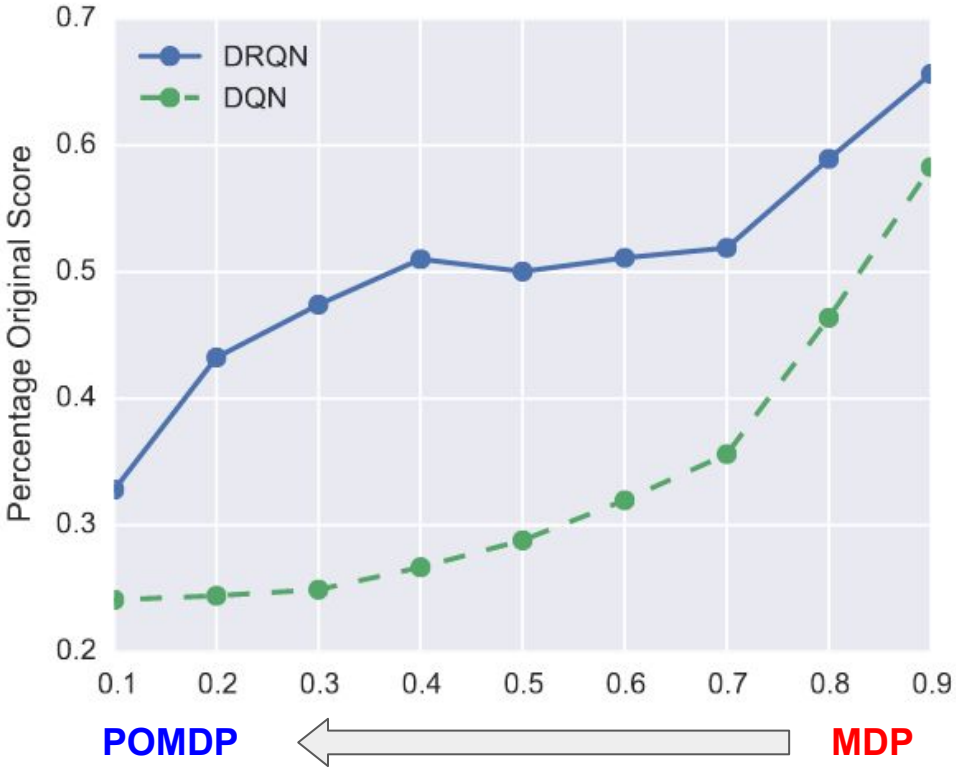
DRQN Results

- Wall Deflections



Hausknecht, Matthew, and Peter Stone. "Deep recurrent q-learning for partially observable mdps." *arXiv preprint arXiv:1507.06527* (2015).

Results - Robustness to partial observability



Hausknecht, Matthew, and Peter Stone. "Deep recurrent q-learning for partially observable mdps." *arXiv preprint arXiv:1507.06527* (2015).

Today’s takeaways

- Bonus RL recap
- Functional Approximation
- Deep Q Network
- Double Deep Q Network
- Dueling Networks
- Recurrent DQN
  - **Solving “Doom”**
- Hierarchical DQN

Application of DRQN: Playing ‘Doom’

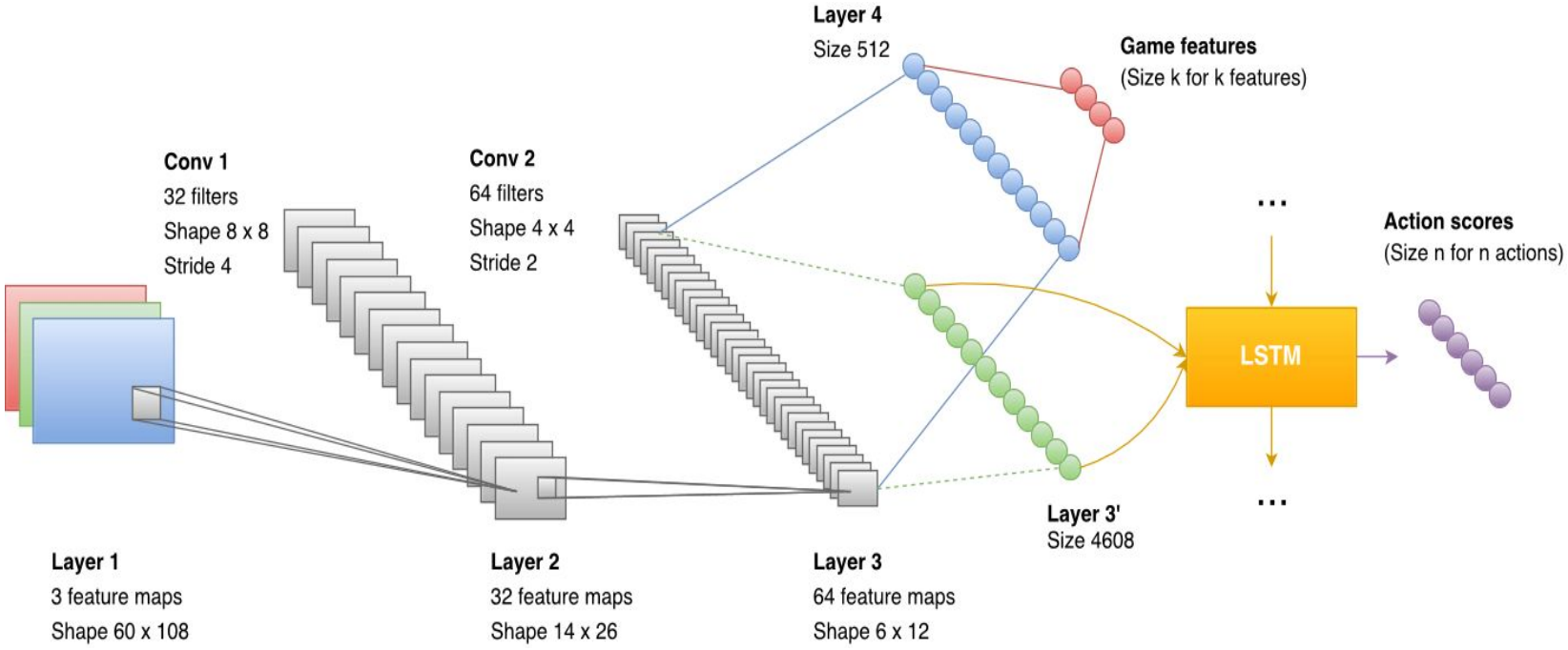
Lample, Guillaume, and Devendra Singh Chaplot. "Playing FPS games with deep reinforcement learning."

Doom Demo

How does DRQN help ?

- Observe o_t instead of s_t
- Limited field of view
- Instead of estimating $Q(s_t, a_t)$, estimate $Q(h_{t,} a_t)$ where $h_t = LSTM(h_{t-1}, o_t)$

Architecture: Comparison with Baseline DRQN



Training Tricks

- Jointly training DRQN model and game feature detection

Training Tricks

- Jointly training DRQN model and game feature detection

What do you think is the advantage of this ?

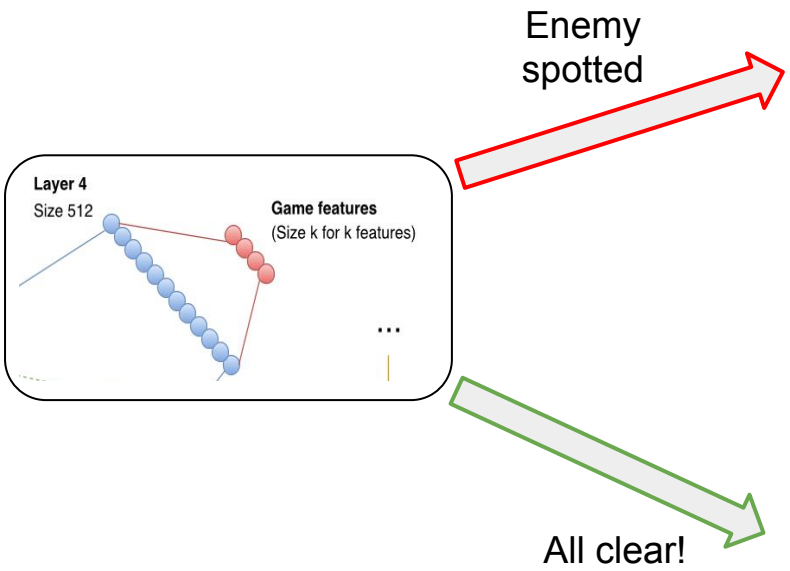
Training Tricks

- Jointly training DRQN model and game feature detection

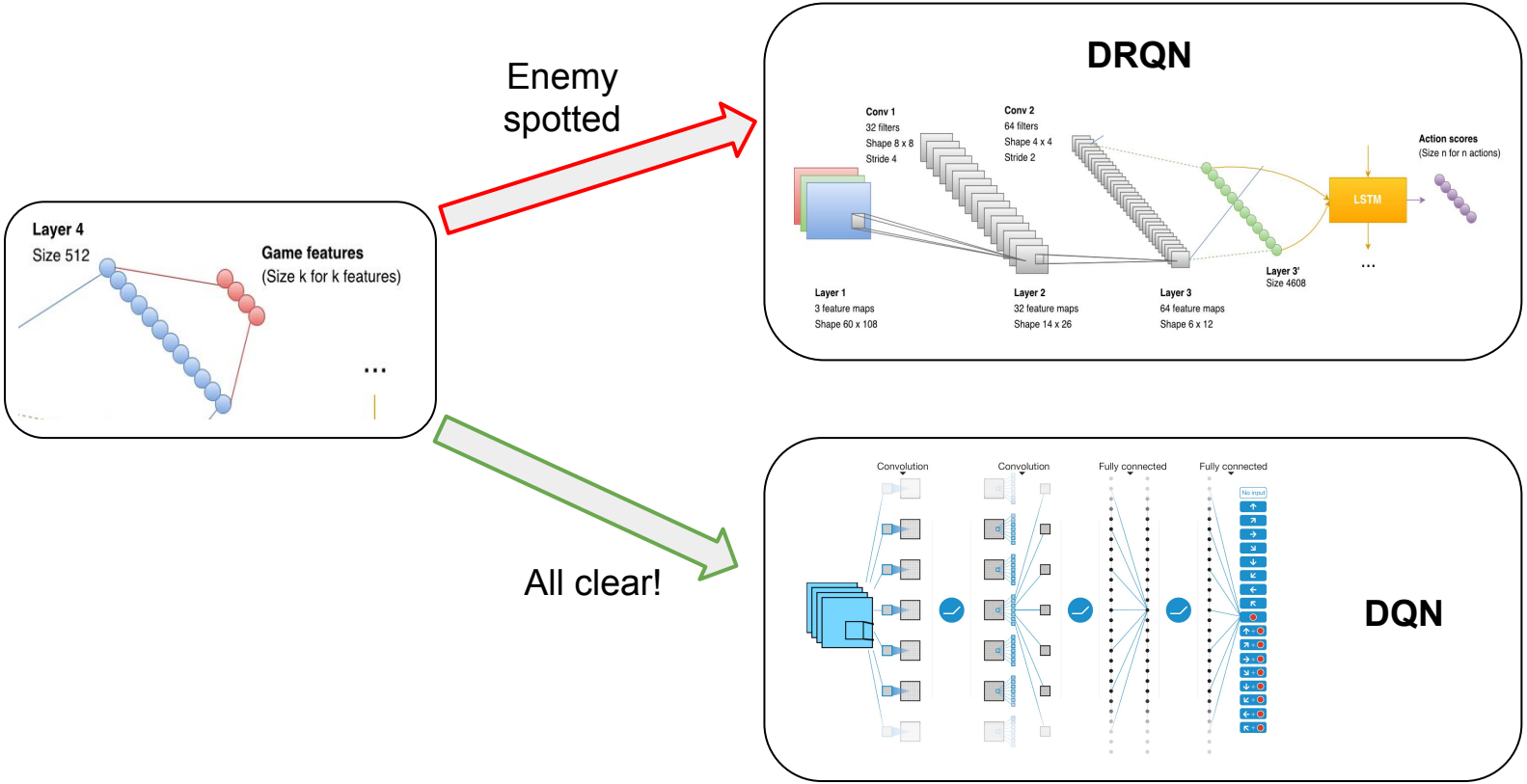
What do you think is the advantage of this ?

