


Deep Reinforcement Learning

Kerawit Somchaipeng, Ph.D.

Peng Brothers Co., Ltd.

1997
New York, USA

3.5

MACHINE

2.5

MAN

“computers can easily perform calculation tasks that people consider incredibly difficult, but totally fail at commonsense tasks we find intuitive”

Moravec’s paradox

gen·er·al·i·za·tion

/ˌjen(ə)rələˈzāSH(ə)n/

noun

noun: **generalization**; plural noun: **generalizations**; noun: **generalisation**; plural noun: **generalisations**

a general statement or concept obtained by inference from specific cases.
"he was making sweeping generalizations"

- the action of generalizing.
  "such anecdotes cannot be a basis for generalization"

a·dapt·a·bil·i·ty

/əˌdaptəˈbilədē/

noun

noun: **adaptability**

the quality of being able to adjust to new conditions.
"adaptability is an advantage in the harshly competitive global economy"

- the capacity to be modified for a new use or purpose.
  "the formal beauty and adaptability of plastic"

learn·ing

/ˈlərniNG/

noun

noun: **learning**

the acquisition of knowledge or skills through experience, study, or by being taught.
"these children experienced difficulties in learning"
synonyms: study, studying, education, schooling, tuition, teaching, academic work; research
"a center of learning"

- knowledge acquired through experience, study, or being taught.
  "I liked to parade my learning in front of my sisters"
  synonyms: scholarship, knowledge, education, erudition, intellect, enlightenment, illumination, edification, book learning, information, understanding, wisdom
  "the astonishing range of his learning"
  antonyms: ignorance

March, 2016
Seoul, Korea

4 MACHINE

1 MAN

Deep Blue

1997

IBM

a Super Computer

Chess

10^{40}

Brute-force Search Algorithm

Hard-coded with a set of specialized rules

Can do only one thing

AlphaGo

2016

DeepMind

a group of machines

Go

10^{170}

Deep Learning

Supervised Learning
Reinforcement Learning

Can be generalized

Deep Learning

aka. (Deep) Neural Networks

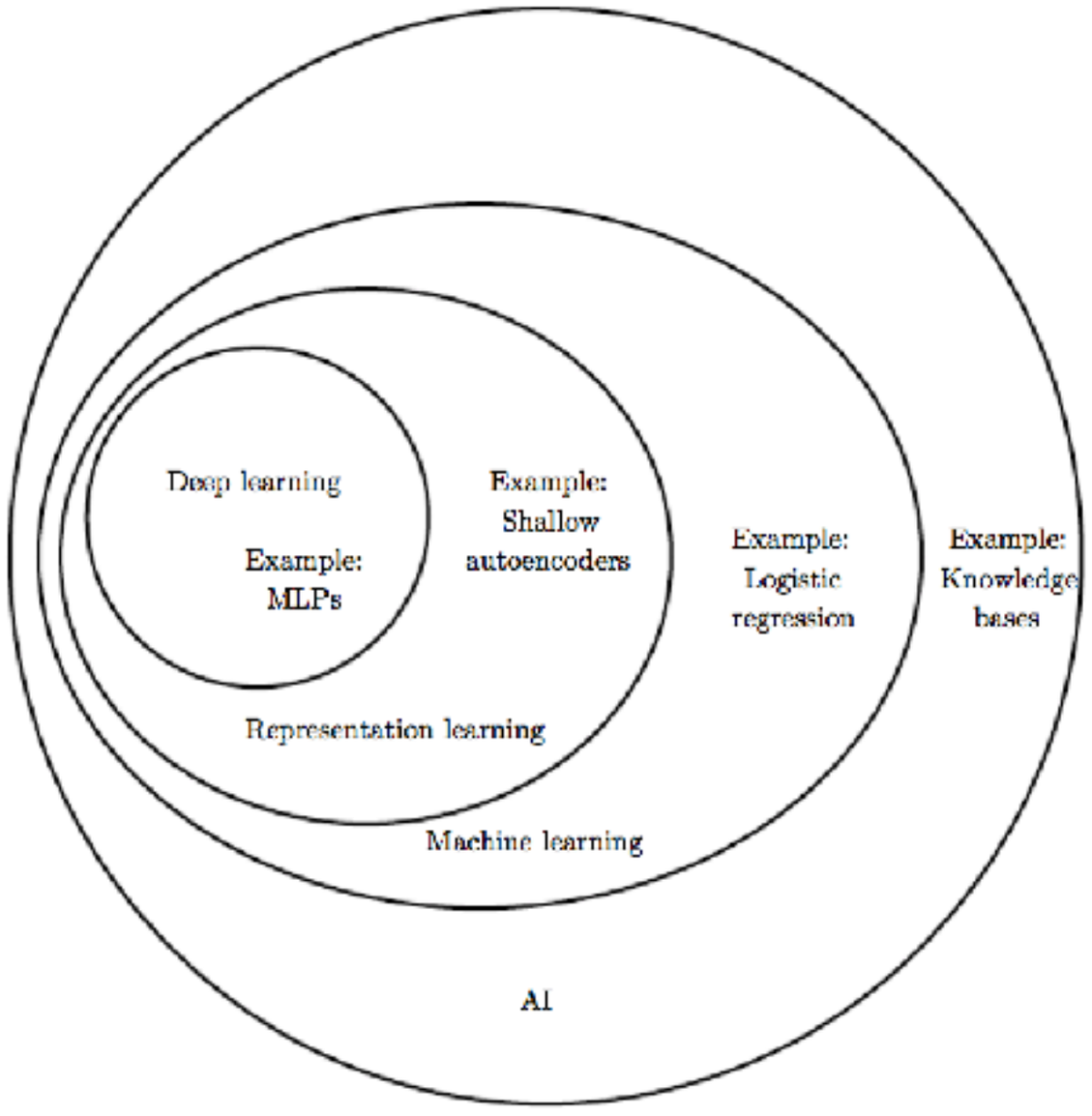


Figure 1.4: A Venn diagram showing how deep learning is a kind of representation learning, which is in turn a kind of machine learning, which is used for many but not all approaches to AI. Each section of the Venn diagram includes an example of an AI technology.

http://www.deeplearningbook.org/contents/intro.html

LeNet-1

1993

Yann LeCun

https://www.youtube.com/watch?v=FwFduRA_L6Q

LeNet-1

(1993)

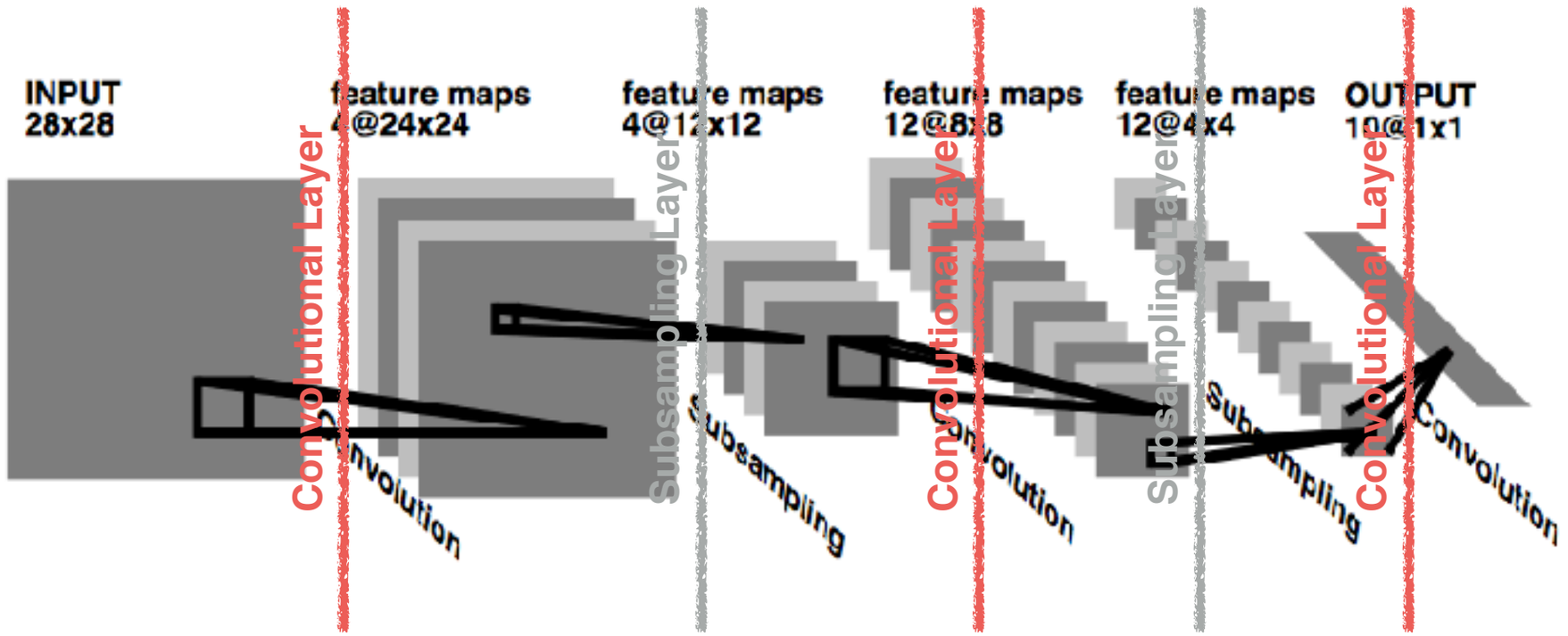


Figure 1: Architecture of LeNet 1. Each plane represents a feaure map, i.e. a set of units whose weights are constrained to be identical. Input images are sized to fit in a 16 x 16 pixel field, but enough blank pixels are added around the border of this field to avoid edge effects in the convolution calculations.

http://yann.lecun.com/exdb/publis/pdf/lecun-95b.pdf

LeNet-5

(1994)

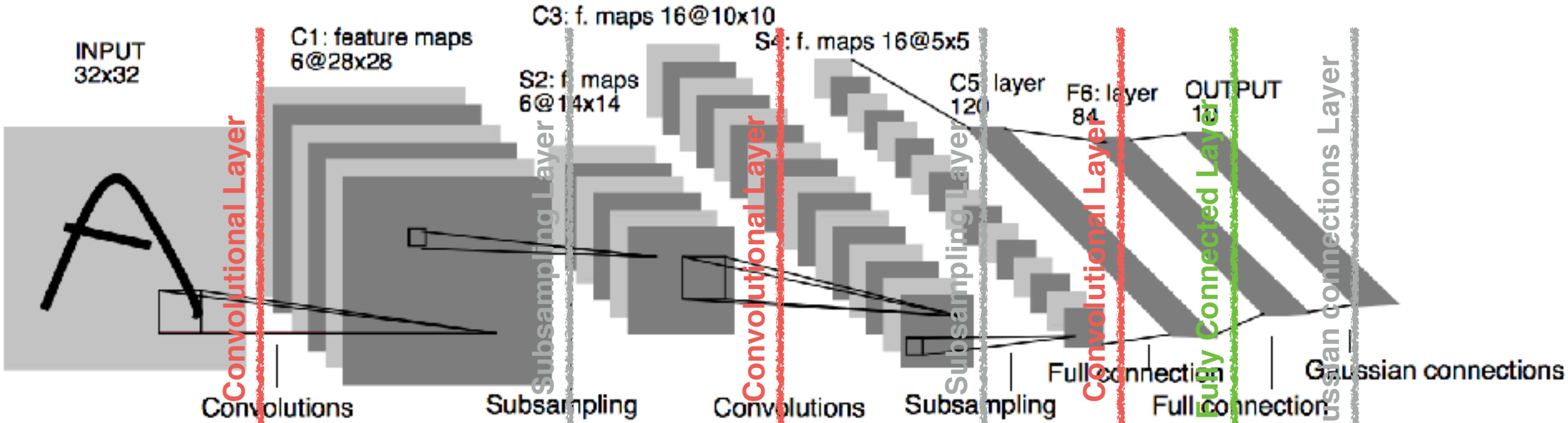


Fig. 2. Architecture of LeNet-5, a Convolutional Neural Network, here for digits recognition. Each plane is a feature map, i.e. a set of units whose weights are constrained to be identical.

http://yann.lecun.com/exdb/publis/pdf/lecun-01a.pdf

1998 - 2010

Abstract

We trained a large, deep convolutional neural network to classify the 1.2 million high-resolution images in the ImageNet LSVRC-2010 contest into the 1000 different classes. On the test data, we achieved top-1 and top-5 error rates of 37.5% and 17.0% which is considerably better than the previous state-of-the-art. The neural network, which has 60 million parameters and 650,000 neurons, consists of five convolutional layers, some of which are followed by max-pooling layers, and three fully-connected layers with a final 1000-way softmax. To make training faster, we used non-saturating neurons and a very efficient GPU implementation of the convolution operation. To reduce overfitting in the fully-connected layers we employed a recently-developed regularization method called "dropout" that proved to be very effective. We also entered a variant of this model in the ILSVRC-2012 competition and achieved a winning top-5 test error rate of 15.3%, compared to 26.2% achieved by the second-best entry.

consists of hundreds of thousands of fully-segmented images, and ImageNet [6], which consists of over 15 million labeled high-resolution images in over 22,000 categories.

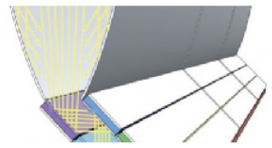
https://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

MIT Technology Review

10 Breakthrough Tech

Smart Watches
The designers of the Pebble watch realized that a mobile phone is more useful if you don't have to take it out of your pocket.

Ultra-Efficient Solar Power
Doubling the efficiency of solar devices would completely change the economics of renewable energy. Here is a design that just might make it possible.

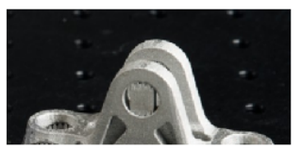


Memory Implants
A maverick neuroscientist believes he has deciphered the code by which the brain forms long-term memories.

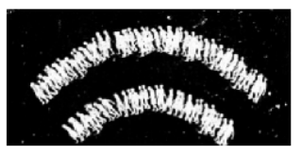
Prenatal DNA Sequencing
Reading the DNA of fetuses is the next frontier of the genome revolution. Do you really want to know the genetic destiny of your unborn child?

Deep Learning
With massive amounts of computational power, machines can now recognize objects and translate speech in real time. Artificial intelligence is finally getting smart.

Additive Manufacturing
GE, the world's largest manufacturer, is on the verge of using 3-D printing to make jet parts.



Big Data from Cheap Phones
Collecting and analyzing information from simple cell phones can provide surprising insights into how people move about and behave—and even help us understand the spread of diseases.



Temporary Social Media
Messages that quickly self-destruct could enhance the privacy of online communication and make people feel freer to be spontaneous.

Supergrids
A high-power circuit breaker could finally make DC power grids practical.

Baxter: The Blue-Collar Robot
Rethink Robotics' new creation is easy to interact with, but the innovations behind the robot show just how hard it is to get along with people.