- CNN layers capture relevant information about features of the game that maximise action value scores

Modular Architecture

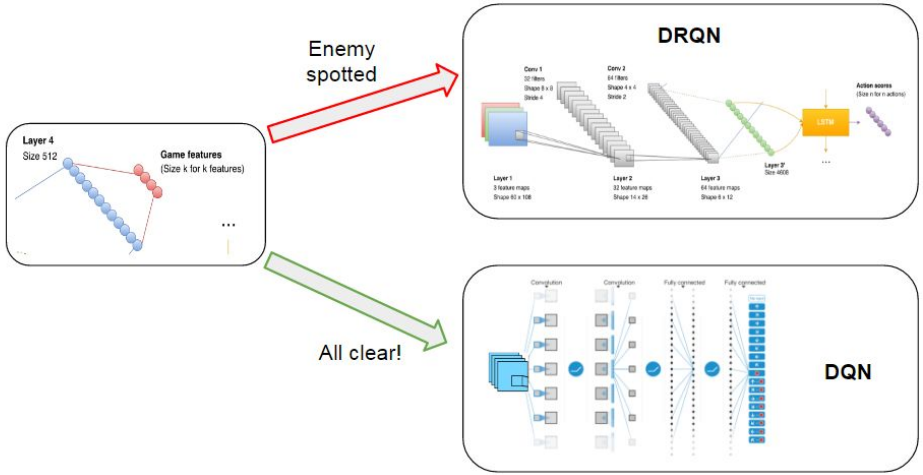


Modular Architecture



Modular Network : Advantages

- Can be trained and tested independently
- Both can be trained in parallel
- Reduces the state-action pairs space : Faster Training
- Mitigates camper behavior : “Tendency to stay in one area of the map and wait for enemies”



Rewards Formulation for DOOM

What do you think ?

Rewards Formulation for DOOM

What do you think ?

- Positive rewards for Kills and Negative rewards for suicides
- Small Intermediate Rewards :
  - Positive Reward for object pickup
  - Negative Reward for losing health
  - Negative Reward for shooting or losing ammo
  - Small Positive Rewards proportional to the distance travelled since last step (Agent avoids running in circles)

Performance with Separate Navigation Network

	Limited Deathmatch		Full Deathmatch			
	Known Map		Train maps		Test maps	
Evaluation Metric	Without navigation	With navigation	Without navigation	With navigation	Without navigation	With navigation
Number of objects	14	46	52.9	92.2	62.3	94.7
Number of kills	167	138	43.0	66.8	32.0	43.0
Number of deaths	36	25	15.2	14.6	10.0	6.0
Number of suicides	15	10	1.7	3.1	0.3	1.3
Kill to Death Ratio	4.64	5.52	2.83	4.58	3.12	6.94

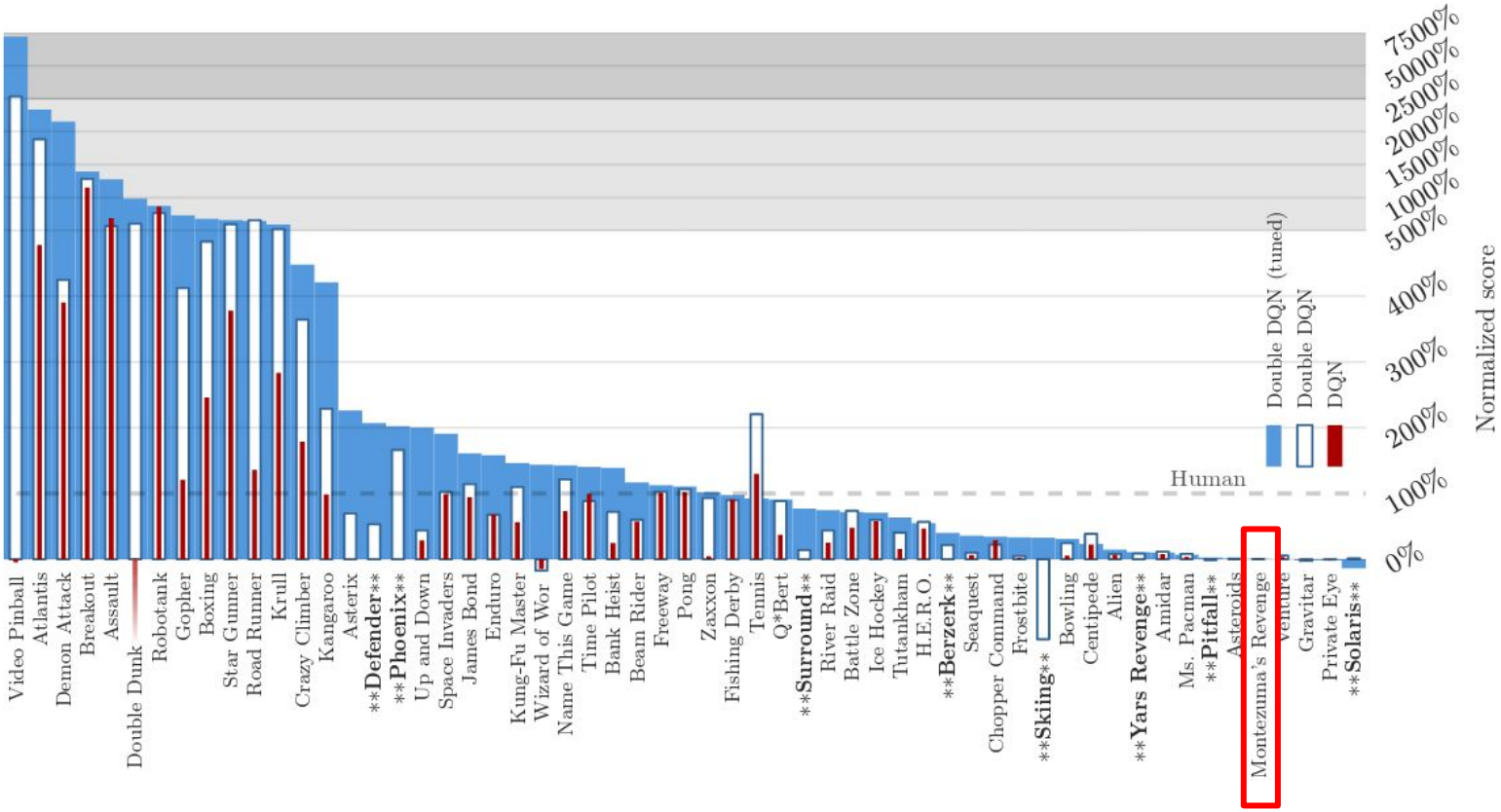
Results

	Single Player		Multiplayer	
Evaluation Metric	Human	Agent	Human	Agent
Number of objects	5.2	9.2	6.1	10.5
Number of kills	12.6	27.6	5.5	8.0
Number of deaths	8.3	5.0	11.2	6.0
Number of suicides	3.6	2.0	3.2	0.5
K/D Ratio	1.52	5.12	0.49	1.33

Today’s takeaways

- Bonus RL recap
- Functional Approximation
- Deep Q Network
- Double Deep Q Network
- Dueling Networks
- Recurrent DQN
  - Solving “Doom”
- **Hierarchical DQN**

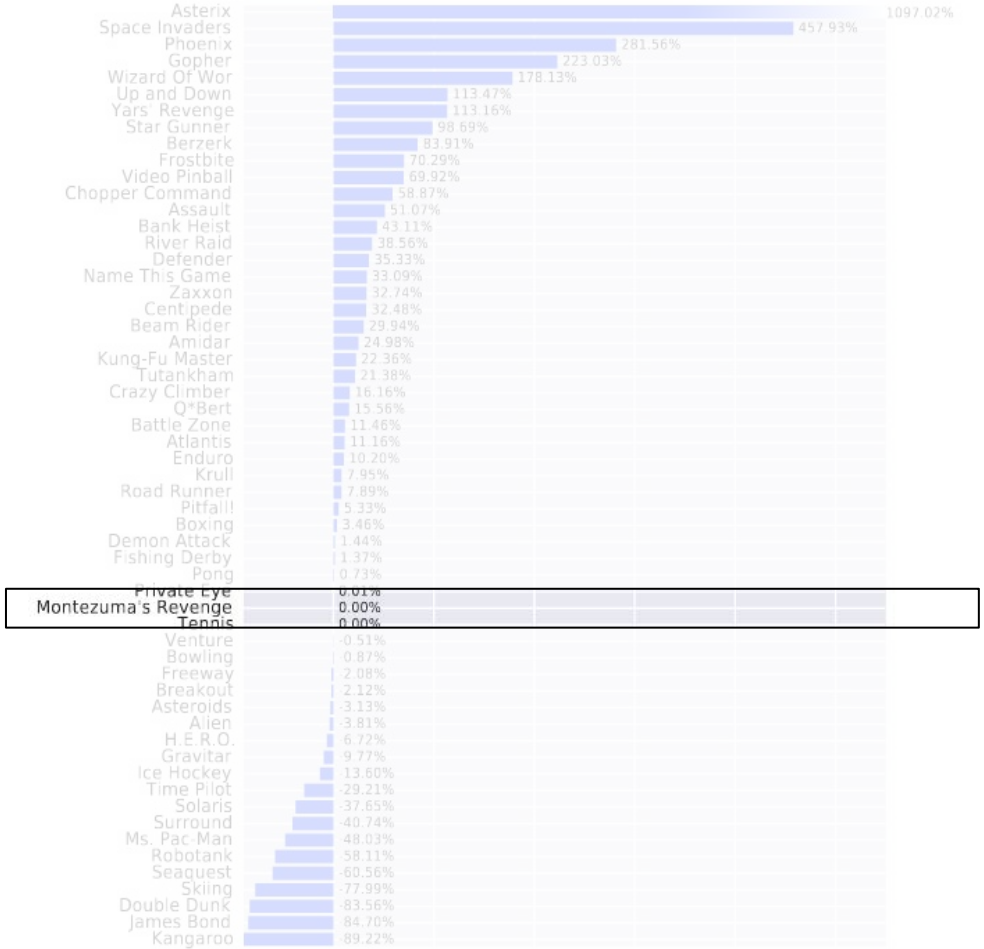
h-DQN



Double DQN

Van Hasselt, Hado, Arthur Guez, and David Silver. "Deep Reinforcement Learning with Double Q-Learning." In *AAAI*, pp. 2094-2100. 2016.

Dueling Networks

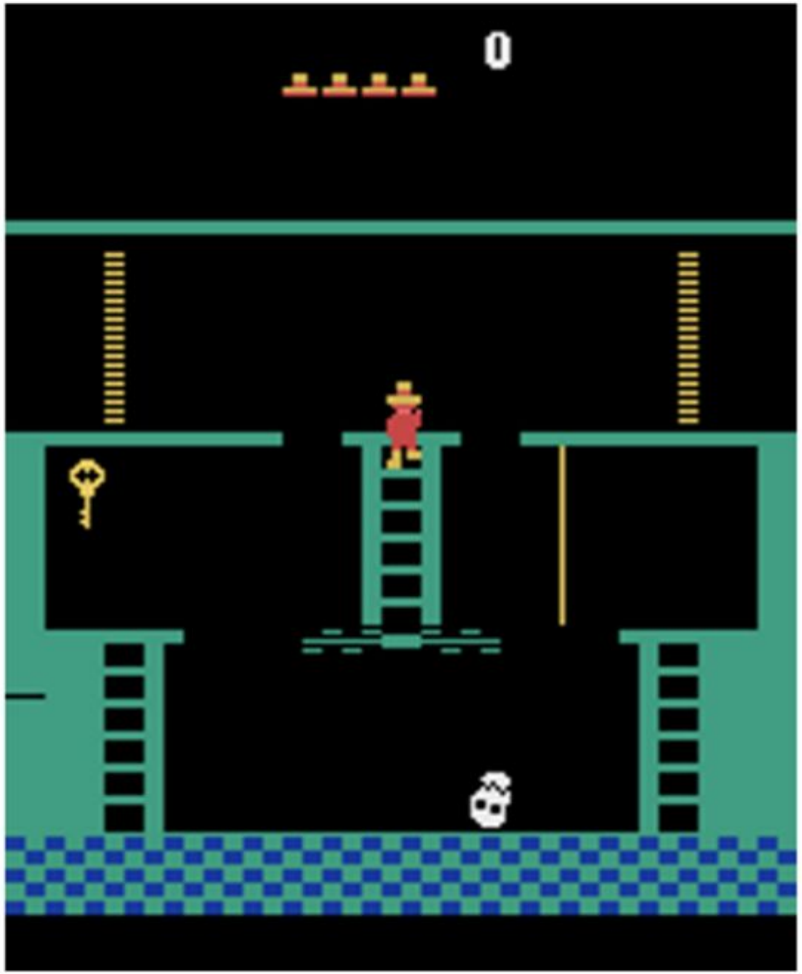


Van Hasselt, Hado, Arthur Guez, and David Silver. "Deep Reinforcement Learning with Double Q-Learning." In *AAAI*, pp. 2094-2100. 2016.