:h Q

re+

Subscribe

10 Breakthrough Tech

able, or

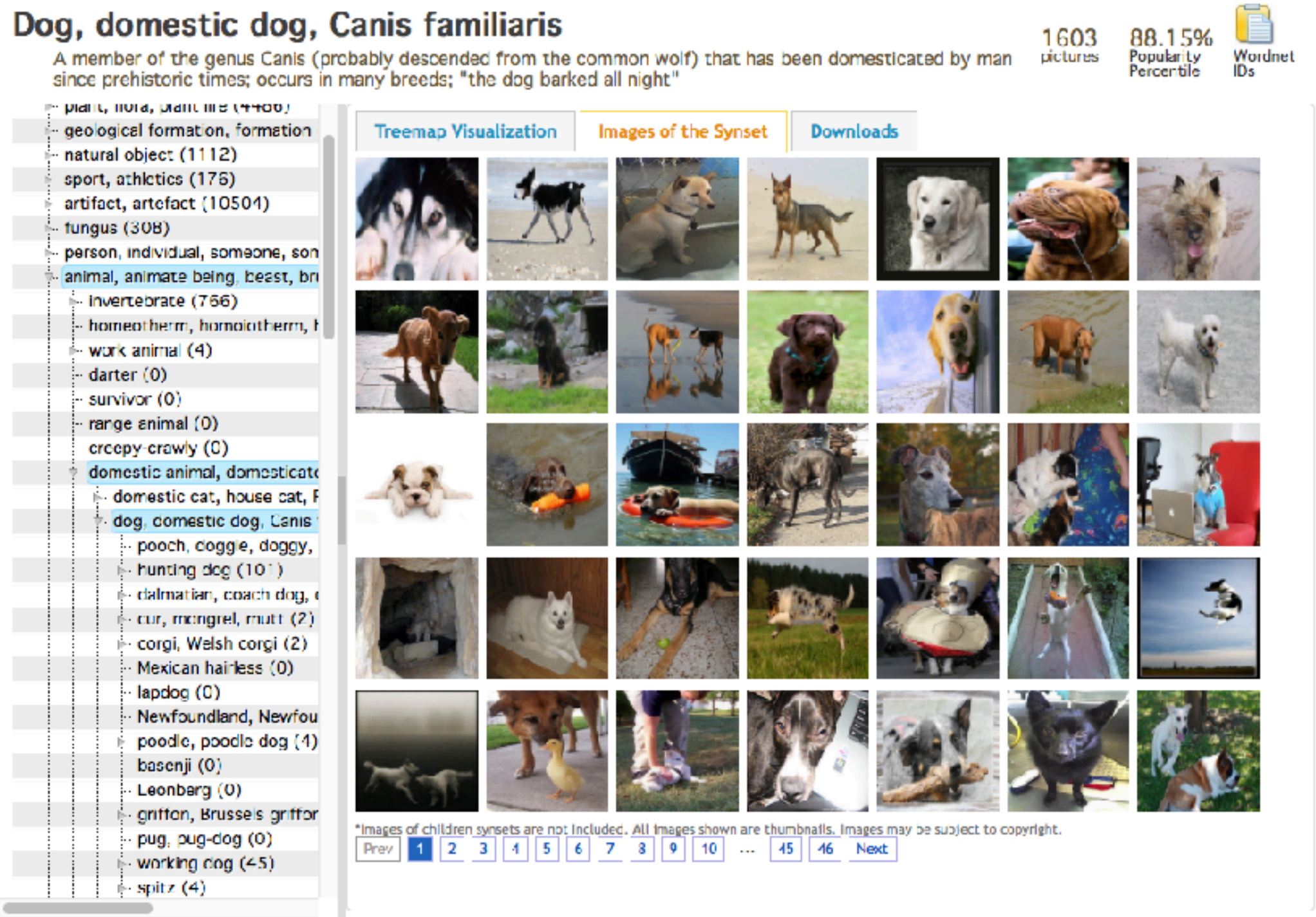
hnology

ng up

ologies.

will

IMAGENET



http://image-net.org/synset?wnid=n02084071

AlexNet

(2012)

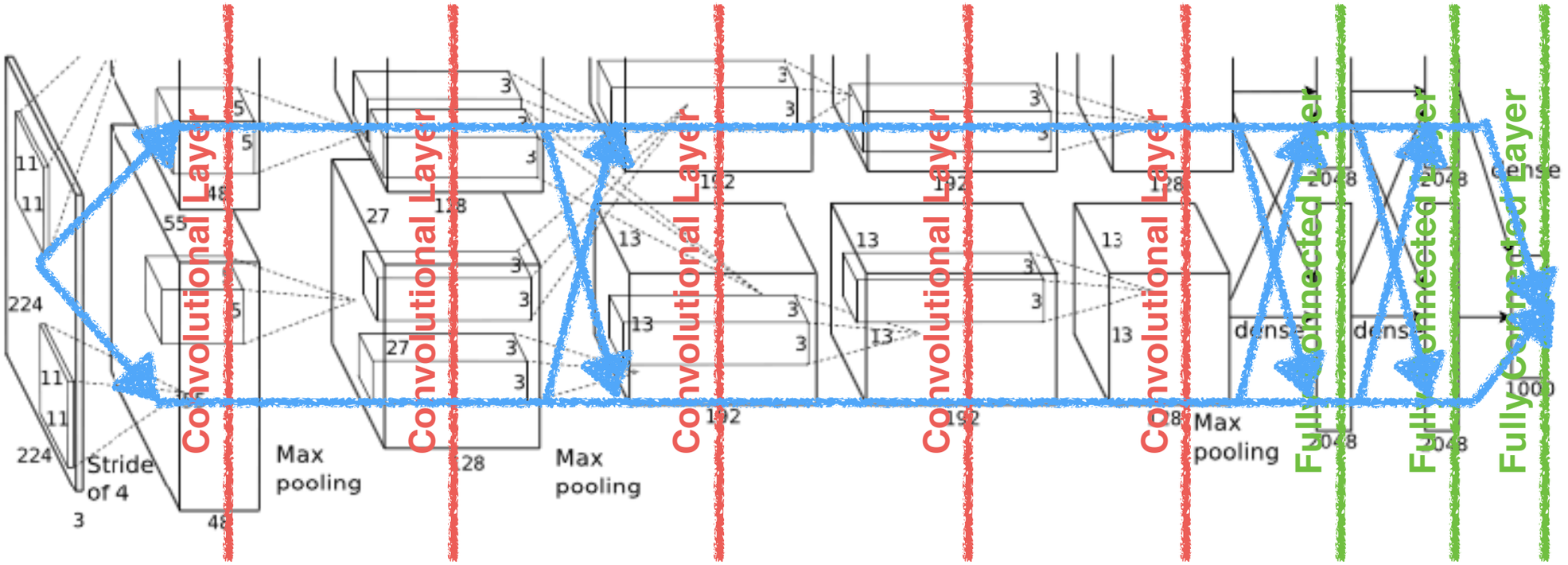


Figure 2: An illustration of the architecture of our CNN, explicitly showing the delineation of responsibilities between the two GPUs. One GPU runs the layer-parts at the top of the figure while the other runs the layer-parts at the bottom. The GPUs communicate only at certain layers. The network's input is 150,528-dimensional, and the number of neurons in the network's remaining layers is given by 253,440–186,624–64,896–64,896–43,264–4096–4096–1000.

https://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

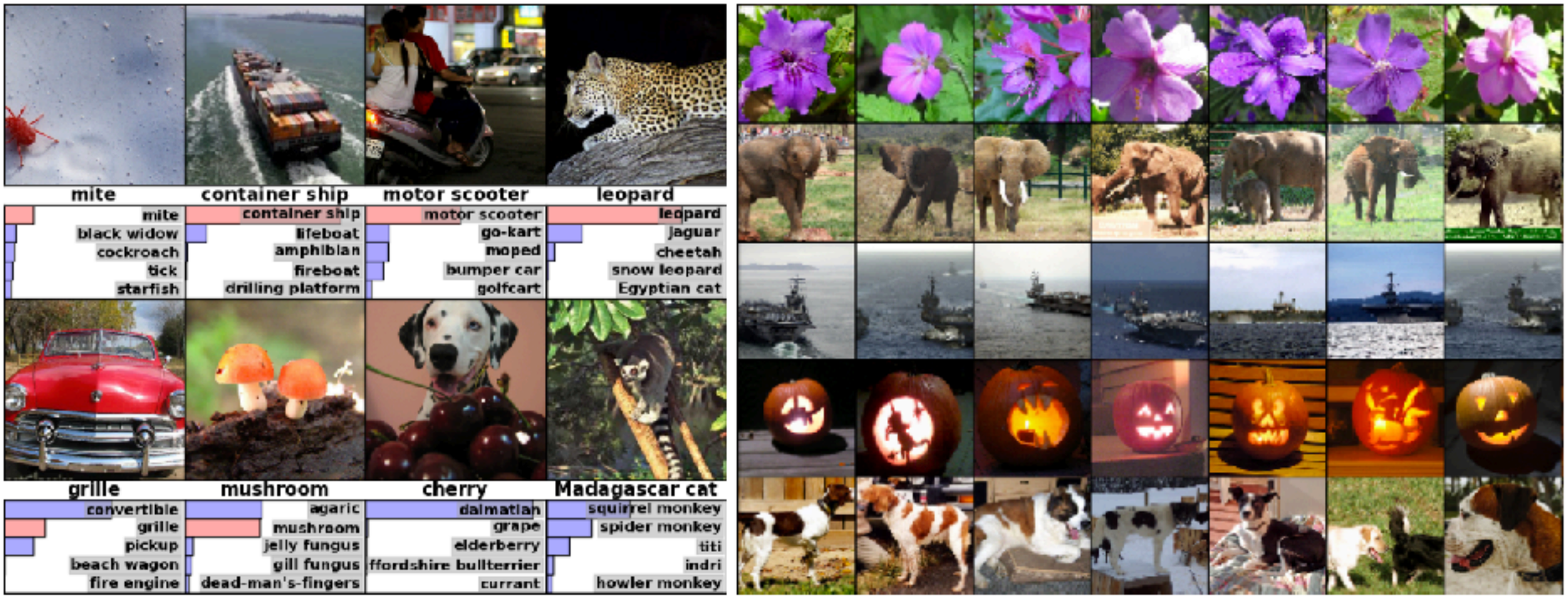
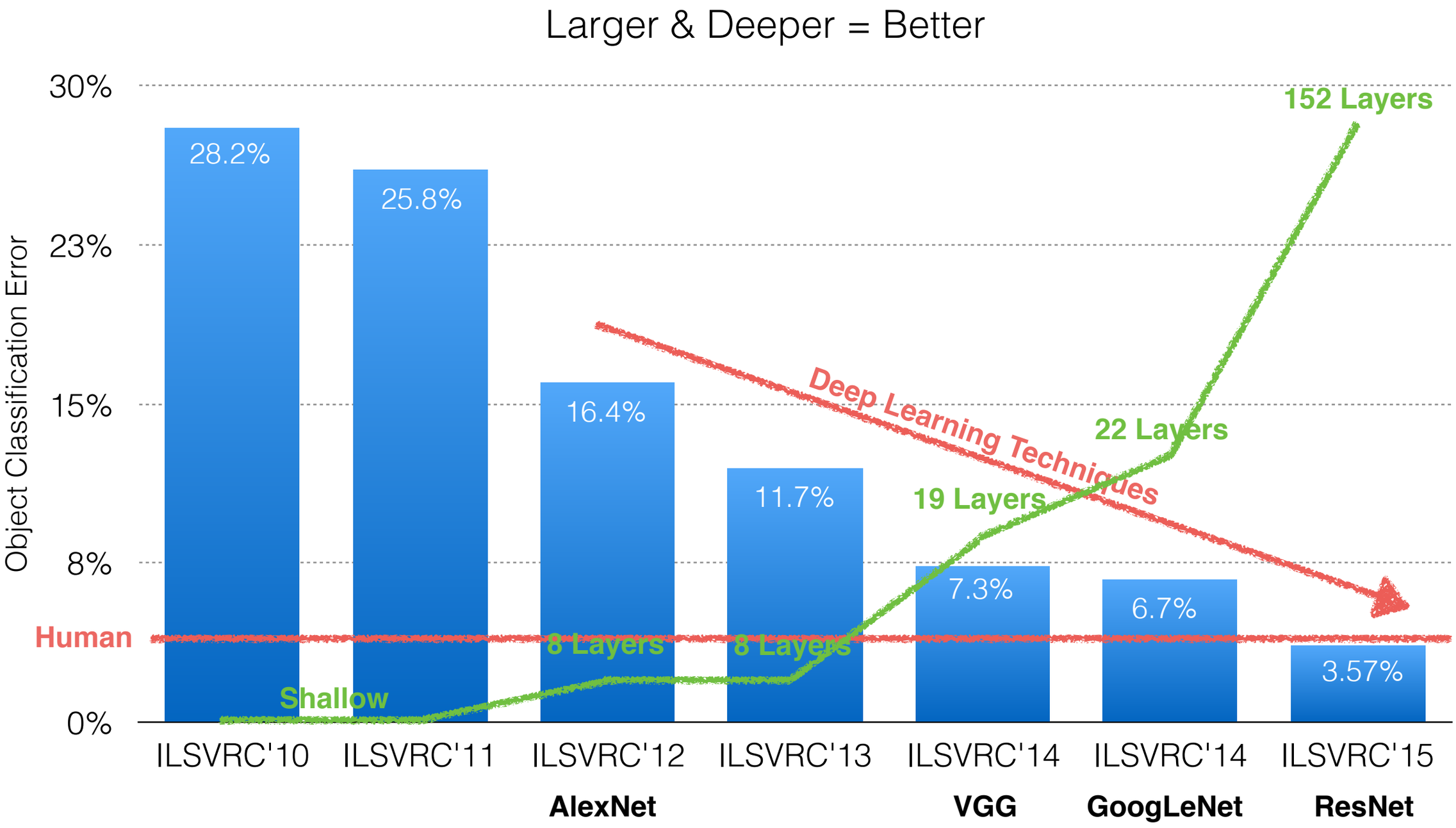


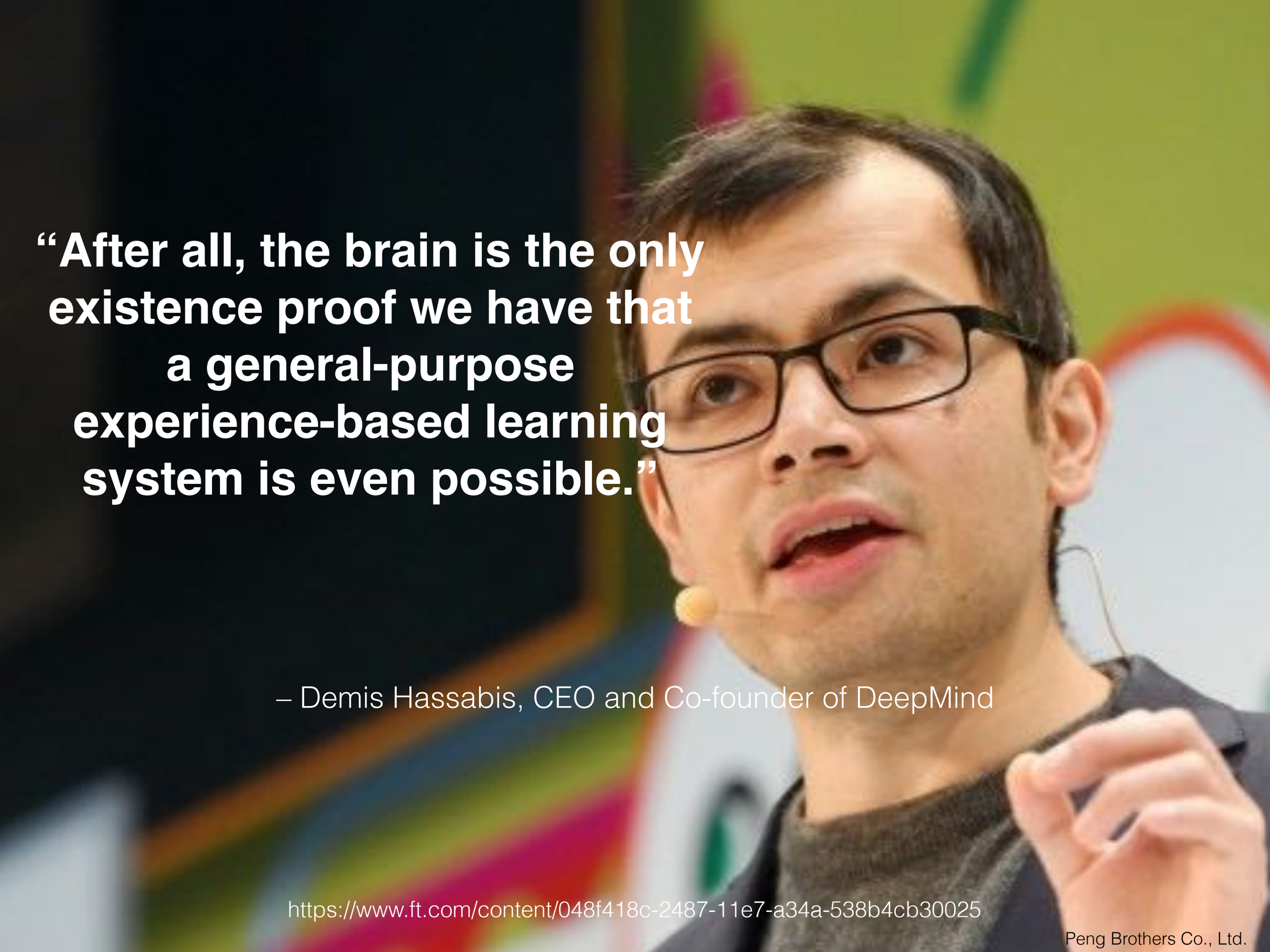
Figure 4: **(Left)** Eight ILSVRC-2010 test images and the five labels considered most probable by our model. The correct label is written under each image, and the probability assigned to the correct label is also shown with a red bar (if it happens to be in the top 5). **(Right)** Five ILSVRC-2010 test images in the first column. The remaining columns show the six training images that produce feature vectors in the last hidden layer with the smallest Euclidean distance from the feature vector for the test image.

https://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

ImageNet ILSVRC



http://image-net.org


“After all, the brain is the only existence proof we have that a general-purpose experience-based learning system is even possible.”
– Demis Hassabis, CEO and Co-founder of DeepMind
https://www.ft.com/content/048f418c-2487-11e7-a34a-538b4cb30025
Peng Brothers Co., Ltd.

MIT Technology Review

10 Breakthrough Tech

Reversing Paralysis
Scientists are making remarkable progress at using brain implants to restore the freedom of movement that spinal cord injuries take away.

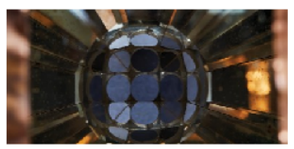
Self-Driving Trucks
Tractor-trailers without a human at the wheel will soon barrel onto highways near you. What will this mean for the nation's 1.7 million truck drivers?

Paying with Your Face
Face-detecting systems in China now authorize payments, provide access to facilities, and track down criminals. Will other countries follow?

Practical Quantum Computers
Advances at Google, Intel, and several research groups indicate that computers with previously unimaginable power are finally within reach.

The 360-Degree Selfie
Inexpensive cameras that make spherical images are opening a new era in photography and changing the way people share stories.

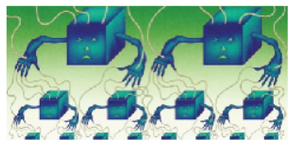
Hot Solar Cells
By converting heat to focused beams of light, a new solar device could create cheap and continuous power.



Gene Therapy 2.0
Scientists have solved fundamental problems that were holding back cures for rare hereditary disorders. Next we'll see if the same approach can take on cancer, heart disease, and other common illnesses.

The Cell Atlas
Biology's next mega-project will find out what we're really made of.

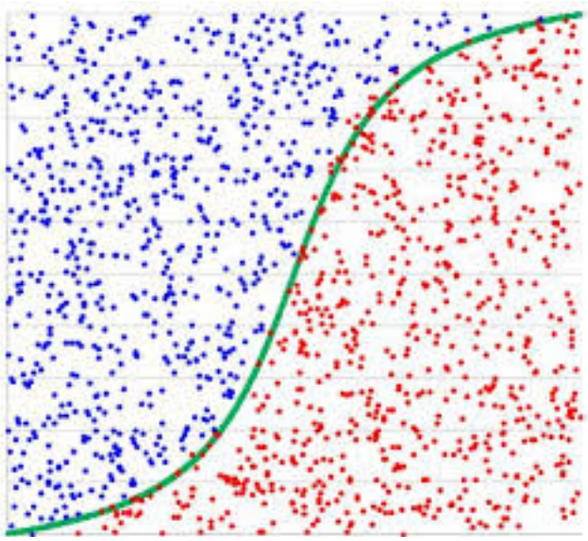
Botnets of Things
The relentless push to add connectivity to home gadgets is creating dangerous side effects that figure to get even worse.



Reinforcement Learning
By experimenting, computers are figuring out how to do things that no programmer could teach them.

10 Breakthrough Tech

Types of Machine Learning

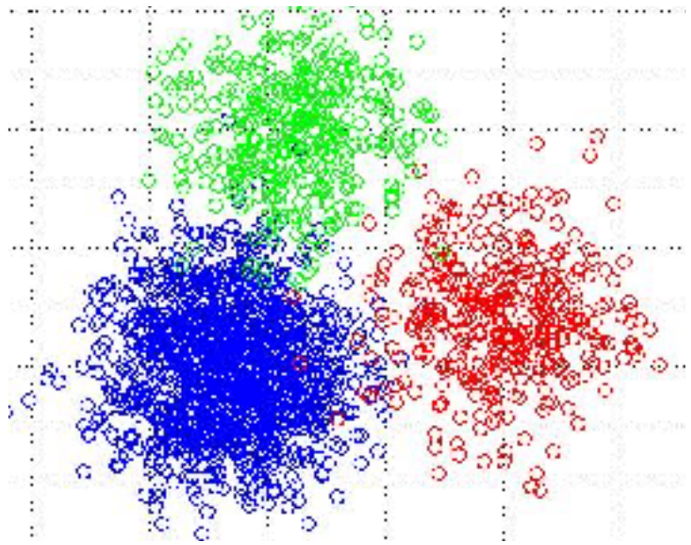


Supervised Learning

- Classification
- Regression
- Segmentation

Reinforcement Learning

- Control
- Long-term Planing
- Decison Making
- Games



Unsupervised Learning

- Clustering
- Anomaly Detection
- Representation Learning
- Dimensionality Reduction

Stanford Autonomous Helicopter

2008

Andrew Ng

https://www.youtube.com/watch?v=0JL04JJjocc

Statemen now now
that we d

That is co

As I've sa rning is
unsuperv arning
would be cake,
and reinf know
how to m hake the
cake.

We need an even
think of g but.
What abo

#deeplea

https://www.facebook.com/yann.lecun/posts/10153426023477143

AlphaGo	AlphaGo Zero
Supervised Learning Reinforcement Learning	Pure Reinforcement Learing
Seperate Policy & Value Network	Combined Policy & Value Network
1[illegible]6 G[illegible]PUs 4[illegible]T[illegible]Us	4 T[illegible]U
Several Months	72 Hrs
0	100

Empirical Evaluation of AlphaGo Zero

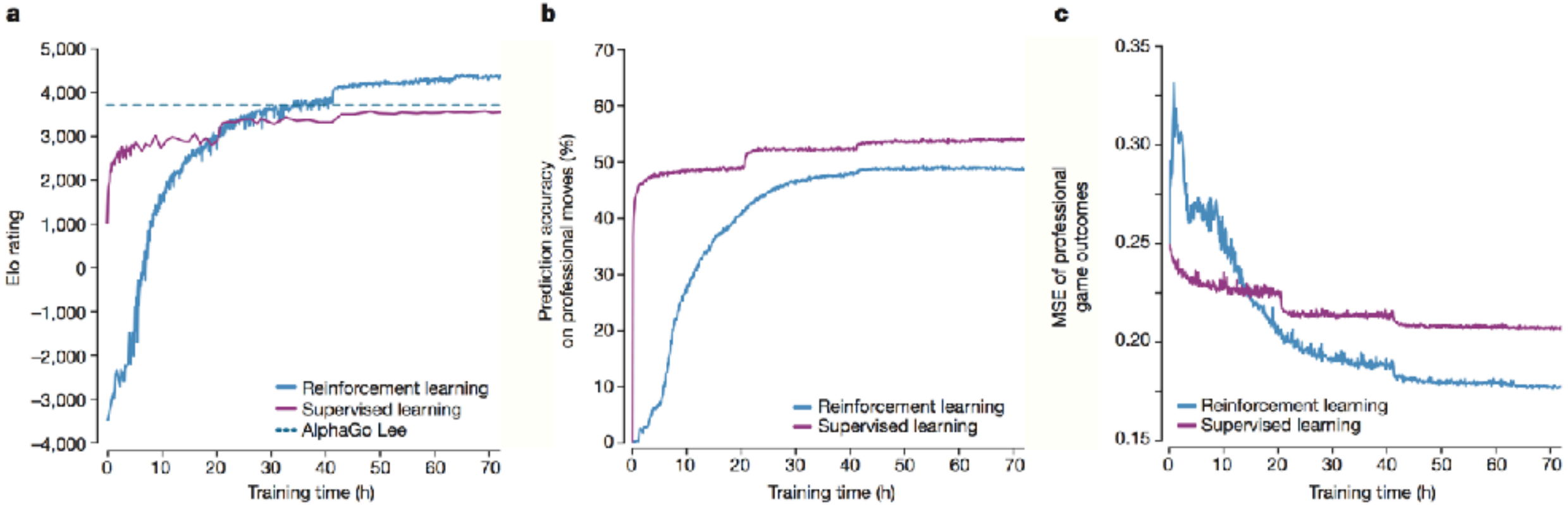


Figure 3 | Empirical evaluation of AlphaGo Zero. a, Performance of self-play reinforcement learning. The plot shows the performance of each MCTS player α_{θ_i} from each iteration i of reinforcement learning in AlphaGo Zero. Elo ratings were computed from evaluation games between different players, using 0.4 s of thinking time per move (see Methods). For comparison, a similar player trained by supervised learning from human data, using the KGS dataset, is also shown. **b**, Prediction accuracy on human professional moves. The plot shows the accuracy of the neural network f_{θ_i} at each iteration of self-play i, in predicting human professional moves from the GoKifu dataset. The accuracy measures the percentage of positions in which the neural network assigns the highest probability to the human move. The accuracy of a neural network trained by supervised learning is also shown. **c**, Mean-squared error (MSE) of human professional game outcomes. The plot shows the MSE of the neural network f_{θ_i}, at each iteration of self-play i, in predicting the outcome of human professional games from the GoKifu dataset. The MSE is between the actual outcome $z \in \{-1, +1\}$ and the neural network value v, scaled by a factor of $\frac{1}{4}$ to the range of 0–1. The MSE of a neural network trained by supervised learning is also shown.

https://www.nature.com/articles/nature24270.epdf?author_access_token=VJXbVjaSHxFoctQQ4p2k4tRgN0jAjWel9jnR3ZoTv0PVW4gB86EEpGqTRDtpIz-2rmo8-KG06gqVobU5NSCFeHILHcVFUeMsbvwS-lxjqQGg98faovwjxeTUgZAUMnRQ

“Human has accumulated Go knowledge from millions of games played over thousands of years, collectively distilled into patterns, proverbs and books. In the space of a few days, starting *tabula rasa*, AlphaGo Zero was able to rediscover much of this Go knowledge, as well as novel strategies that provide new insights into the oldest of games”

https://www.nature.com/articles/nature24270.epdf?author_access_token=VJXbVjaSHxFoctQQ4p2k4tRgN0jAjWel9jnR3ZoTv0PVW4gB86EEpGqTRDtpIz-2rmo8-KG06gqVobU5NSCFeHILHcVFUeMsbvwS-lxjqQGg98faovwjxeTUgZAUMnRQ

LETTER

Here we use recent advances in training deep neural networks[9–11] to develop a novel artificial agent, termed a deep Q-network, that can learn successful policies directly from high-dimensional sensory inputs using end-to-end reinforcement learning. We tested this agent on the challenging domain of classic Atari 2600 games[12]. We demonstrate that the deep Q-network agent, receiving only the pixels and the game score as inputs, was able to surpass the performance of all previous algorithms and achieve a level comparable to that of a professional human games tester across a set of 49 games, using the same algorithm, network architecture and hyperparameters. This work bridges the divide between high-dimensional sensory inputs and actions, resulting in the first artificial agent that is capable of learning to excel at a diverse array of challenging tasks.

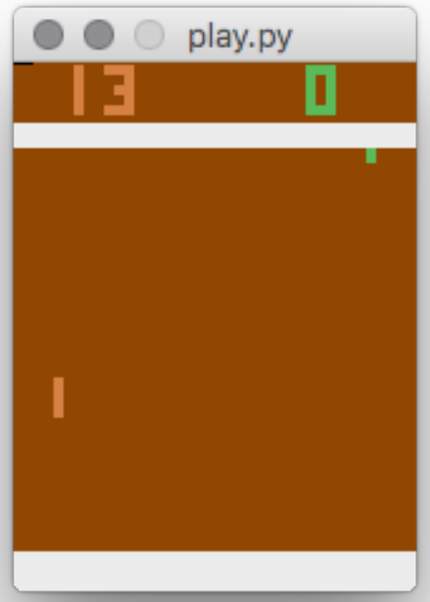
ing to excel at a diverse array of challenging tasks.

We set out to create a single algorithm that would be able to develop a wide range of competencies on a varied range of challenging tasks—a

convolutional neural network shown in Fig. 1, in which θ_i are the parameters (that is, weights) of the Q-network at iteration i. To perform experience replay we store the agent's experiences $e_t = (s_t, a_t, r_t, s_{t+1})$

https://storage.googleapis.com/deepmind-media/dqn/DQNNaturePaper.pdf

Atari 2600

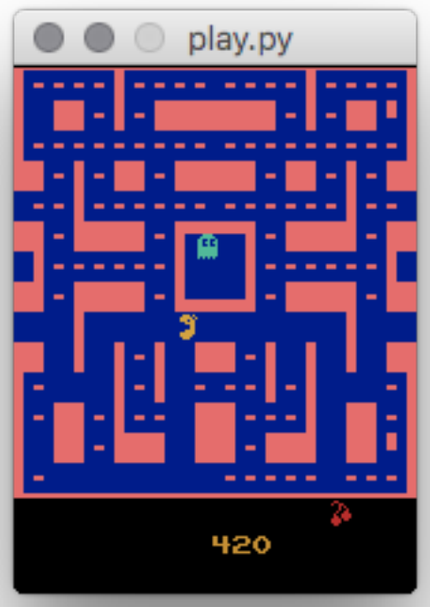


Pong

Atari 2600

Breakout

Ms PacMan



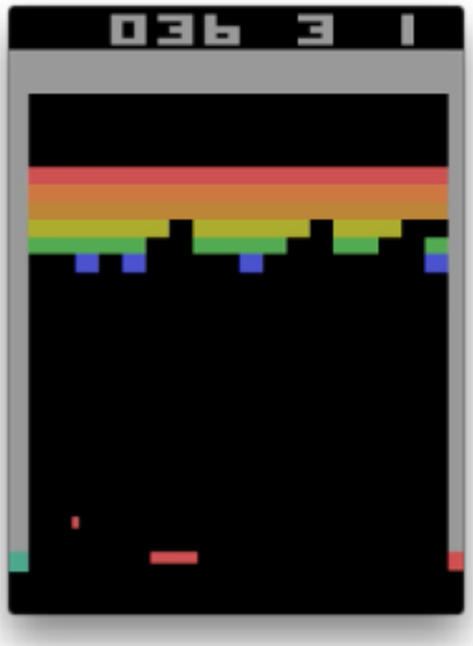
SpaceInvader

https://en.wikipedia.org/wiki/Atari_2600

Peng Brothers Co., Ltd.