How is this game different ?

- Complex Game Environment
- Sparse and Longer Range Delayed Rewards

- **Insufficient Exploration : We need temporally extended exploration**



Kulkarni, Tejas D., Karthik Narasimhan, Ardavan Saeedi and Joshua B. Tenenbaum. "Hierarchical Deep Reinforcement Learning: Integrating Temporal Abstraction and Intrinsic Motivation." NIPS 2016

How is this game different ?

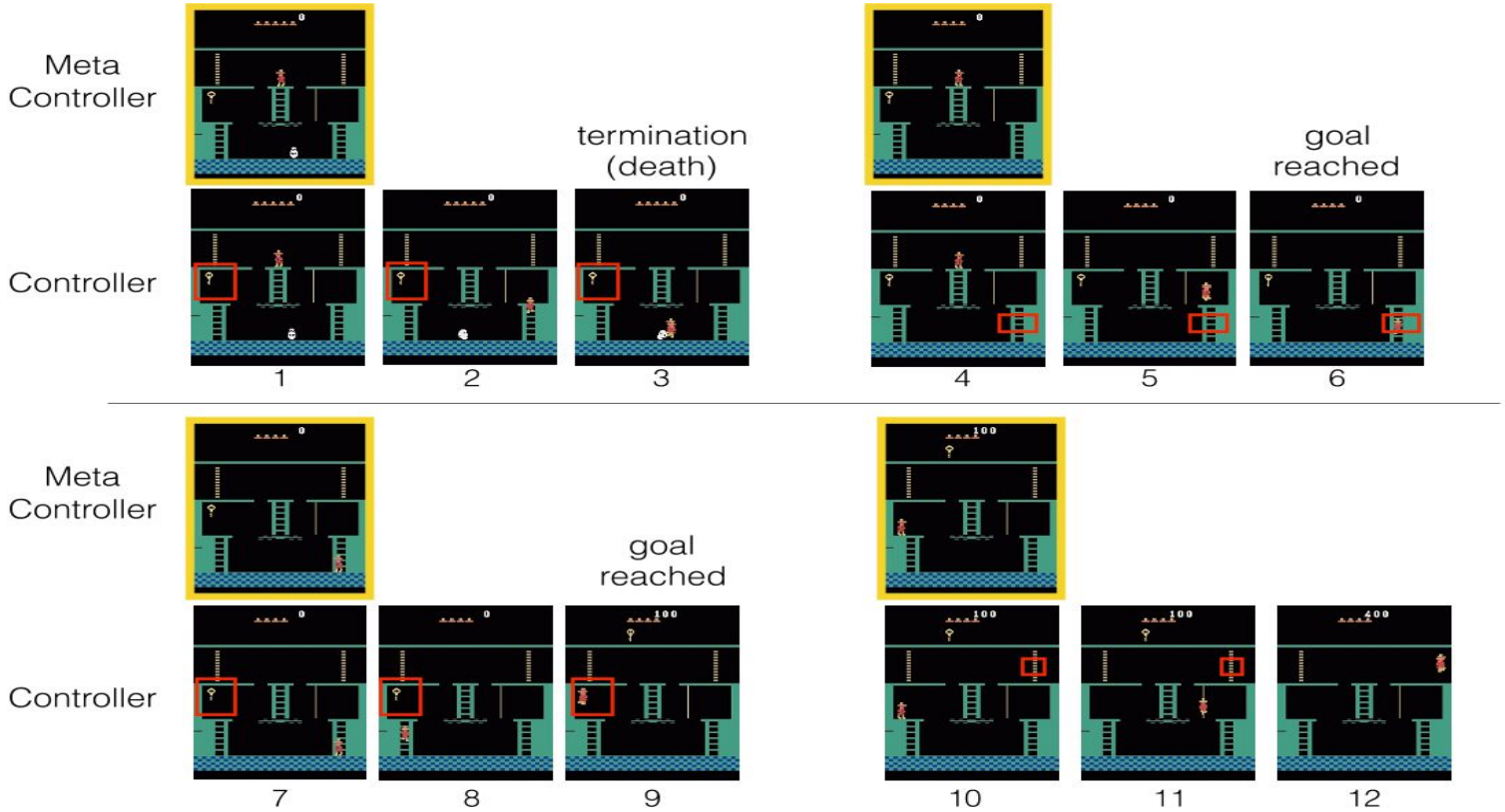
- Complex Game Environment
- Sparse and Longer Range Delayed Rewards

- **Insufficient Exploration : We need temporally extended exploration**

↓

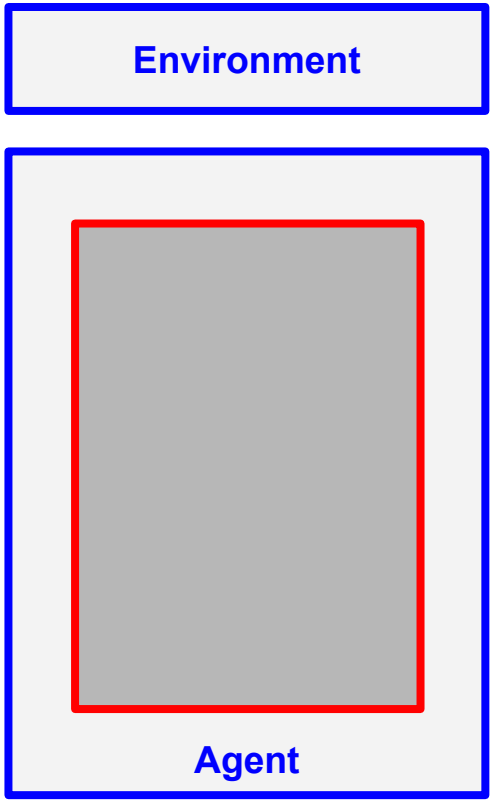
Dividing Extrinsic Goal into Hierarchical Intrinsic Subgoals

Intrinsic Goals in Montezuma's Revenge

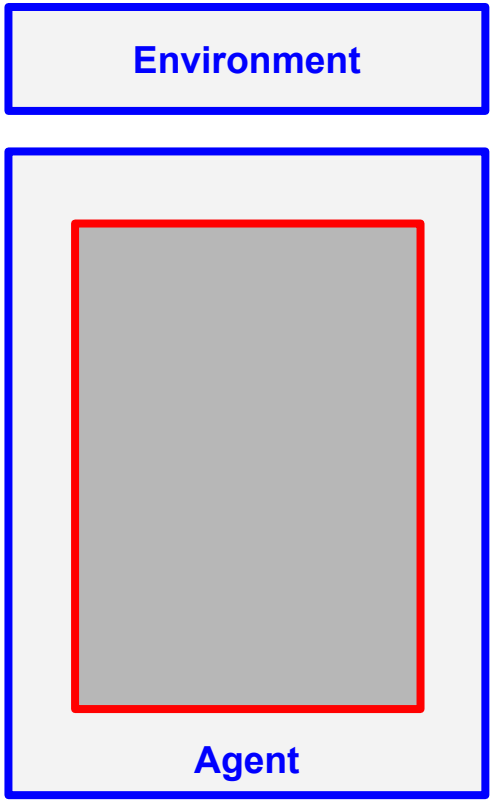


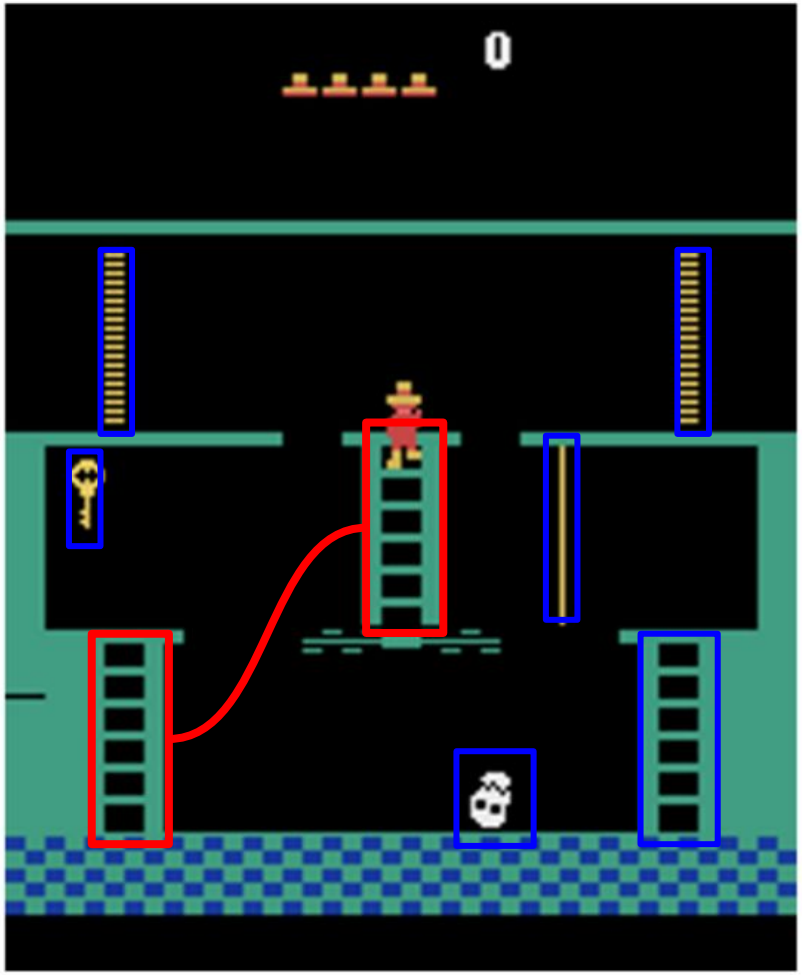
Kulkarni, Tejas D., Karthik Narasimhan, Ardavan Saeedi and Joshua B. Tenenbaum. "Hierarchical Deep Reinforcement Learning: Integrating Temporal Abstraction and Intrinsic Motivation." NIPS 2016

Hierarchy of DQNs

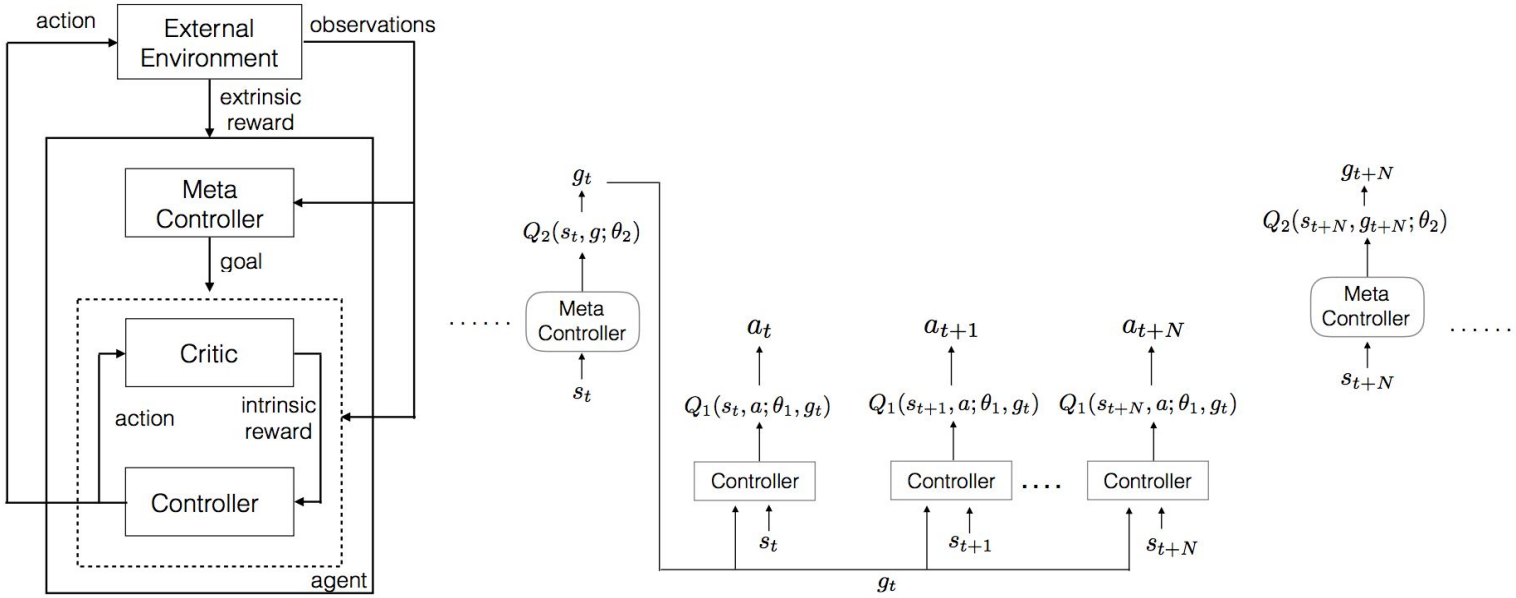


Hierarchy of DQNs





Architecture Block for h-DQN



Kulkarni, Tejas D., Karthik Narasimhan, Ardavan Saeedi and Joshua B. Tenenbaum. "Hierarchical Deep Reinforcement Learning: Integrating Temporal Abstraction and Intrinsic Motivation." NIPS 2016

h-DQN Learning Framework (1)

- V(s,g) : Value function of a state for achieving the given goal $g \in G$
- Option :
  - A multi-step action policy to achieve these intrinsic goals $g \in G$
  - Can also be primitive actions
- Π_g : Policy Over Options to achieve goal g
- Agents learns
  - which Intrinsic goals are important
  - Π_g
  - correct sequence of such policies Π_g

Kulkarni, Tejas D., Karthik Narasimhan, Ardavan Saeedi and Joshua B. Tenenbaum. "Hierarchical Deep Reinforcement Learning: Integrating Temporal Abstraction and Intrinsic Motivation." NIPS 2016

h-DQN Learning Framework (2)

Objective Function for Meta-Controller :

- Maximise Cumulative Extrinsic Reward

$$F_t = \sum_{t'=t}^{\infty} \gamma^{t'-t} f_{t'}$$

Objective Function for Controller :

- Maximise Cumulative Intrinsic Reward

$$R_t(g) = \sum_{t'=t}^{\infty} \gamma^{t'-t} r_{t'}(g)$$

Kulkarni, Tejas D., Karthik Narasimhan, Ardavan Saeedi and Joshua B. Tenenbaum. "Hierarchical Deep Reinforcement Learning: Integrating Temporal Abstraction and Intrinsic Motivation." NIPS 2016

Training

- Two disjoint memories D_1 and D_2 for Experience Replay
- Experiences (s_t, g_t, f_t, s_{t+N}) for Q_2 are stored in D_2
- Experiences $(s_t, a_t, g_t, r_t, s_{t+1})$ for Q_1 are stored in D_1
- Different time scales
  - Transitions from Controller (Q_1) are picked at every time step
  - Transitions from Meta-Controller (Q_2) are picked only when controller terminates on reaching the intrinsic goal or epsiode ends

Kulkarni, Tejas D., Karthik Narasimhan, Ardavan Saeedi and Joshua B. Tenenbaum. "Hierarchical Deep Reinforcement Learning: Integrating Temporal Abstraction and Intrinsic Motivation." NIPS 2016

Results :

Kulkarni, Tejas D., Karthik Narasimhan, Ardavan Saeedi and Joshua B. Tenenbaum. "Hierarchical Deep Reinforcement Learning: Integrating Temporal Abstraction and Intrinsic Motivation." NIPS 2016

Today’s takeaways

- Bonus RL recap
- Functional Approximation
- Deep Q Network
- Double Deep Q Network
- Dueling Networks
- Recurrent DQN
  - Solving “Doom”
- Hierarchical DQN

References

Basic RL

- David Silver's Introduction to RL lectures
- Peter Abbeel's Artificial Intelligence - Berkeley (Spring 2015)

DQN

- Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." *Nature* 518.7540 (2015): 529-533.
- Mnih, Volodymyr, et al. "Playing atari with deep reinforcement learning." *arXiv preprint arXiv:1312.5602* (2013).

DDQN

- Hasselt, Hado V. "Double Q-learning." In *Advances in Neural Information Processing Systems*, pp. 2613-2621. 2010.
- Van Hasselt, Hado, Arthur Guez, and David Silver. "Deep Reinforcement Learning with Double Q-Learning." In *AAAI*, pp. 2094-2100. 2016.

Dueling DQN

- Wang, Ziyu, Tom Schaul, Matteo Hessel, Hado van Hasselt, Marc Lanctot, and Nando de Freitas. "Dueling network architectures for deep reinforcement learning." *arXiv preprint arXiv:1511.06581* (2015).

References

DRQN

- Hausknecht, Matthew, and Peter Stone. "Deep recurrent q-learning for partially observable mdps." *arXiv preprint arXiv:1507.06527* (2015).

Doom

- Lample, Guillaume, and Devendra Singh Chaplot. "Playing FPS games with deep reinforcement learning."

h-DQN

- Kulkarni, Tejas D., Karthik Narasimhan, Ardavan Saeedi and Joshua B. Tenenbaum. “Hierarchical Deep Reinforcement Learning: Integrating Temporal Abstraction and Intrinsic Motivation.” NIPS 2016

Additional NLP/Vision applications

- Narasimhan, Karthik, Tejas Kulkarni, and Regina Barzilay. "Language understanding for text-based games using deep reinforcement learning." EMNLP 20155
- Caicedo, Juan C., and Svetlana Lazebnik. "Active object localization with deep reinforcement learning." *Proceedings of the IEEE International Conference on Computer Vision*. 2015.
- Zhu, Yuke, et al. "Target-driven visual navigation in indoor scenes using deep reinforcement learning." *arXiv preprint arXiv:1609.05143* (2016).

Deep Q Learning for text-based games

State 1: The old bridge
You are standing very close to the bridge's eastern foundation. If you go east you will be back on solid ground ... The bridge sways in the wind.

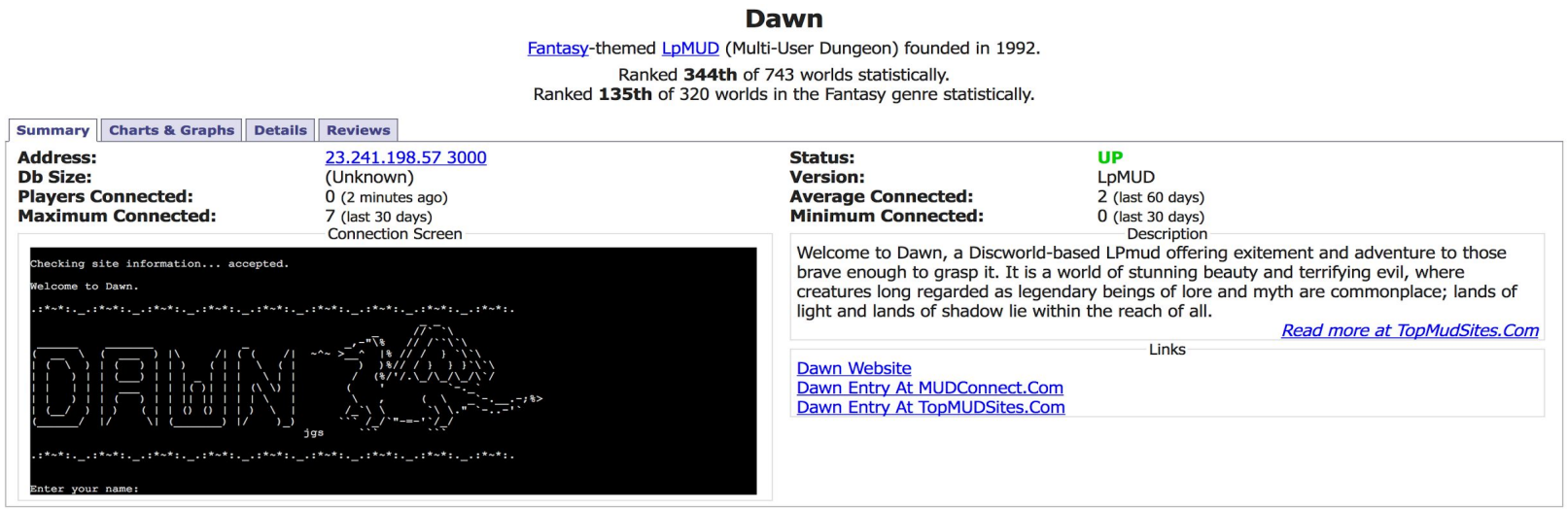
Command: **Go east**

State 2: Ruined gatehouse
The old gatehouse is near collapse. Part of its northern wall has already fallen down ... East of the gatehouse leads out to a small open area surrounded by the remains of the castle. There is also a standing archway offering passage to a path along the old southern inner wall.
Exits: Standing archway, castle corner, Bridge over the abyss

Narasimhan, Karthik, Tejas Kulkarni, and Regina Barzilay. "Language understanding for text-based games using deep reinforcement learning." EMNLP 2015

Text Based Games : Back in 1970’s

- Predecessors to Modern Graphical Games
- MUD (Multi User Dungeon Games) still prevalent



Narasimhan, Karthik, Tejas Kulkarni, and Regina Barzilay. "Language understanding for text-based games using deep reinforcement learning." EMNLP 2015

State Spaces and Action Spaces

$$\langle H, A, T, R, \Psi \rangle.$$

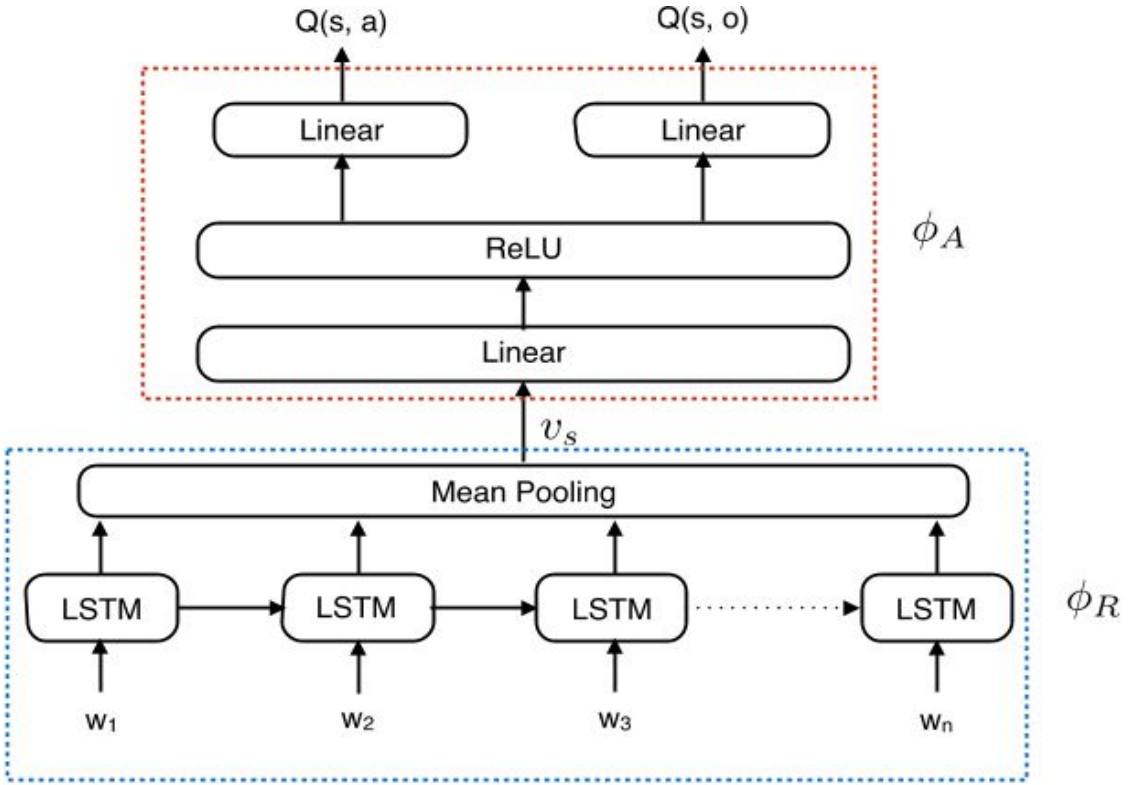
- Hidden state space h ∈ H but given textual description

$$\{ \psi : H \Rightarrow S \}$$

- Actions are commands (action-object pairs) A = {(a,o)}
- $T_{hh'}^{(a,o)}$: Transition Probabilities
- Jointly learn state representations and control policies as learned Strategy/Policy directly builds on the text interpretation