Reward Hypothesis

"All goals can be described by the maximisation of expected cumulative reward"



$$r_t = \Delta Distance \qquad r_t = \Delta Score \qquad r_{T-1} = \begin{cases} 1 & , \text{win} \\ -1 & , \text{lose} \end{cases}$$

Cumulative Rewards (Return)

The total discounted rewards from time-step t

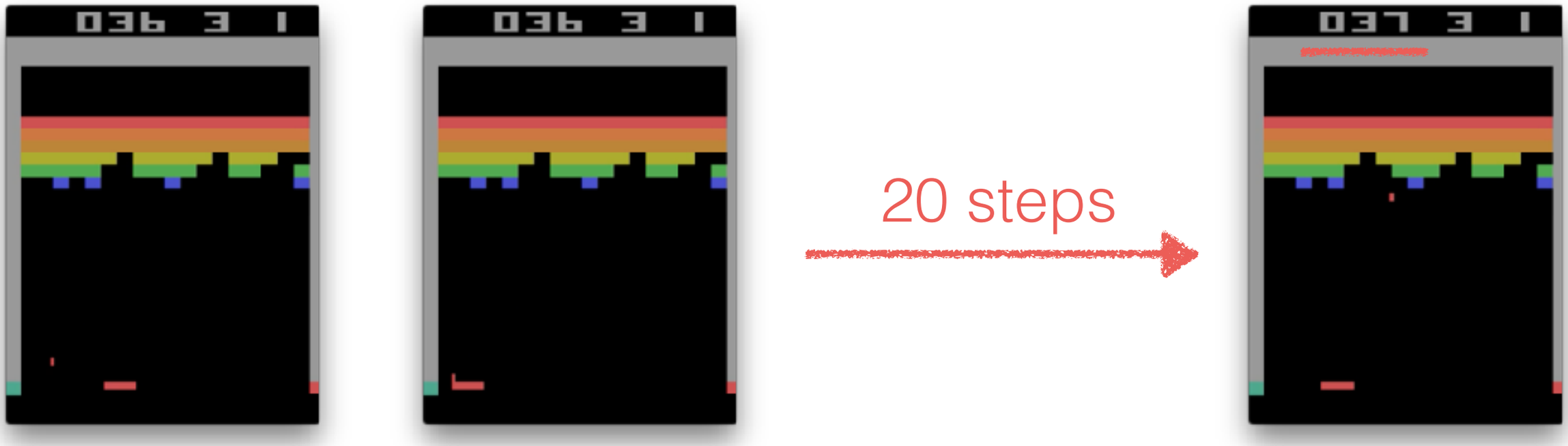
$$G_t = r_t + \gamma r_{t+1} + ... = \sum_{t'=t}^{\infty} \gamma^{t'-t} r_t$$

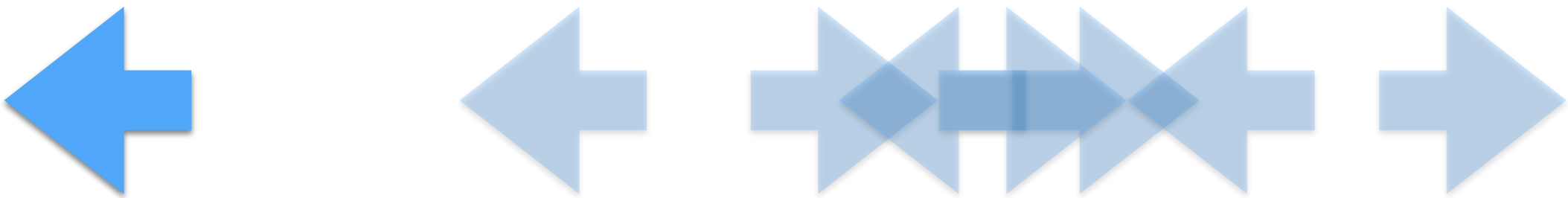
$r_t = 0$ $r_{t+26} = 1$ $r_{t+49} = 0$ $r_{t+65} = 4$ $r_{t+82} = -1$

Credit Assignment Problem

$$r_t = 0$$

$$r_{t+20} = 1$$

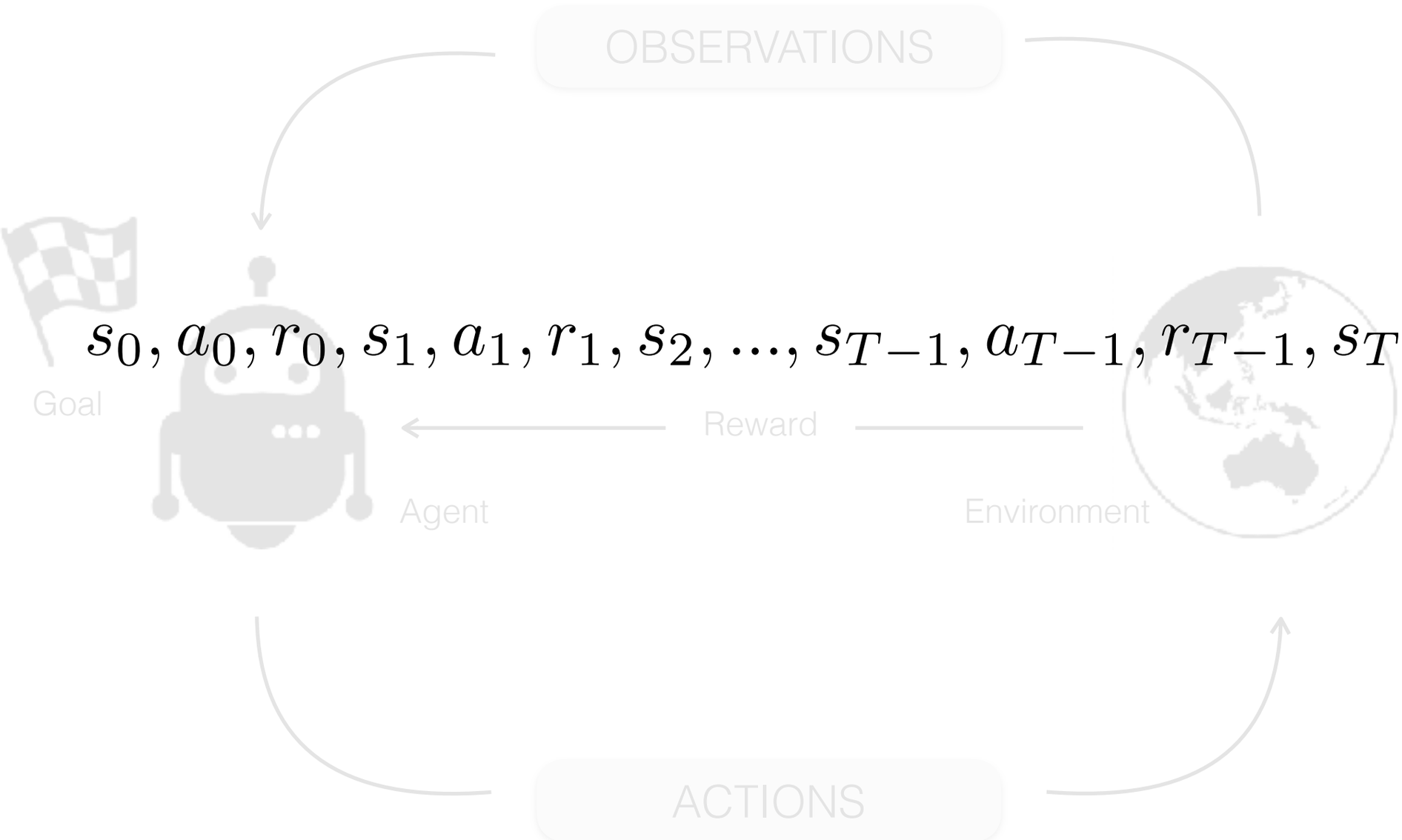




the “right” action

doesn’t really matter

Reinforcement Learning Framework

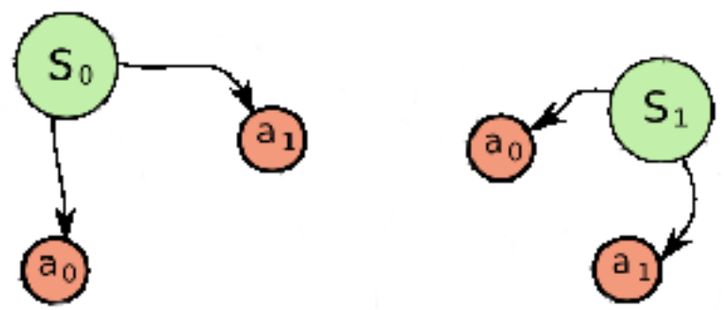


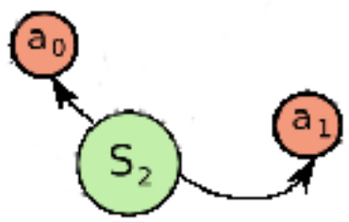
$$s_0, a_0, r_0, s_1, a_1, r_1, s_2, ..., s_{T-1}, a_{T-1}, r_{T-1}, s_T$$

https://www.youtube.com/watch?v=PGDsTEmXNAo

Markov Decision Process

$$\langle \mathcal{S}, \mathcal{A}, \cancel{\mathcal{P}}, \cancel{\mathcal{R}}, \gamma \rangle$$





$\mathcal{S}$ is a finite set of states

$\mathcal{A}$ is a finite set of actions

$\cancel{\mathcal{P}}$ is a state transition probability matrix (model)

$$\cancel{\mathcal{P}^a_{s,s'} = \mathbb{P}\left[s_{t+1} = s' | s_t = s, a_t = a\right]}$$

$\cancel{\mathcal{R}}$ is a reward function (immediate reward)

$$\cancel{\mathcal{R}^a_s = \mathbb{E}\left[r_t | s_t = s, a_t = a\right]}$$

γ is a discount factor (typically 0.99)

$$\gamma \in [0, 1]$$

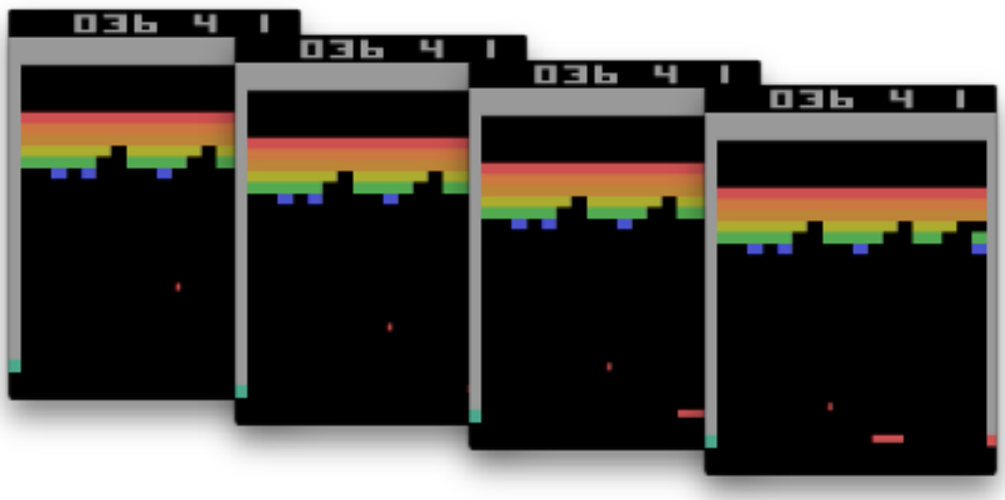
https://en.wikipedia.org/wiki/Markov_decision_process

Markov Sate

Markov state contains all useful information from the history

$$\mathbb{P}\left[s_{t+1}|s_t\right] = \mathbb{P}\left[s_{t+1}|s_1, ...s_t\right]$$

"The future is independent of the past given the present"



Stacking 4 previous frames
to "approximate" Markov state

https://en.wikipedia.org/wiki/Markov_decision_process

Policy

A distribution over actions given states (map from state to action)

$$\pi(a|s) = \mathbb{P}\left[a_t = a | s_t = s\right]$$

Deterministic policy

$$a = \pi(s)$$

A policy fully defines the behaviour of an agent

https://en.wikipedia.org/wiki/Markov_decision_process

Action-Value Function

Expected total future rewards starting from a state s
taking action a and then following policy π

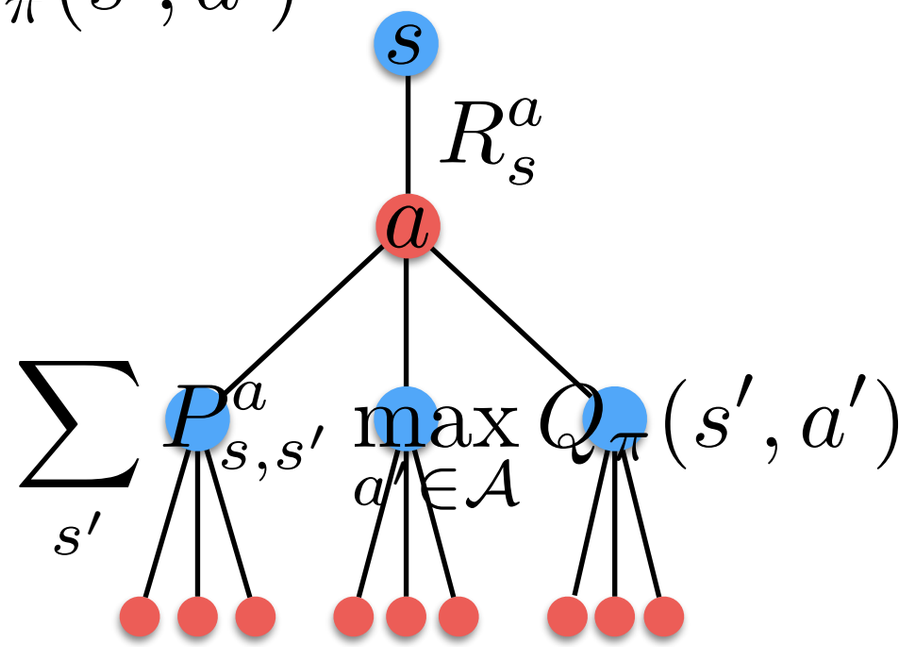
$$Q_\pi(s,a) = \mathbb{E}_\pi \left[r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + ... | s_t = s, a_t = a \right]$$

$$= \mathbb{E}_\pi \left[r_t + \gamma Q_\pi(s', a') | s_t = s, a_t = a \right]$$

$$= \mathcal{R}_s^a + \gamma \sum_{s'} P_{s,s'}^a \max_{a' \in \mathcal{A}} Q_\pi(s', a')$$

Optimal action-value function is the maximum
action-value function over all policies

$$Q^*(s,a) = \max_\pi Q_\pi(s,a)$$



https://en.wikipedia.org/wiki/Markov_decision_process

Optimal Policy

There is always a deterministic optimal policy for any MDP

$$\pi^*(a|s) = \begin{cases} 1 & \text{if } a = \arg\max_{a \in \mathcal{A}} Q^*(s,a) \\ 0 & \text{otherwise} \end{cases}$$

Deterministic policy

$$a = \pi(s) = \arg\max_{a \in \mathcal{A}} Q^*(s,a)$$

If we know $Q^*(s,a)$, we immediately have the optimal policy.