Narasimhan, Karthik, Tejas Kulkarni, and Regina Barzilay. "Language understanding for text-based games using deep reinforcement learning." EMNLP 2015

Learning Representations and Control Policies

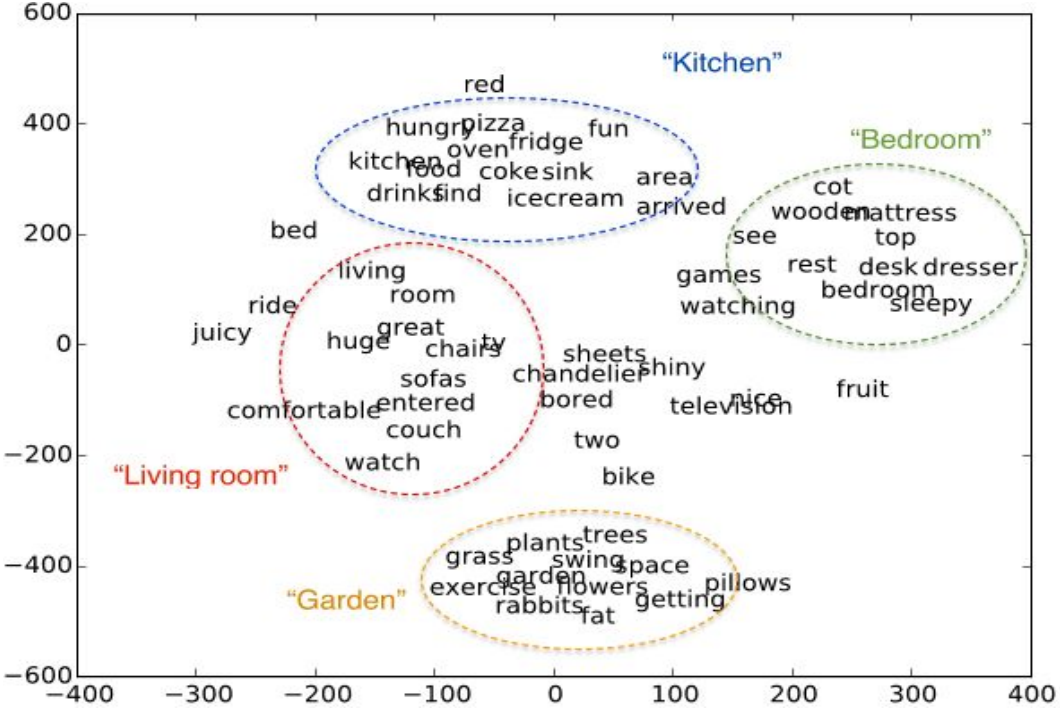


$$v_s = \frac{1}{n}\sum_{k=1}^{n} x_k$$

Narasimhan, Karthik, Tejas Kulkarni, and Regina Barzilay. "Language understanding for text-based games using deep reinforcement learning." EMNLP 2015

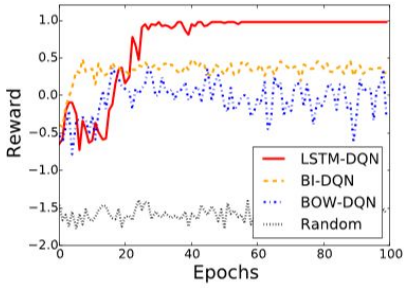
Results (1) :

Learnt Useful Representations for the Game

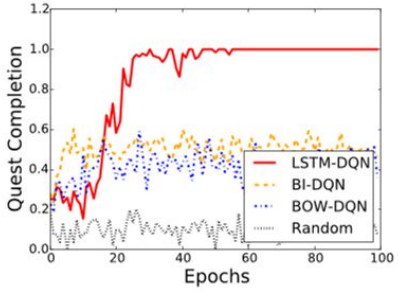


Narasimhan, Karthik, Tejas Kulkarni, and Regina Barzilay. "Language understanding for text-based games using deep reinforcement learning." EMNLP 2015

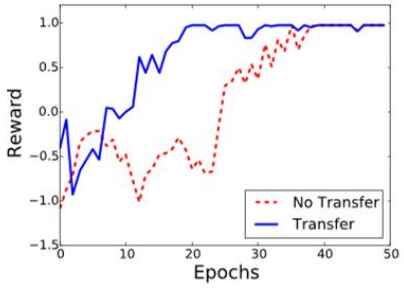
Results (2) :



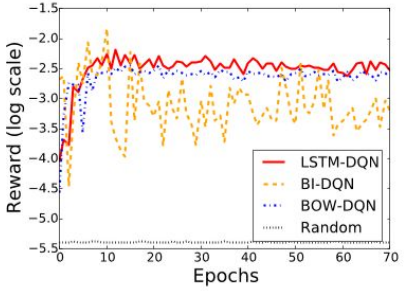
Reward (Home)



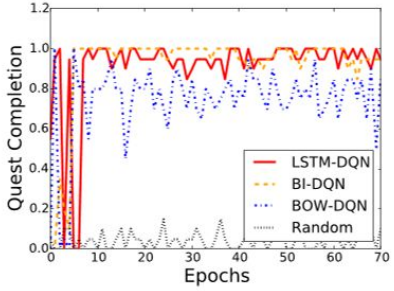
Quest completion (Home)



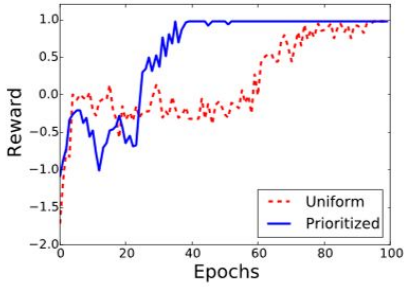
Transfer Learning (Home)



Reward (Fantasy)



Quest completion (Fantasy)



Prioritized Sampling (Home)

Narasimhan, Karthik, Tejas Kulkarni, and Regina Barzilay. "Language understanding for text-based games using deep reinforcement learning." EMNLP 2015

Today’s takeaways

- Bonus RL recap
- Functional Approximation
- Deep Q Network
- Double Deep Q Network
- Dueling Networks
- Recurrent DQN
  - Solving “Doom”
- Hierarchical DQN
- **More applications:**
  - Text based games
  - **Object Detection**
  - Indoor Navigation

Object Detection as a RL problem?

- States:
- Actions:

Object detection as a RL problem?

- States:
- Actions:

[image-link]

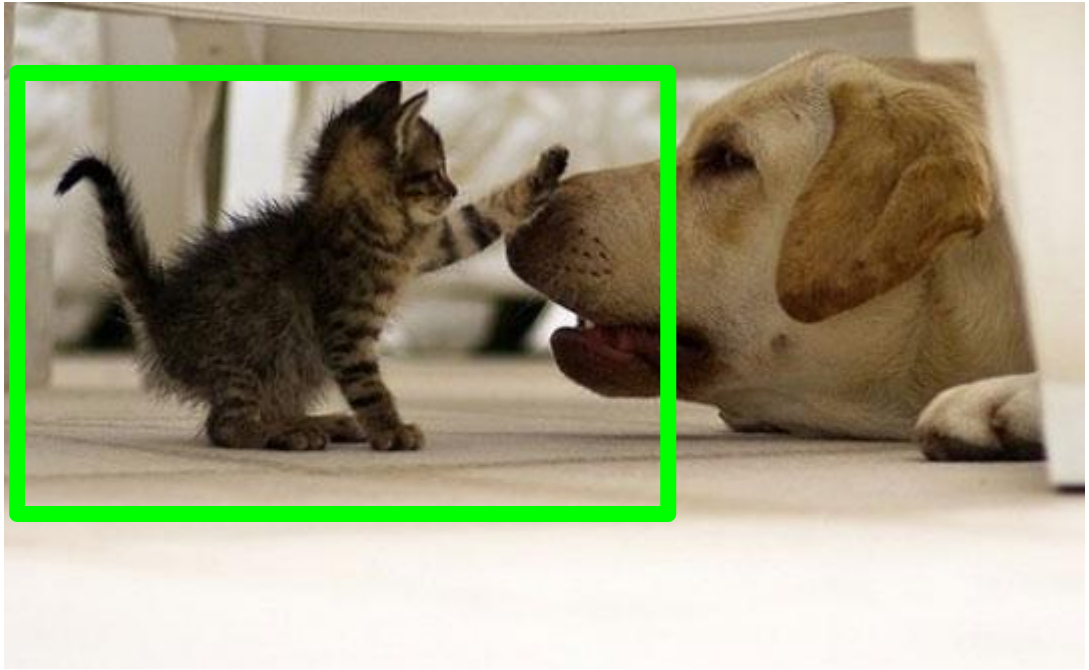
Object detection as a RL problem?

- States:
- Actions:

[image-link]

Object detection as a RL problem?

- States:
- Actions:



[image-link]

Object detection as a RL problem?

- States:
- Actions:

[image-link]

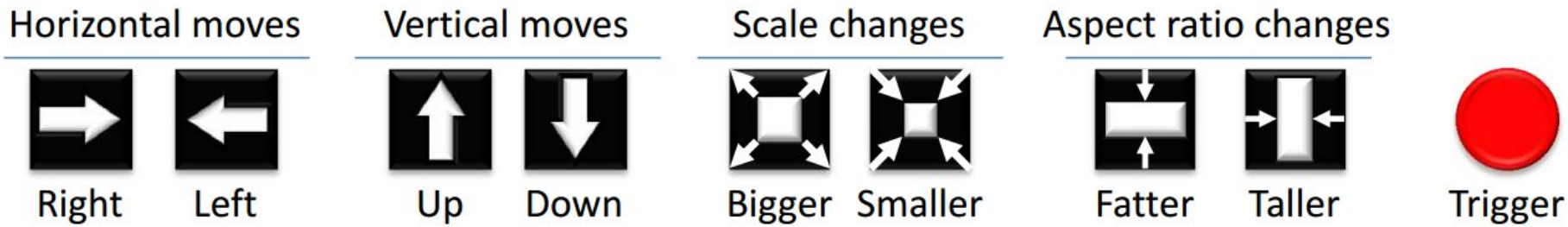
Object detection as a RL problem?

- States:
- **Actions:**

[image-link]

Object detection as a RL problem?

- States:
- **Actions:**
  - **c*(x2-x1) , c*(y2-y1) relative translation**
  - **scale**
  - **aspect ratio**
  - **trigger when IoU is high**

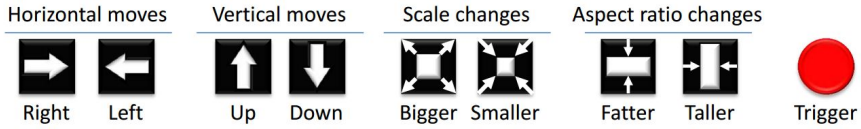


[image-link]

J. Caicedo and S. Lazebnik, *ICCV 2015*

Object detection as a RL problem?

- **States: fc6 feature of pretrained VGG19**
- Actions:
  - c*(x2-x1) , c*(y2-y1) relative translation
  - scale
  - aspect ratio
  - trigger when IoU is high

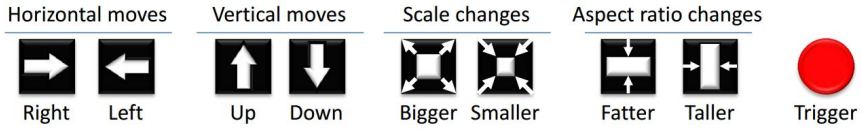


[image-link]

J. Caicedo and S. Lazebnik, *ICCV 2015*

Object detection as a RL problem?