https://en.wikipedia.org/wiki/Markov_decision_process

Bellman Equation

Learning Optimal Action-Value Function

$$Q_{i+1}(s,a) = \mathbb{E}_{s'}[r + \gamma \max_{a' \in \mathcal{A}} Q_i(s',a')|s,a]$$

$$= \mathcal{R}_s^a + \gamma \sum_{s'} \mathcal{P}_{s,s'}^a \max_{a' \in \mathcal{A}} Q_i(s',a')$$

Iterating infinitely with randomized initial values is guaranteed to converge Q to the optimal action-value function Q^* (under certain assumptions)

$$\lim_{i \to \infty} Q_i = Q^*$$

https://en.wikipedia.org/wiki/Markov_decision_process

Temperal Difference Q-Learning

Estimating the summation over different new states according to their probabilities using leaky integration (EMA)

"Computing the average as we go"

$$Q_{i+1}(s,a) = (1-\alpha)Q_i(s,a) + \alpha(r + \gamma \max_{a' \in \mathcal{A}} Q_i(s',a'))$$

$$= Q_i(s,a) + \alpha(r + \gamma \max_{a' \in \mathcal{A}} Q_i(s',a') - Q_t(s,a))$$

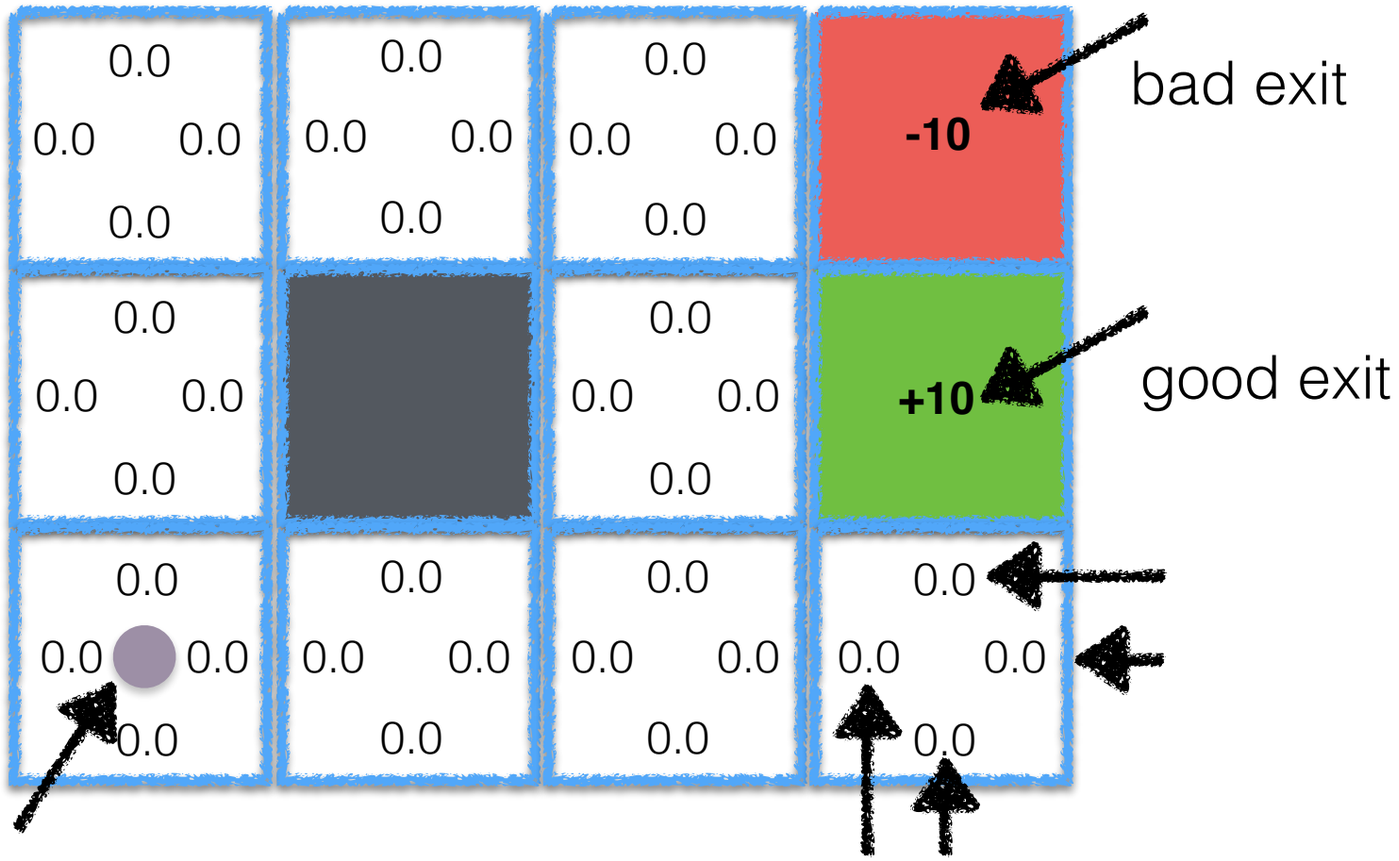
The parameter $\alpha \in [0,1]$ (typically 0.1) determines how fast the system adapts to new samples

https://en.wikipedia.org/wiki/Markov_decision_process

Grid World Example

$\gamma = 1.0$

$\alpha = 0.5$



$\mathcal{R} = -1.0$

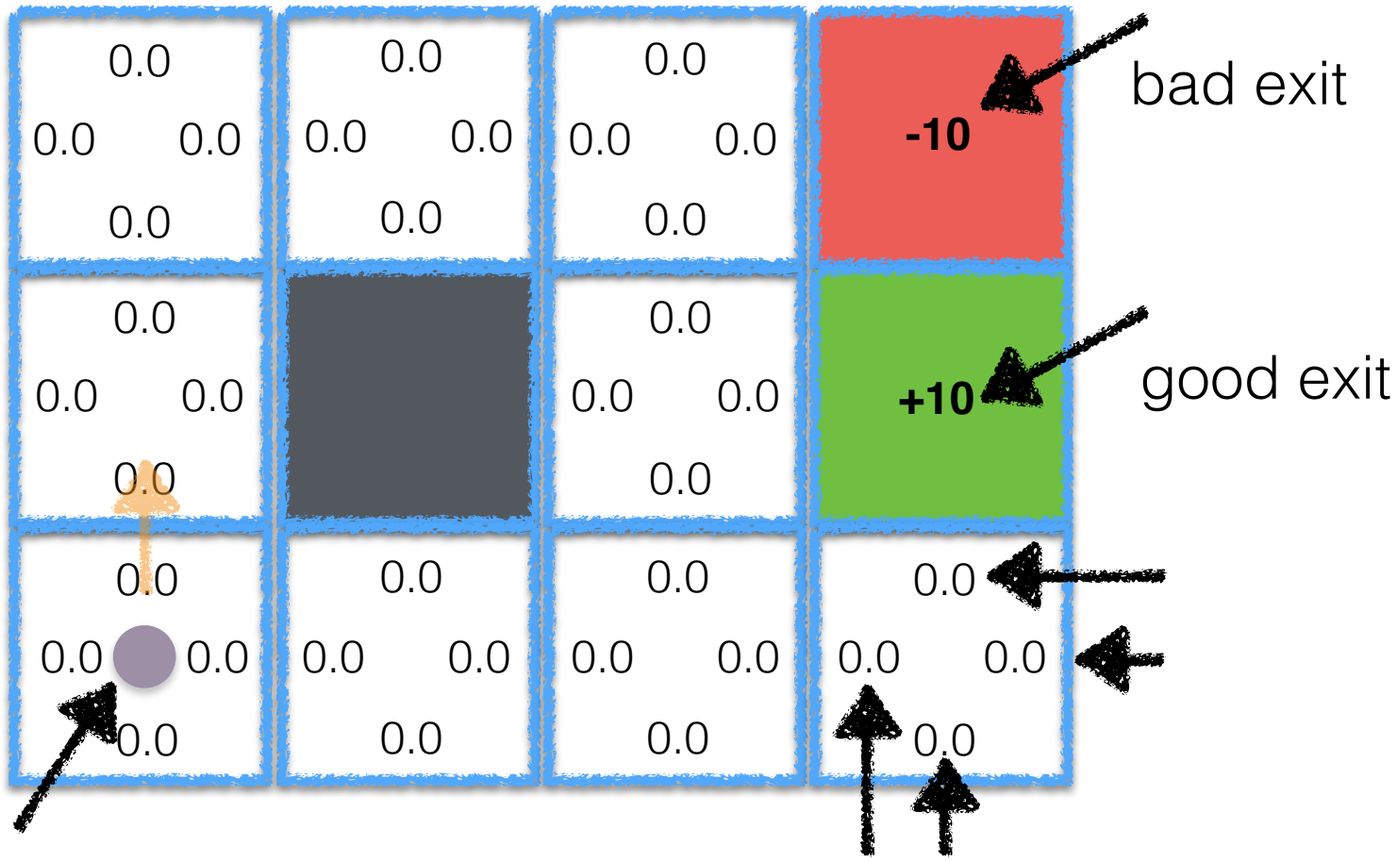
$\mathcal{A} = \{N, E, W, S\}$

$t = 0$

Grid World Example

$\gamma = 1.0$

$\alpha = 0.5$

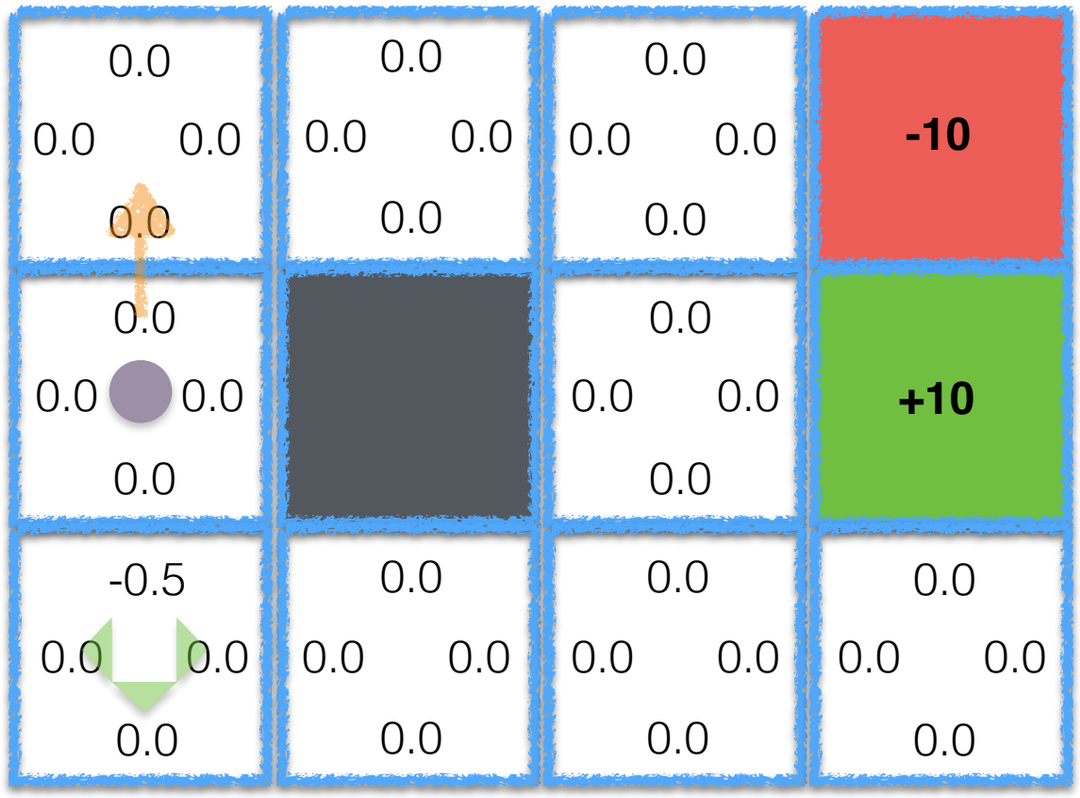


$\mathcal{R} = -1.0$

$\mathcal{A} = \{N, E, W, S\}$

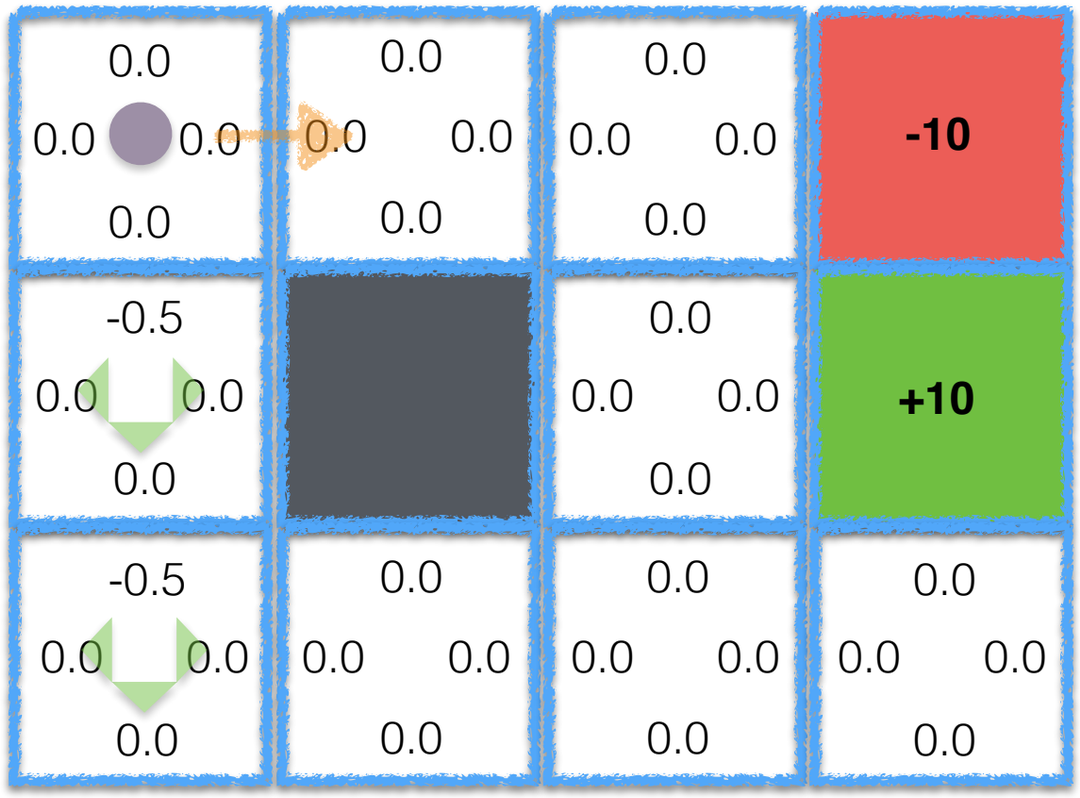
$t = 0$

Grid World Example



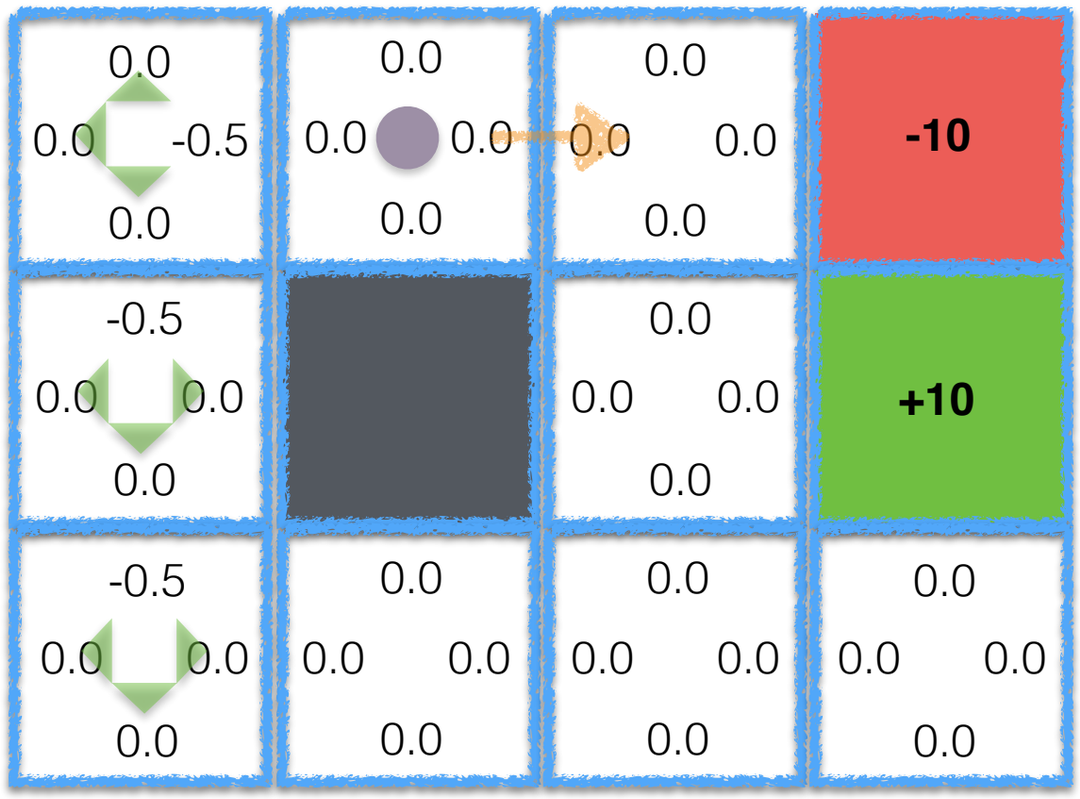
$t = 1$

Grid World Example



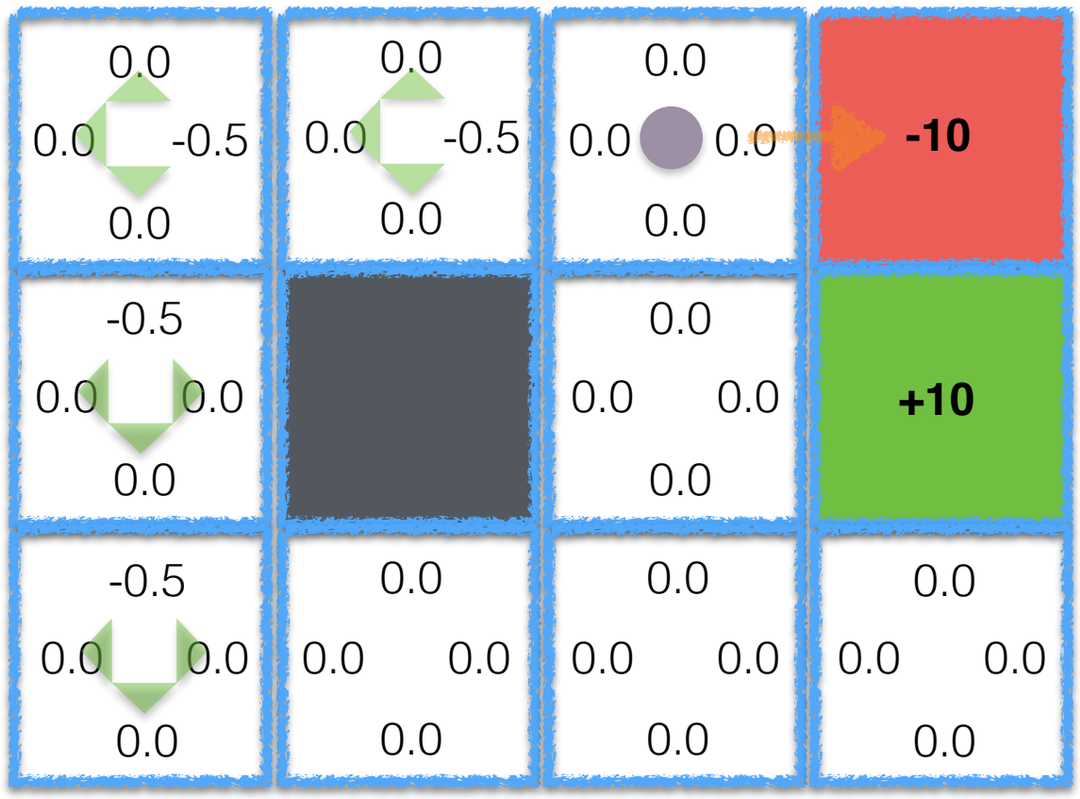
$t = 2$

Grid World Example