- States: fc6 feature of pretrained VGG19
- Actions:
  - c*(x2-x1) , c*(y2-y1) relative translation
  - scale
  - aspect ratio
  - trigger when IoU is high
- **Reward:**



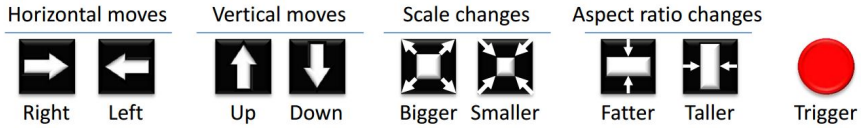
[image-link]

J. Caicedo and S. Lazebnik, *ICCV 2015*

Object detection as a RL problem?

- States: fc6 feature of pretrained VGG19
- Actions:
  - c*(x2-x1) , c*(y2-y1) relative translation
  - scale
  - aspect ratio
  - trigger when IoU is high
- **Reward:**

$$R_a(s, s') = sign\left(IoU(b', g) - IoU(b, g)\right)$$



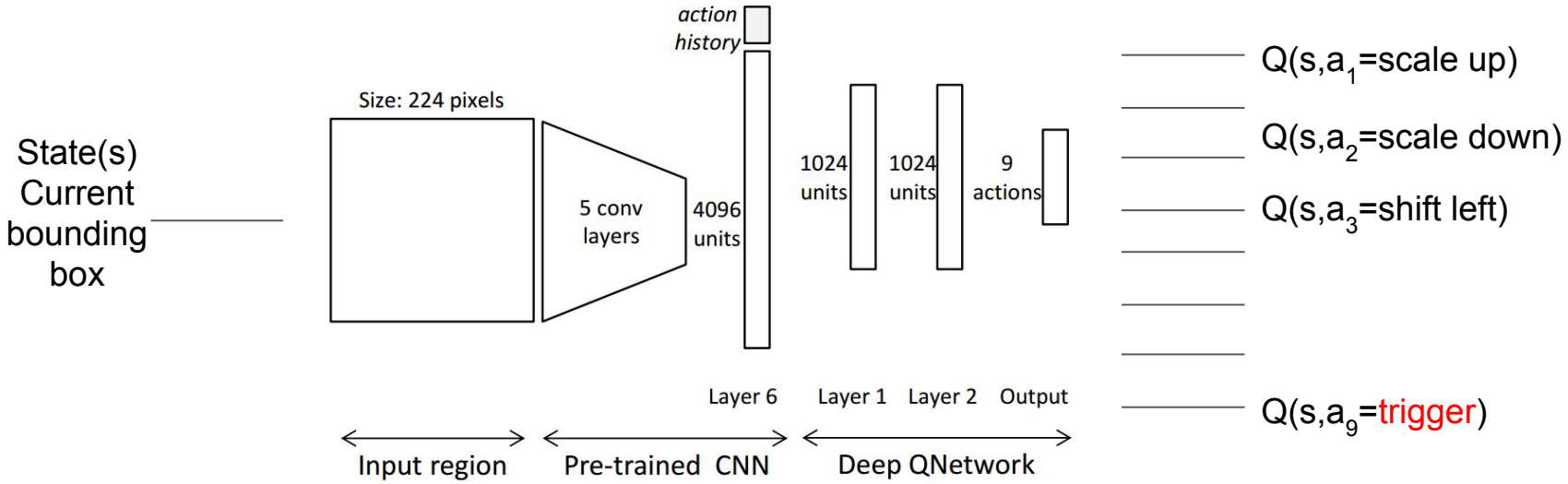
[image-link]

J. Caicedo and S. Lazebnik, *ICCV 2015*

Object detection as a RL problem?

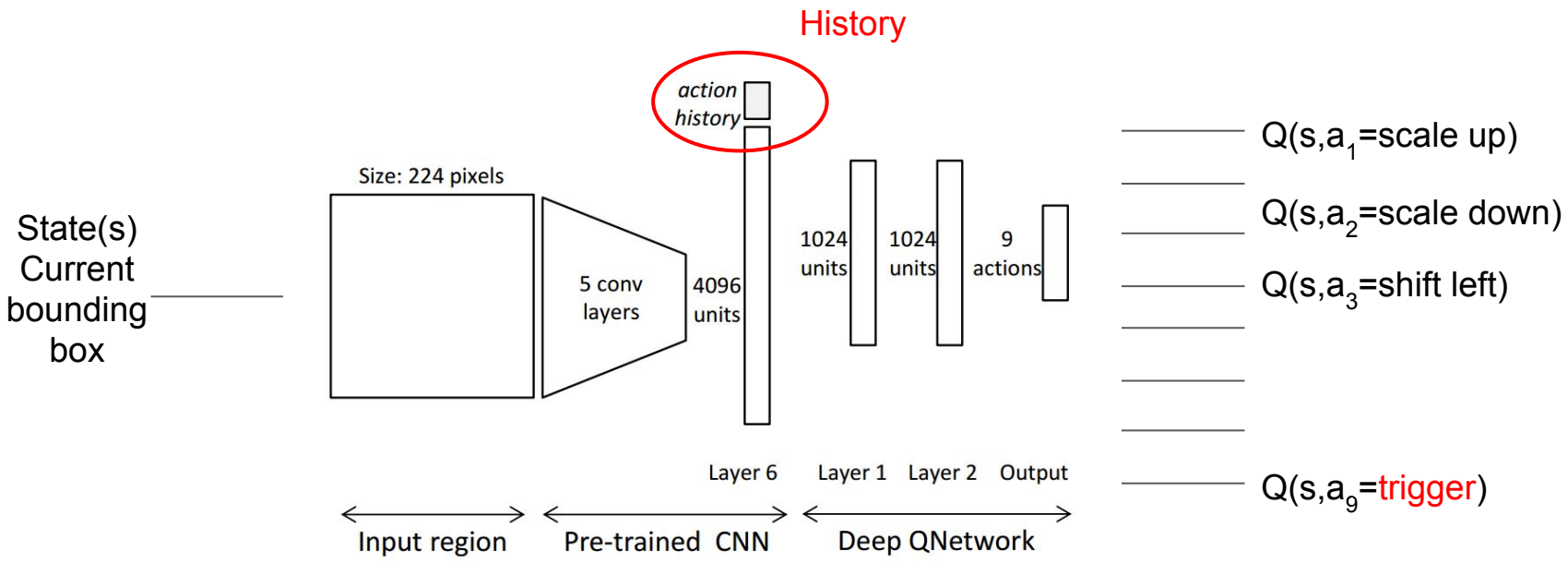
J. Caicedo and S. Lazebnik, *ICCV 2015*

Object detection as a RL problem?



J. Caicedo and S. Lazebnik, *ICCV 2015*

Object detection as a RL problem?

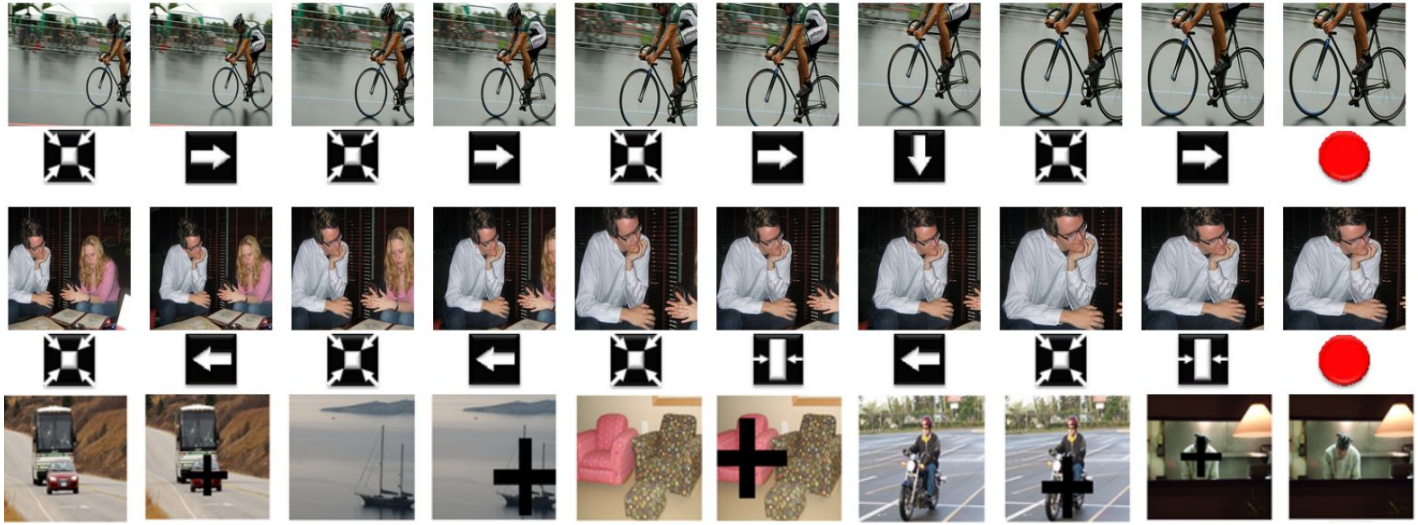


J. Caicedo and S. Lazebnik, *ICCV 2015*

Object detection as a RL problem?

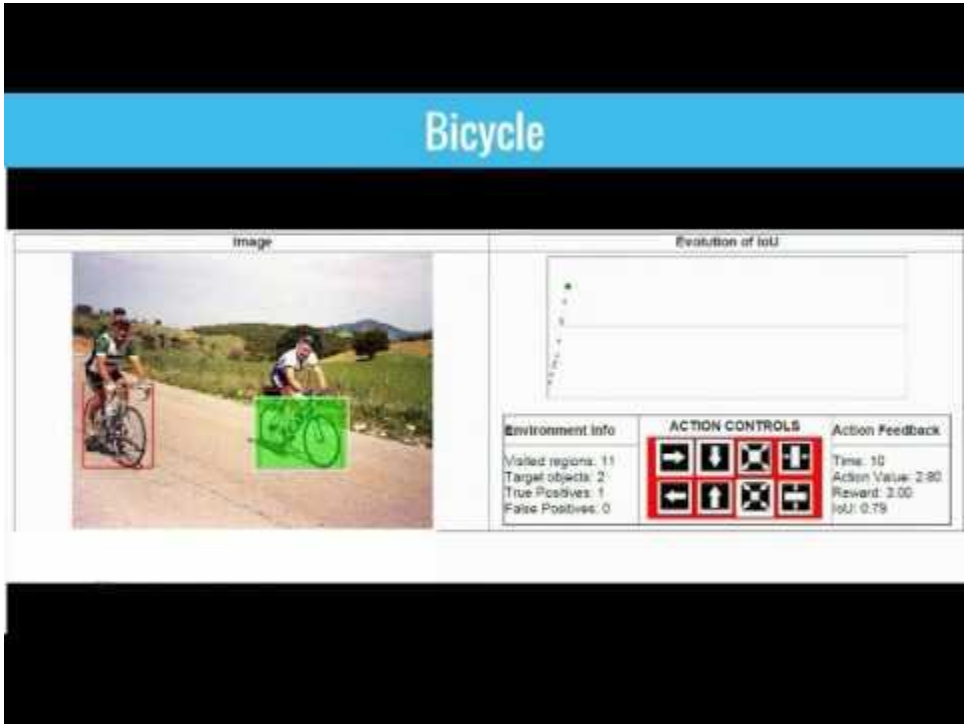
Fine details:

- Class specific, attention-action model
- Does not follow a fixed sliding window trajectory, image dependent trajectory
- Use 16 pixel neighbourhood to incorporate context



J. Caicedo and S. Lazebnik, *ICCV 2015*

Object detection as a RL problem?