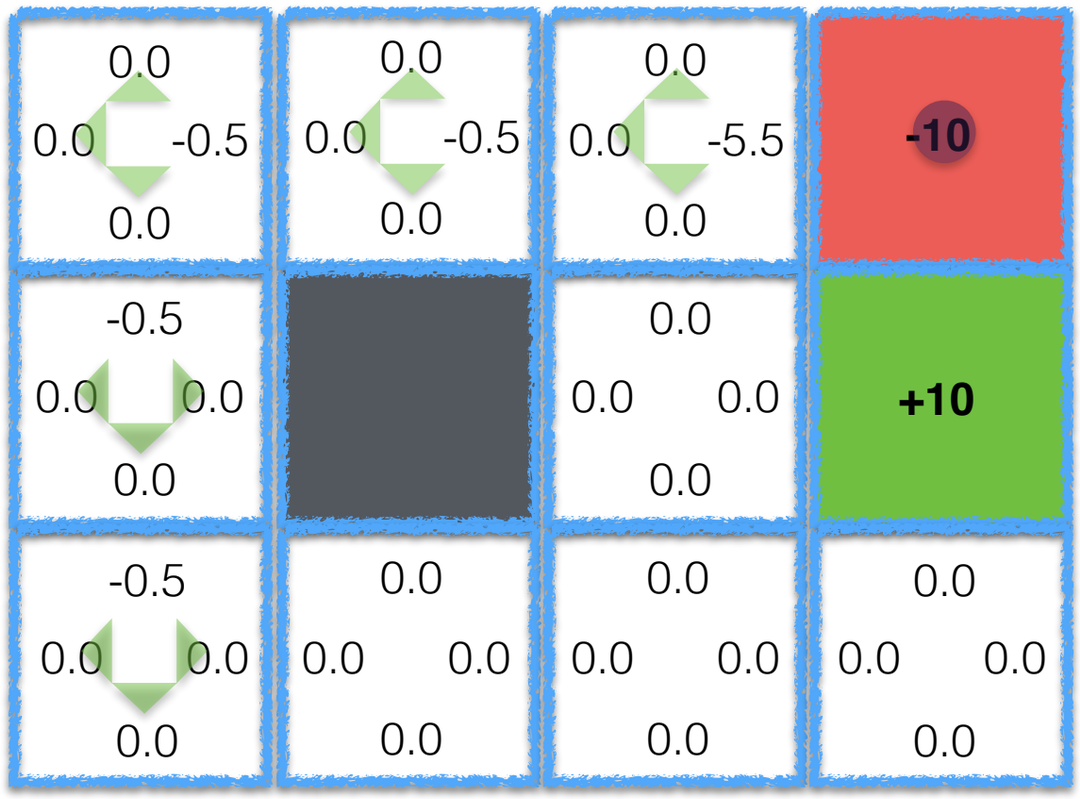
$t = 3$

Grid World Example



$t = 4$

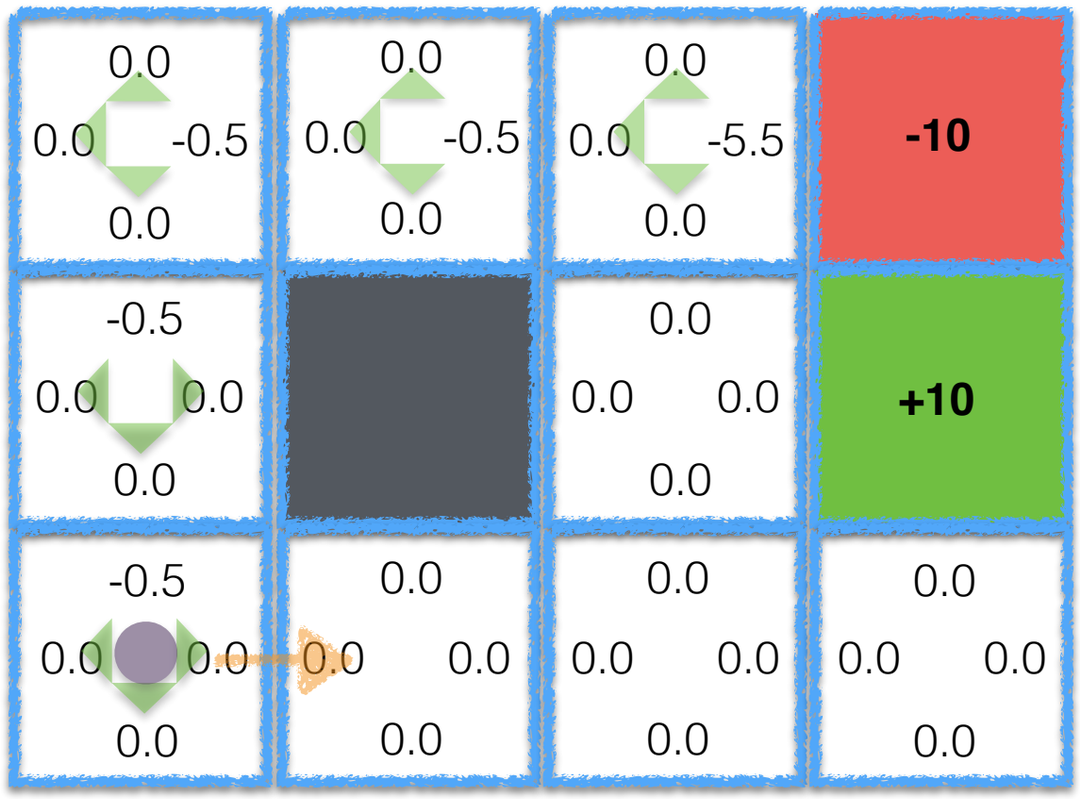
Grid World Example



$terminal = true$

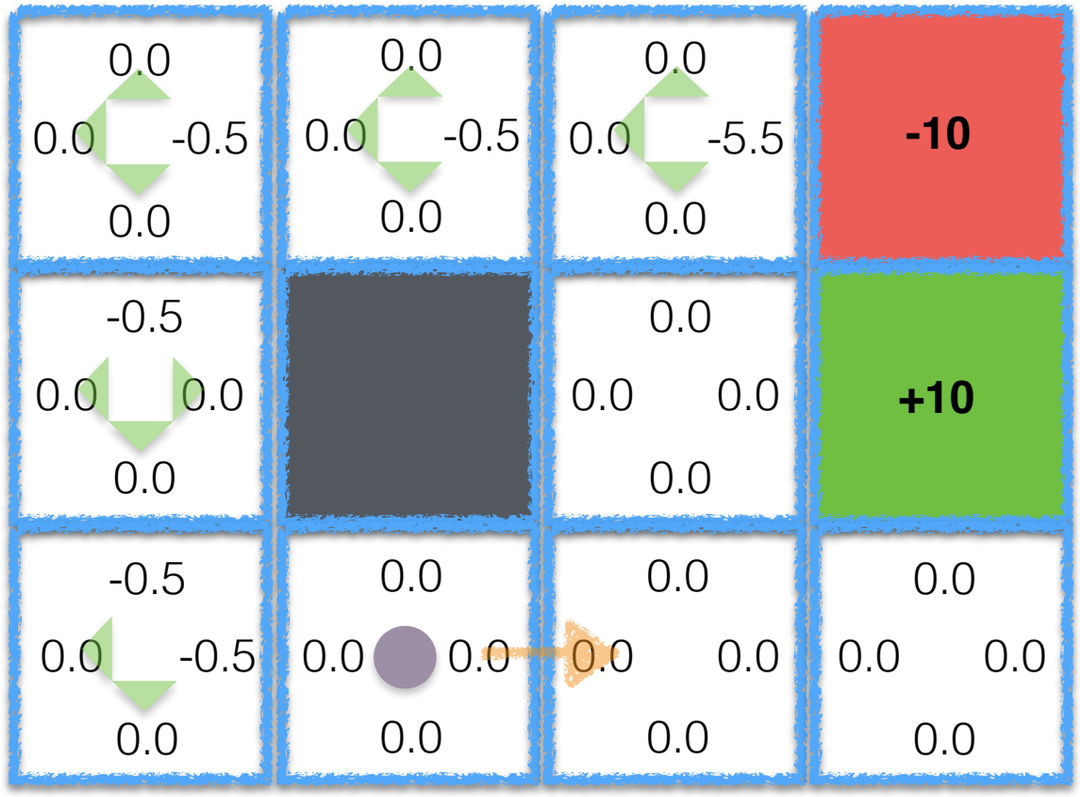
$t = 5$

Grid World Example



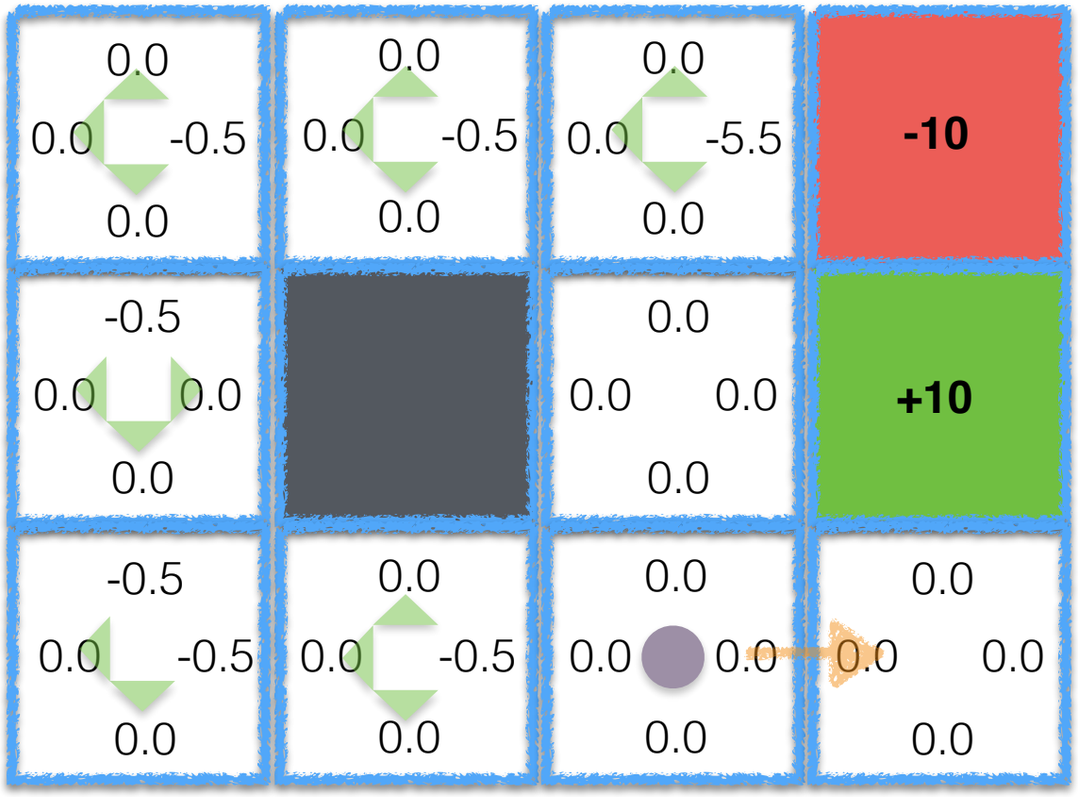
$t = 6$

Grid World Example



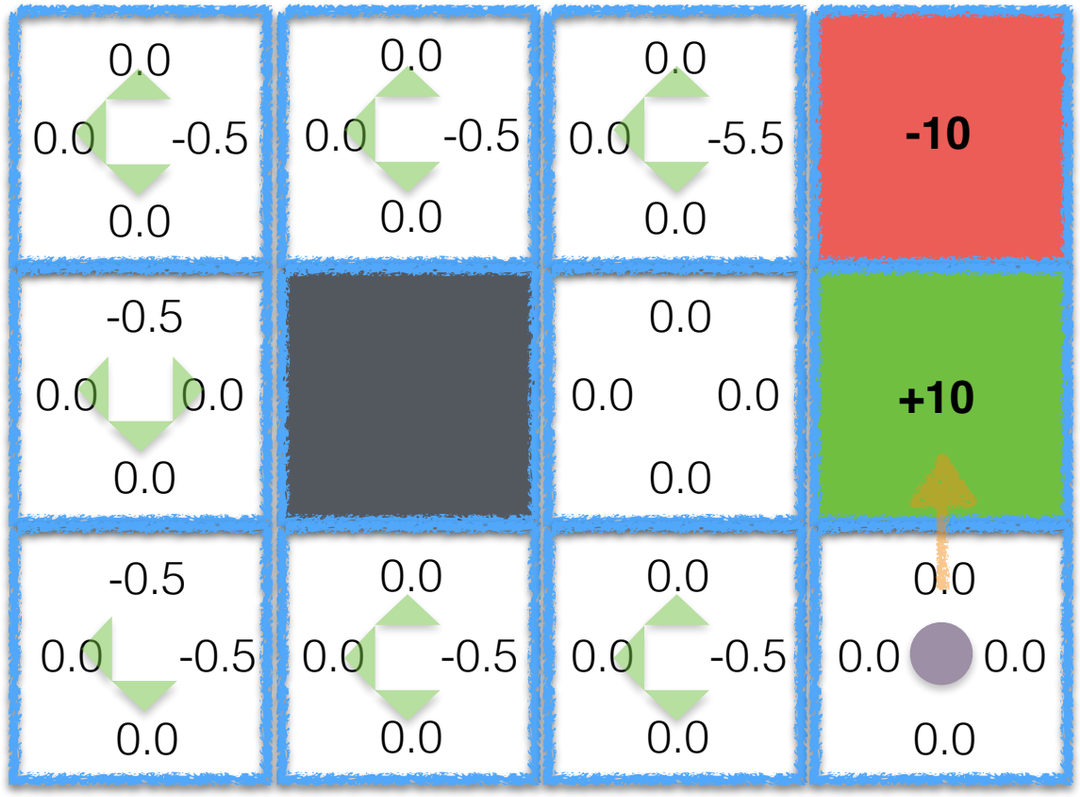
$t = 7$

Grid World Example



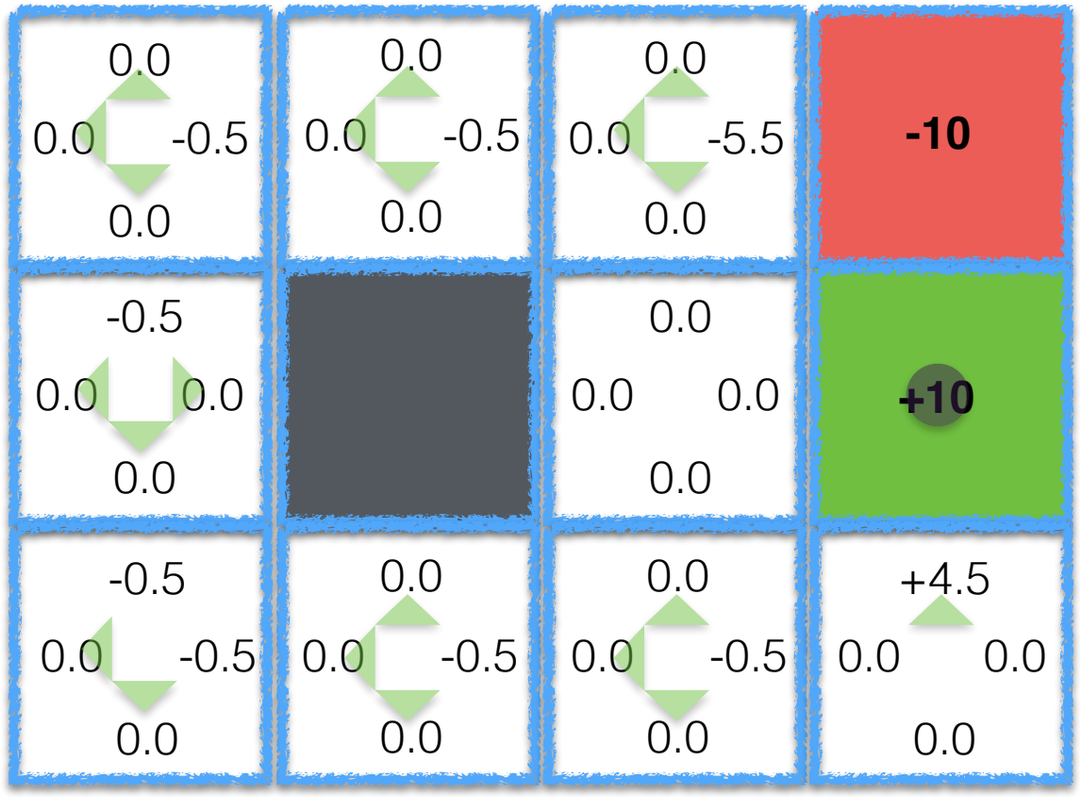
$t = 8$

Grid World Example



$t = 9$

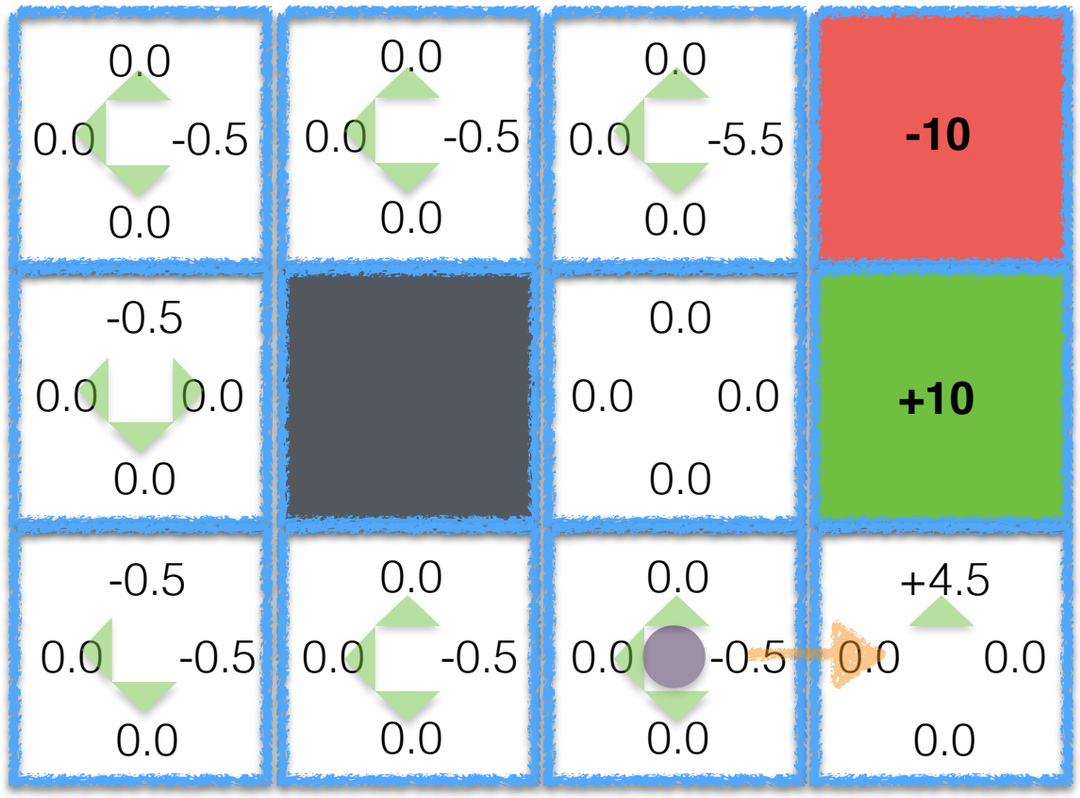
Grid World Example



$terminal = true$

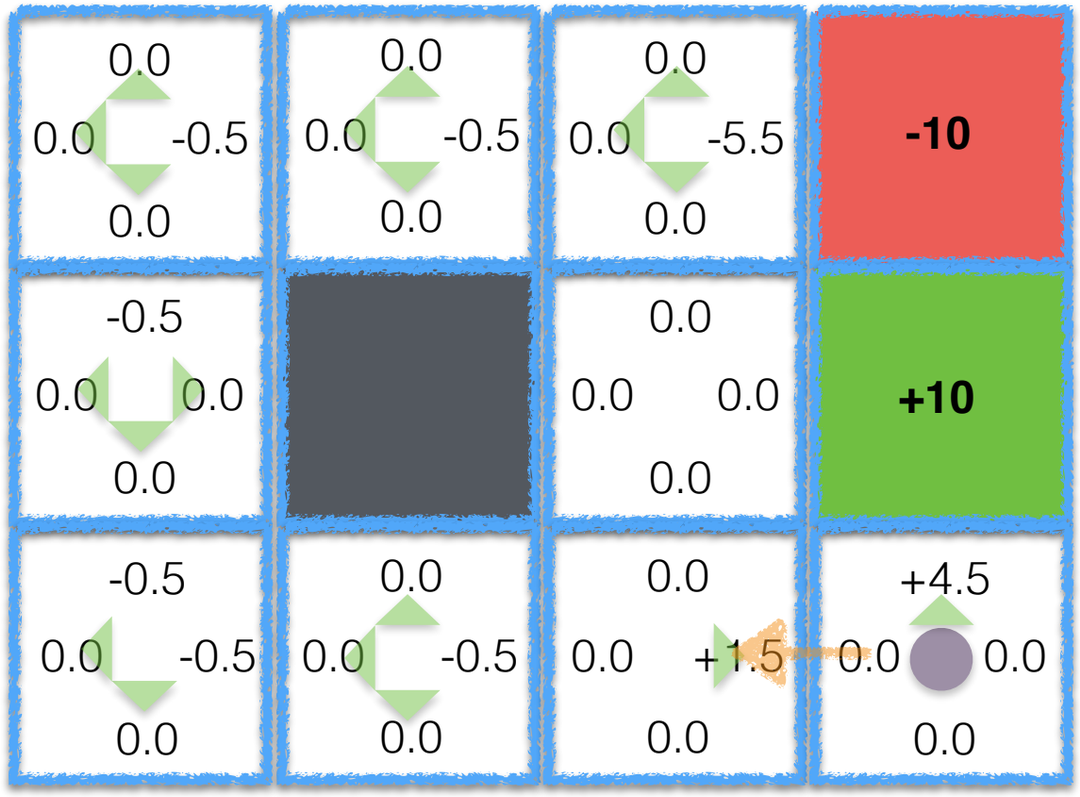