J. Caicedo and S. Lazebnik, *ICCV 2015*

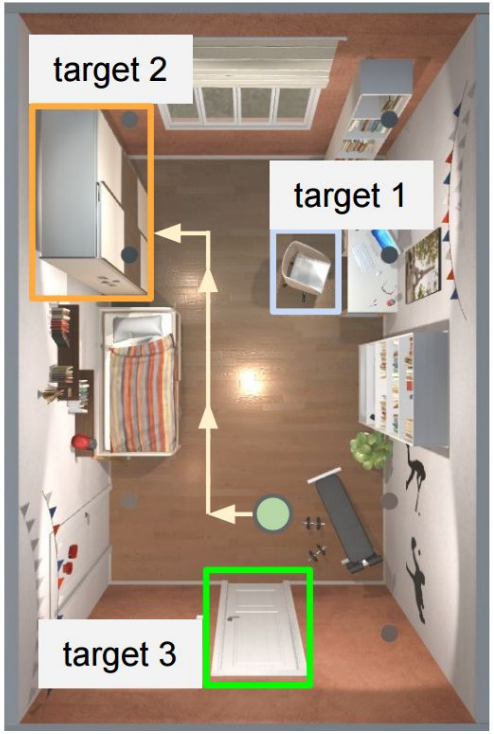
Today’s takeaways

- Bonus RL recap
- Functional Approximation
- Deep Q Network
- Double Deep Q Network
- Dueling Networks
- Recurrent DQN
  - Solving “Doom”
- Hierarchical DQN
- **More applications:**
  - Text based games
  - Object detection
  - **Indoor Navigation**

Navigation as a RL problem?

- **States:**
- Actions:

target-driven visual navigation



observation

target 2

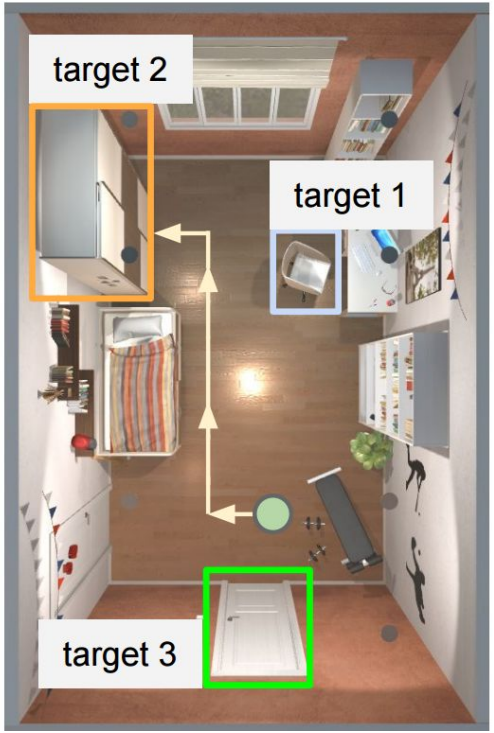
"Target-driven Visual Navigation in Indoor Scenes using Deep Reinforcement Learning"
Yuke Zhu, Roozbeh Mottaghi, Eric Kolve, Joseph J. Lim, Abhinav Gupta, Li Fei-Fei, Ali Farhadi

Navigation as a RL problem?

- States: ResNet-50 features

- Actions:

target-driven visual navigation

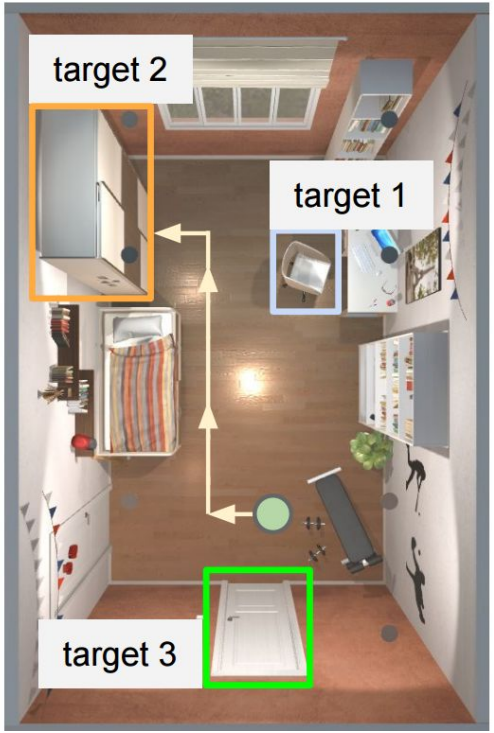


"Target-driven Visual Navigation in Indoor Scenes using Deep Reinforcement Learning"
Yuke Zhu, Roozbeh Mottaghi, Eric Kolve, Joseph J. Lim, Abhinav Gupta, Li Fei-Fei, Ali Farhadi

Navigation as a RL problem?

- States: ResNet-50 feature
- **Actions:**

target-driven visual navigation

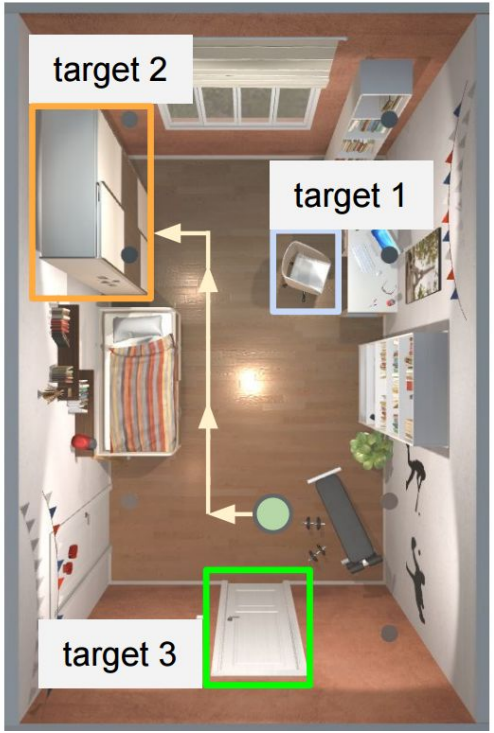


"Target-driven Visual Navigation in Indoor Scenes using Deep Reinforcement Learning"
Yuke Zhu, Roozbeh Mottaghi, Eric Kolve, Joseph J. Lim, Abhinav Gupta, Li Fei-Fei, Ali Farhadi

Navigation as a RL problem?

- States: ResNet-50 feature
- **Actions:**
    - **Forward/backward 0.5 m**
    - **Turn left/right 90 deg**
    - **Trigger**

target-driven visual navigation



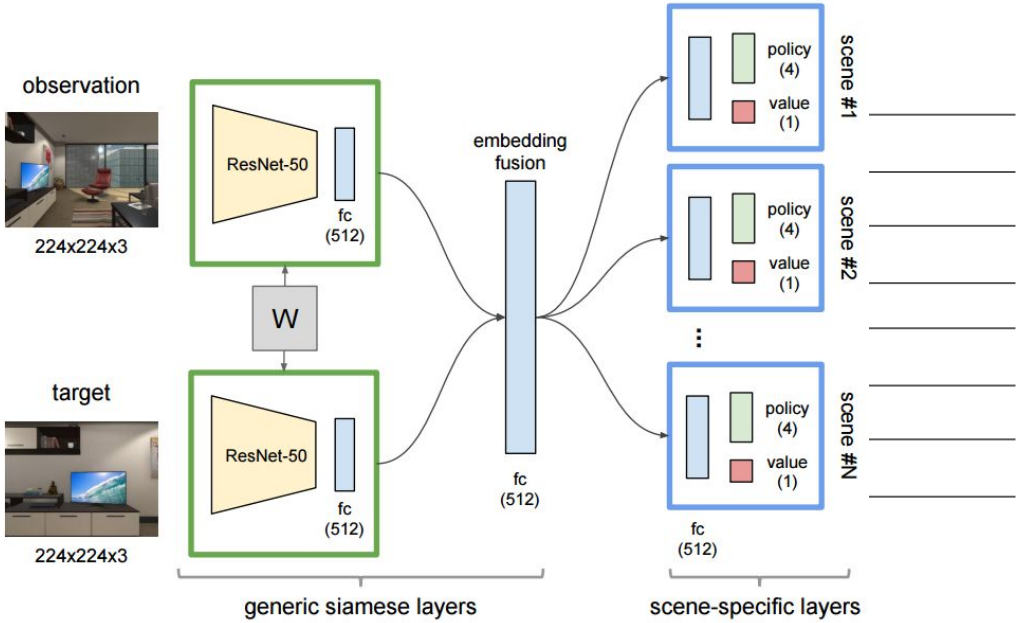
"Target-driven Visual Navigation in Indoor Scenes using Deep Reinforcement Learning"
Yuke Zhu, Roozbeh Mottaghi, Eric Kolve, Joseph J. Lim, Abhinav Gupta, Li Fei-Fei, Ali Farhadi

Navigation as a RL problem?

"Target-driven Visual Navigation in Indoor Scenes using Deep Reinforcement Learning"
Yuke Zhu, Roozbeh Mottaghi, Eric Kolve, Joseph J. Lim, Abhinav Gupta, Li Fei-Fei, Ali Farhadi

Navigation as a RL problem?

State (s)
Current frame and
the target frame



$Q(s,a_1=\text{forward})$

$Q(s,a_2=\text{backward})$

$Q(s,a_3=\text{turn left})$

$Q(s,a_2=\text{turn right})$

$Q(s,a_6=\text{trigger})$

"Target-driven Visual Navigation in Indoor Scenes using Deep Reinforcement Learning"
Yuke Zhu, Roozbeh Mottaghi, Eric Kolve, Joseph J. Lim, Abhinav Gupta, Li Fei-Fei, Ali Farhadi

Navigation as a RL problem?

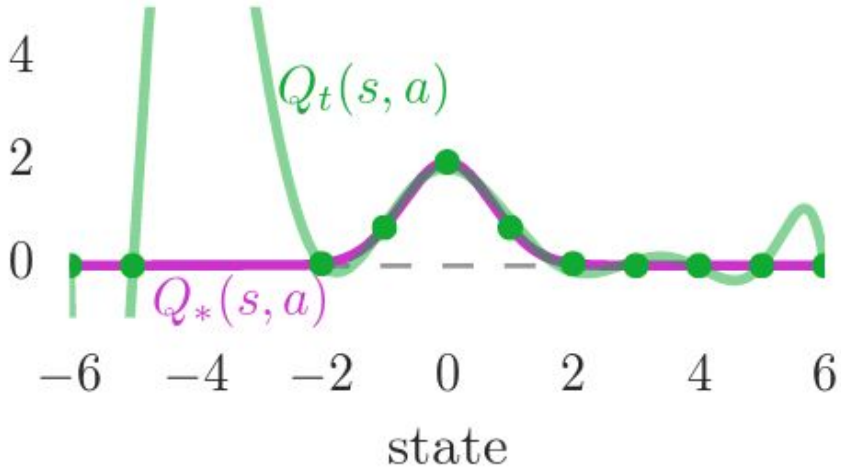
Simulated environment

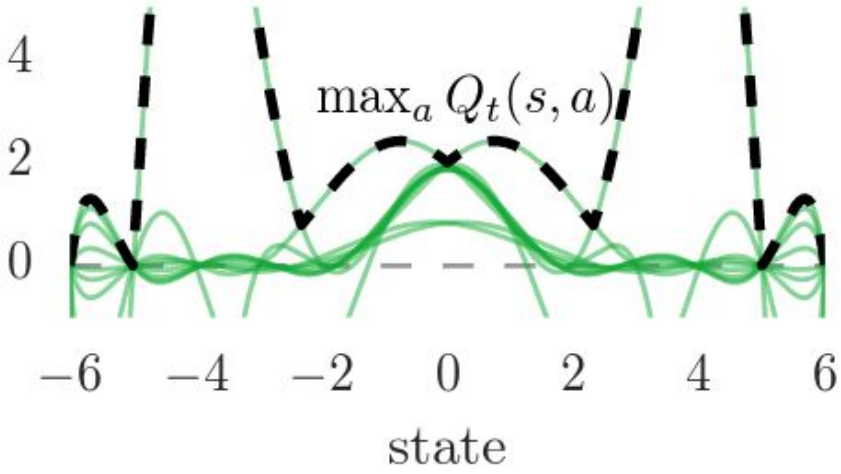
Real environment

Today’s takeaways

- Bonus RL recap
- Functional Approximation
- Deep Q Network
- Double Deep Q Network
- Dueling Networks
- Recurrent DQN
  - Solving “Doom”
- Hierarchical DQN
- More applications:
  - Text based games
  - Object detection
  - Indoor Navigation

Q-Learning Overestimation : Function Approximation





Van Hasselt, Hado, Arthur Guez, and David Silver. "Deep Reinforcement Learning with Double Q-Learning." In *AAAI*, pp. 2094-2100. 2016.

Q-Learning Overestimation : Intuition

$$E\left\{\max_b Q_t(s', b)\right\} > \max_b E\left\{Q_t(s', b)\right\}$$

[Jensen's Inequality]

https://hadovanhasselt.files.wordpress.com/2015/12/doubleqposter.pdf

Q-Learning Overestimation : Intuition

$$E\left\{\max_b Q_t(s', b)\right\} > \underline{\max_b E\left\{Q_t(s', b)\right\}}$$

[Jensen's Inequality]

What we want

https://hadovanhasselt.files.wordpress.com/2015/12/doubleqposter.pdf

Q-Learning Overestimation : Intuition

$$E\left\{\max_b Q_t(s', b)\right\} > \max_b E\left\{Q_t(s', b)\right\}$$

[Jensen's Inequality]

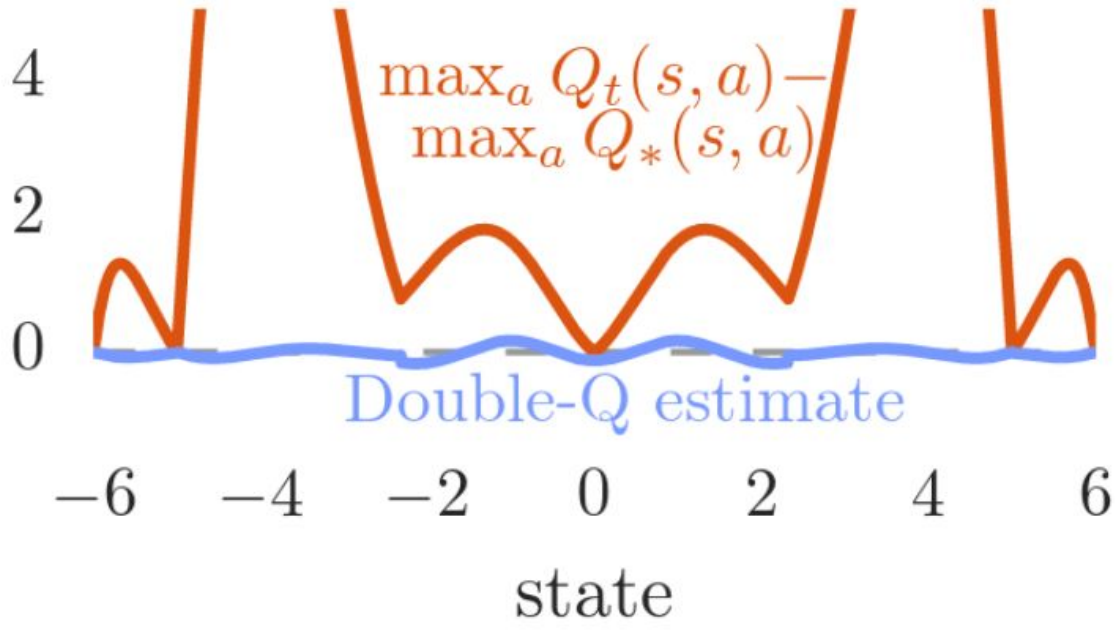
What we estimate in Q-Learning

What we want

$$\boxed{Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha\left[r_{t+1} + \gamma \max_a Q(s_{t+1}, a) - Q(s_t, a_t)\right].}$$

https://hadovanhasselt.files.wordpress.com/2015/12/doubleqposter.pdf

Double Q-Learning : Function Approximation



Average error

+3.35

−0.02

Van Hasselt, Hado, Arthur Guez, and David Silver. "Deep Reinforcement Learning with Double Q-Learning." In *AAAI*, pp. 2094-2100. 2016.

Results

	30 no-ops		Human Starts	
	Mean	**Median**	**Mean**	**Median**
Prior. Duel Clip	**591.9%**	**172.1%**	**567.0%**	**115.3%**
Prior. Single	434.6%	123.7%	386.7%	112.9%
Duel Clip	**373.1%**	**151.5%**	**343.8%**	**117.1%**
Single Clip	341.2%	132.6%	302.8%	114.1%
Single	307.3%	117.8%	332.9%	110.9%
Nature DQN	227.9%	79.1%	219.6%	68.5%

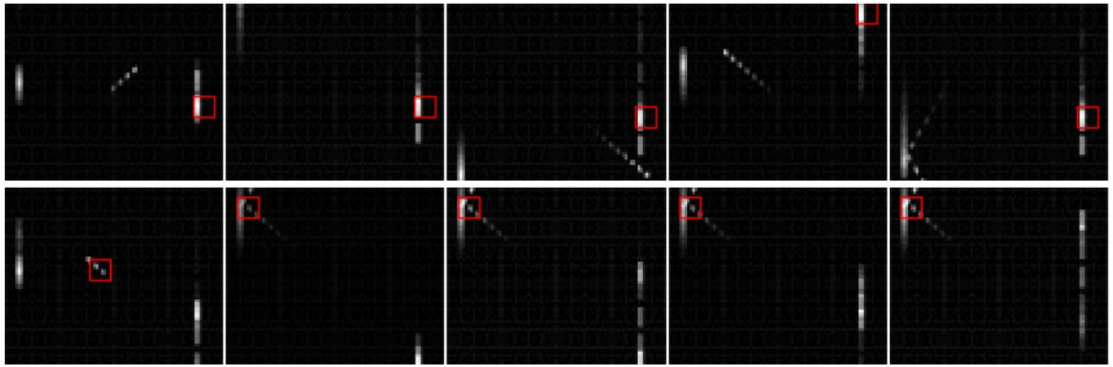
Mean and median scores across all 57 Atari games, measured in percentages of human performance

Wang, Ziyu, Tom Schaul, Matteo Hessel, Hado van Hasselt, Marc Lanctot, and Nando de Freitas. "Dueling network architectures for deep reinforcement learning." *arXiv preprint arXiv:1511.06581* (2015).

Results : Comparison to 10 frame DQN

- Captures in one frame (and history state) what DQN captures in a stack of 10 for Flickering Pong

- 10 frame DQN conv-1 captures paddle information



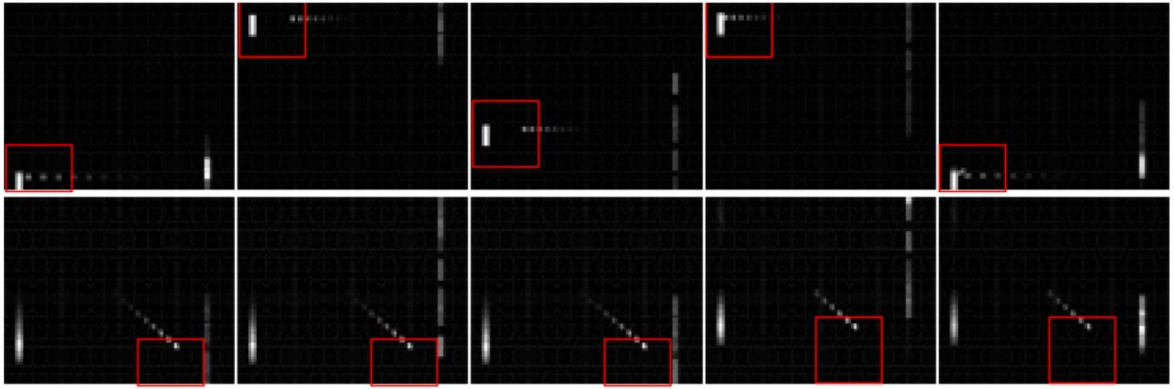
(a) Conv1 Filters

Hausknecht, Matthew, and Peter Stone. "Deep recurrent q-learning for partially observable mdps." *arXiv preprint arXiv:1507.06527* (2015).

Results : Comparison to 10 frame DQN

- Captures in one frame (and history state) what DQN captures in a stack of 10 for Flickering Pong

- 10 frame DQN conv-2 captures paddle and ball direction information



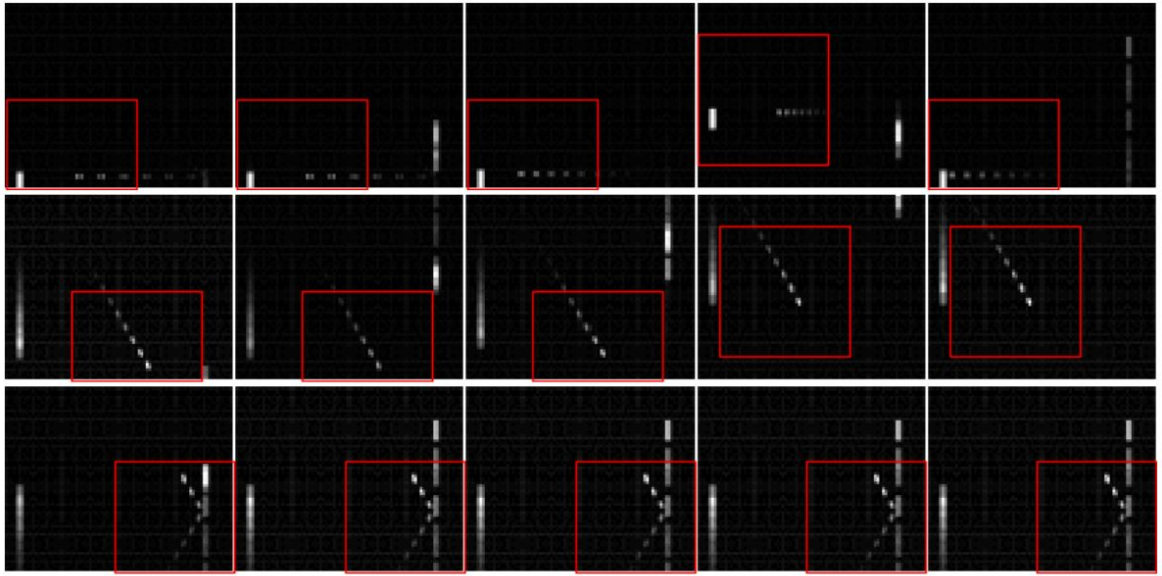
(b) Conv2 Filters

Hausknecht, Matthew, and Peter Stone. "Deep recurrent q-learning for partially observable mdps." *arXiv preprint arXiv:1507.06527* (2015).

Results : Comparison to 10 frame DQN

- Captures in one frame (and history state) what DQN captures in a stack of 10 for Flickering Pong

- 10 frame DQN conv-3 captures paddle, ball direction, velocity and deflection information



(c) Conv3 Filters

Hausknecht, Matthew, and Peter Stone. "Deep recurrent q-learning for partially observable mdps." *arXiv preprint arXiv:1507.06527* (2015).

Results : Comparison to 10 frame DQN

Scores are comparable to 10-frame DQN - outperforming in some and losing in some

	DRQN $\pm std$	DQN $\pm std$	
Game		Ours	Mnih et al.
Asteroids	1020 ($\pm$312)	1070 ($\pm$345)	1629 ($\pm$542)
Beam Rider	3269 ($\pm$1167)	**6923** ($\pm$1027)	6846 ($\pm$1619)
Bowling	62 ($\pm$5.9)	72 ($\pm$11)	42 ($\pm$88)
Centipede	3534 ($\pm$1601)	3653 ($\pm$1903)	8309 ($\pm$5237)
Chopper Cmd	2070 ($\pm$875)	1460 ($\pm$976)	6687 ($\pm$2916)
Double Dunk	**-2** ($\pm$7.8)	-10 ($\pm$3.5)	-18.1 ($\pm$2.6)
Frostbite	**2875** ($\pm$535)	519 ($\pm$363)	328.3 ($\pm$250.5)
Ice Hockey	-4.4 ($\pm$1.6)	-3.5 ($\pm$3.5)	-1.6 ($\pm$2.5)
Ms. Pacman	2048 ($\pm$653)	2363 ($\pm$735)	2311 ($\pm$525)

Table 1: On standard Atari games, DRQN performance parallels DQN, excelling in the games of Frostbite and Double Dunk, but struggling on Beam Rider. Bolded font indicates statistical significance between DRQN and our DQN.[5]

Hausknecht, Matthew, and Peter Stone. "Deep recurrent q-learning for partially observable mdps." *arXiv preprint arXiv:1507.06527* (2015).