$t = 10$

Grid World Example



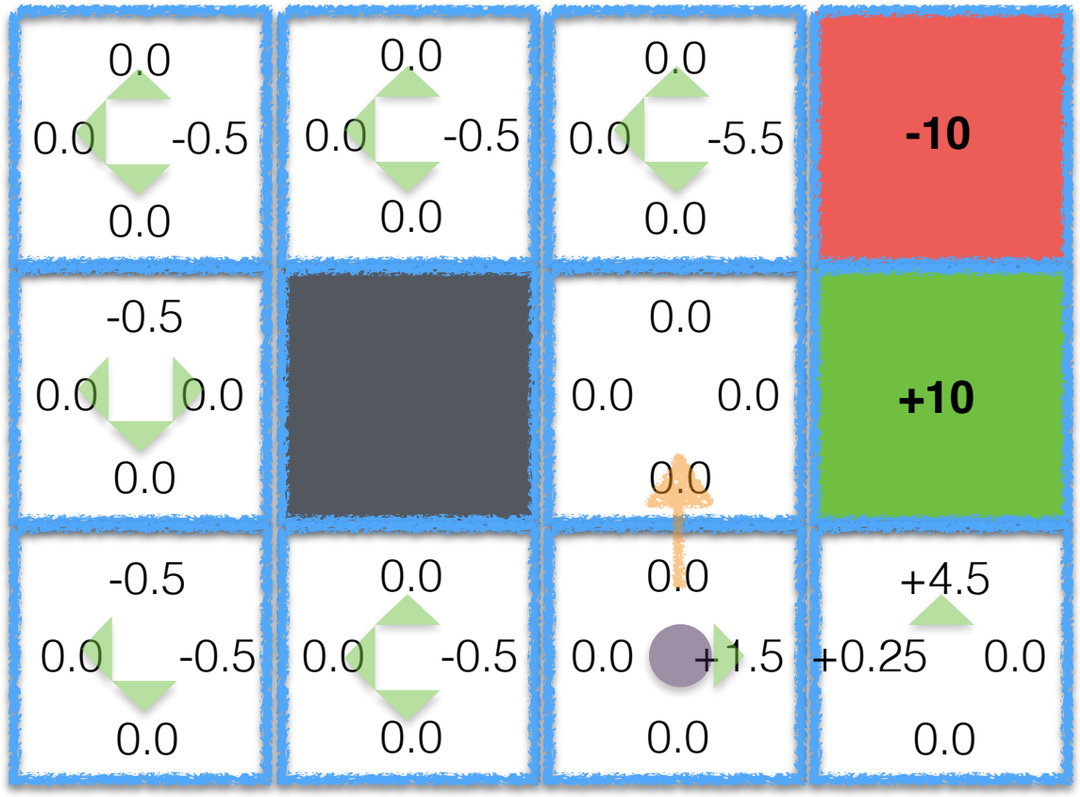
$t = 11$

Grid World Example



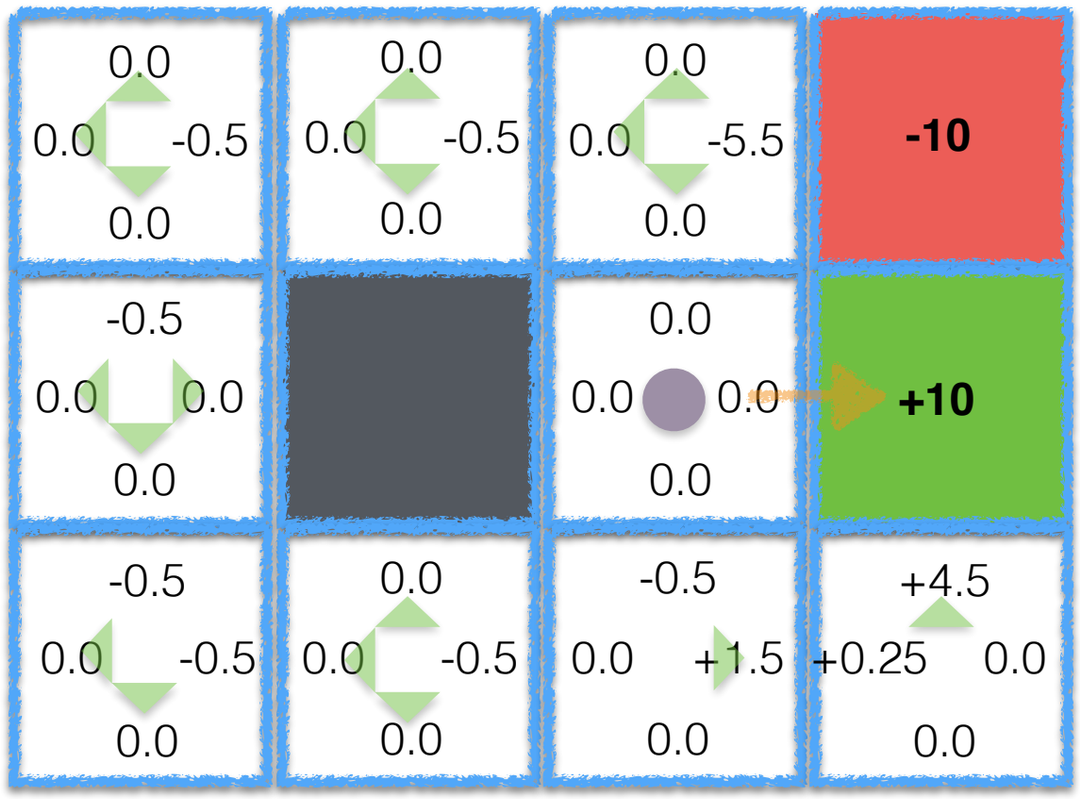
$t = 12$

Grid World Example



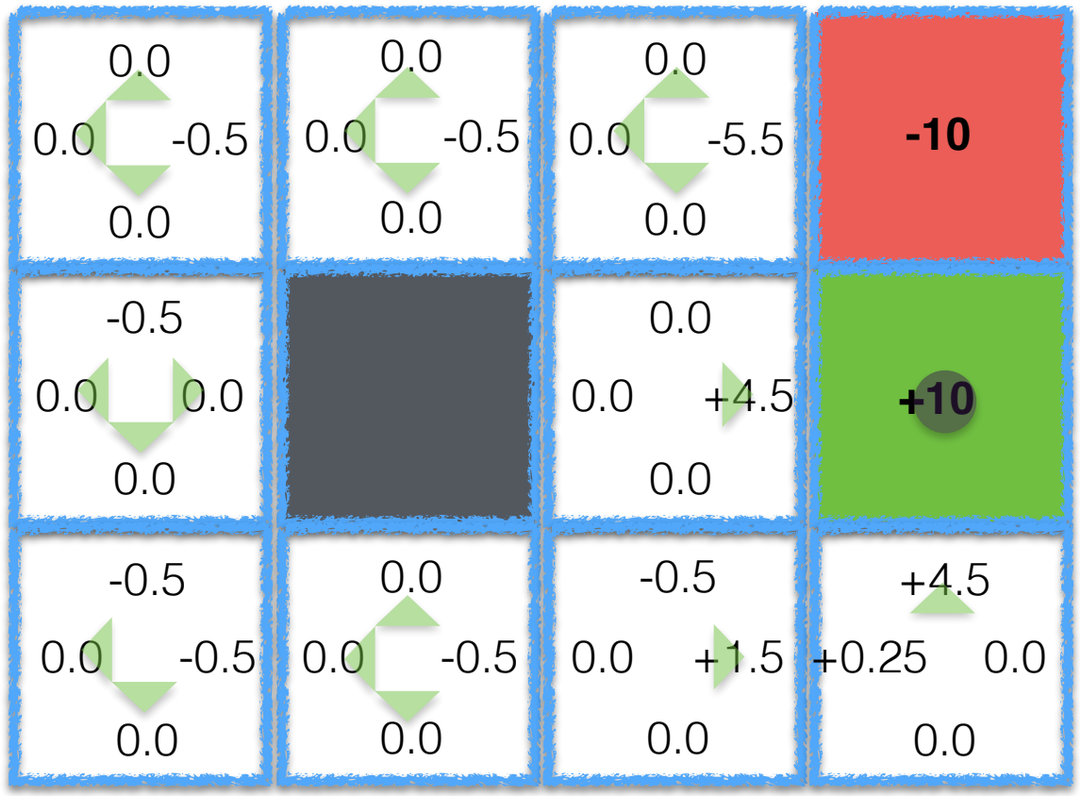
$t = 13$

Grid World Example



$t = 14$

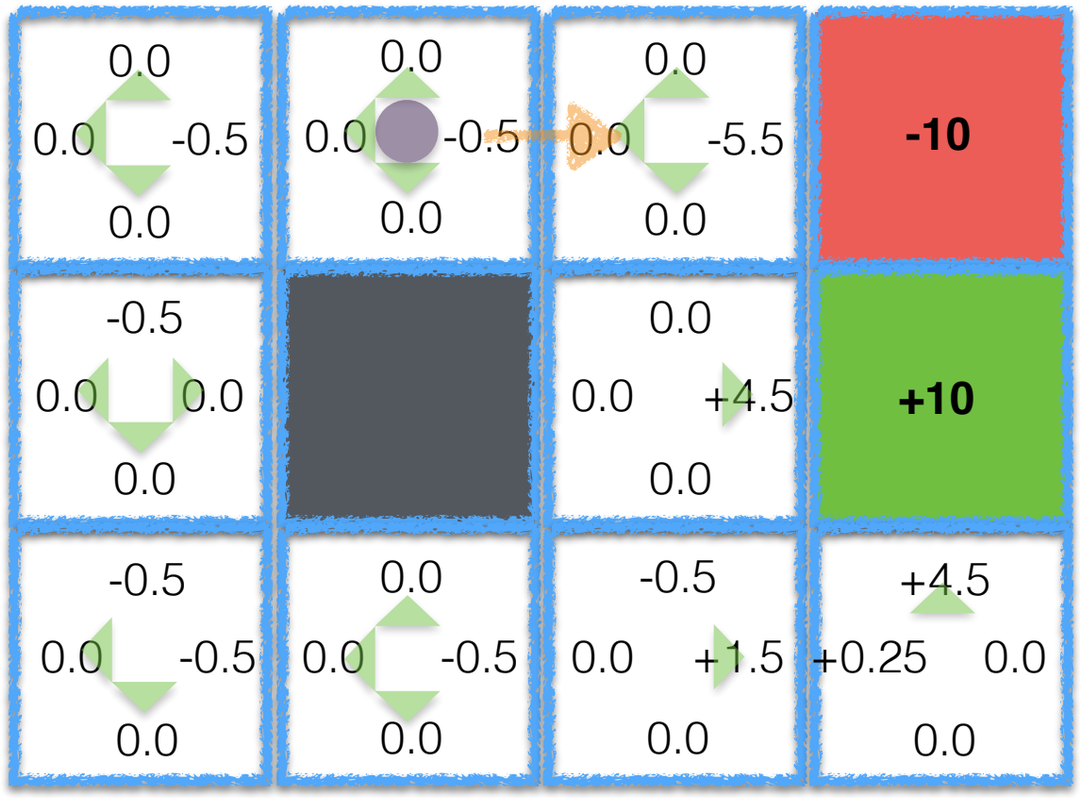
Grid World Example



$terminal = true$

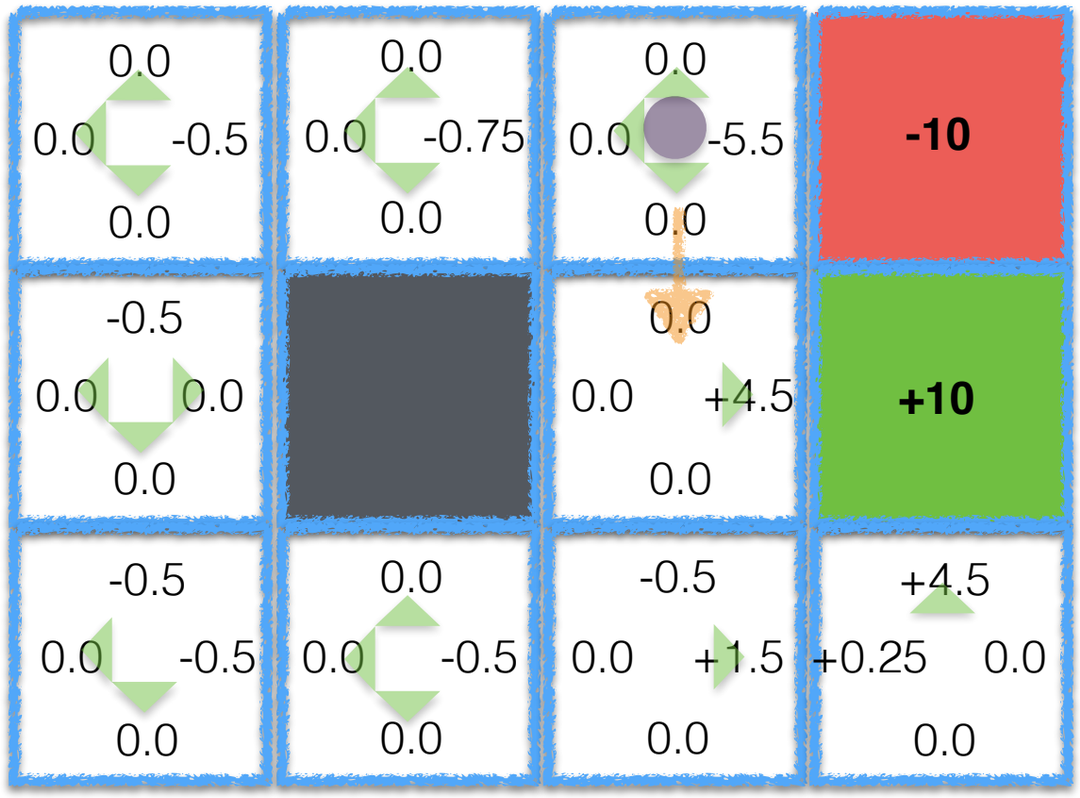
$t = 15$

Grid World Example



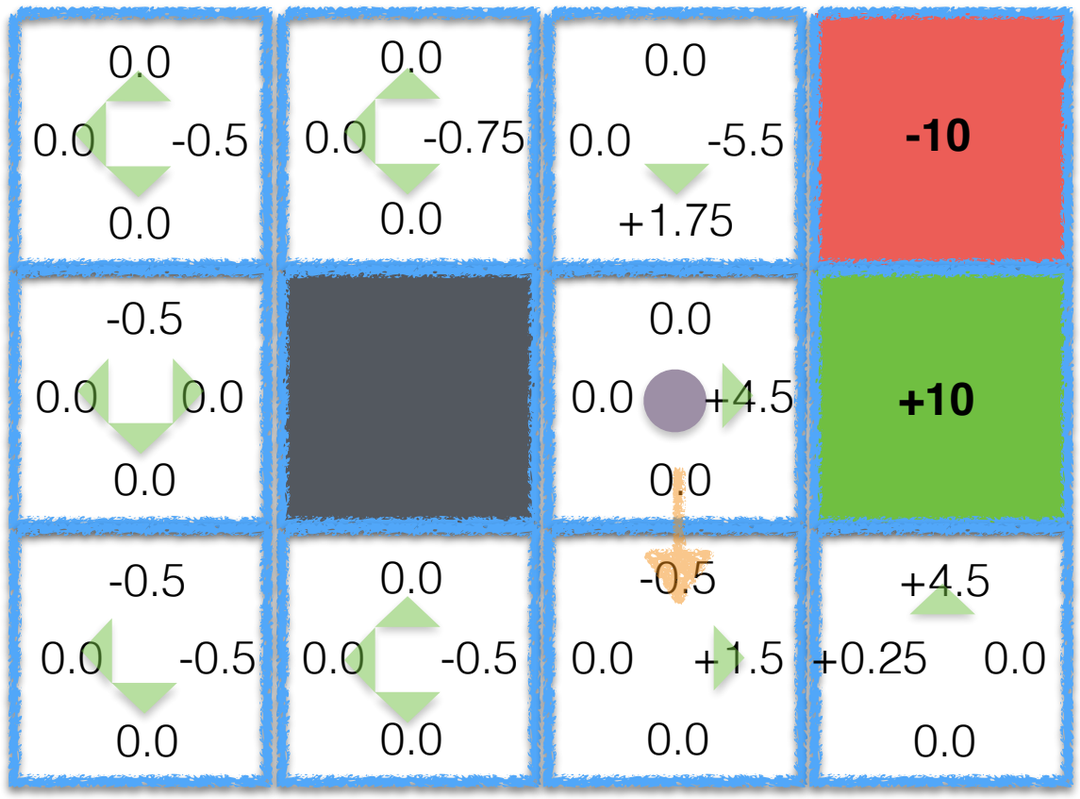
$t = 16$

Grid World Example



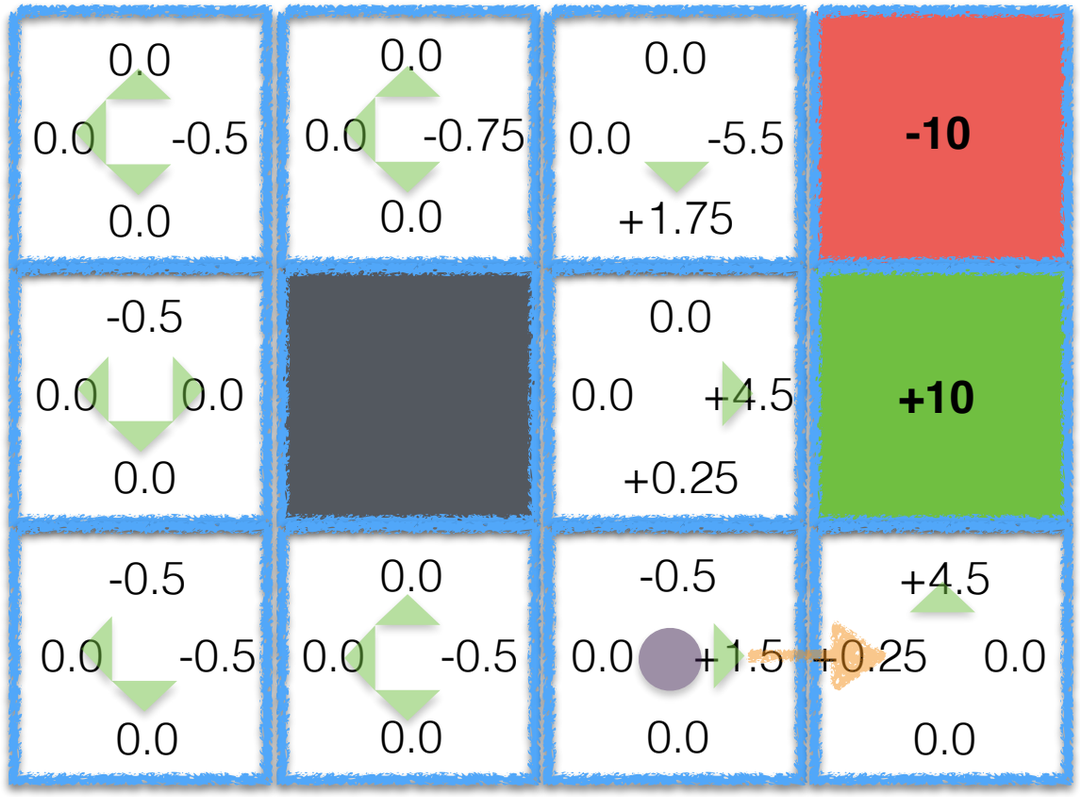
$t = 17$

Grid World Example



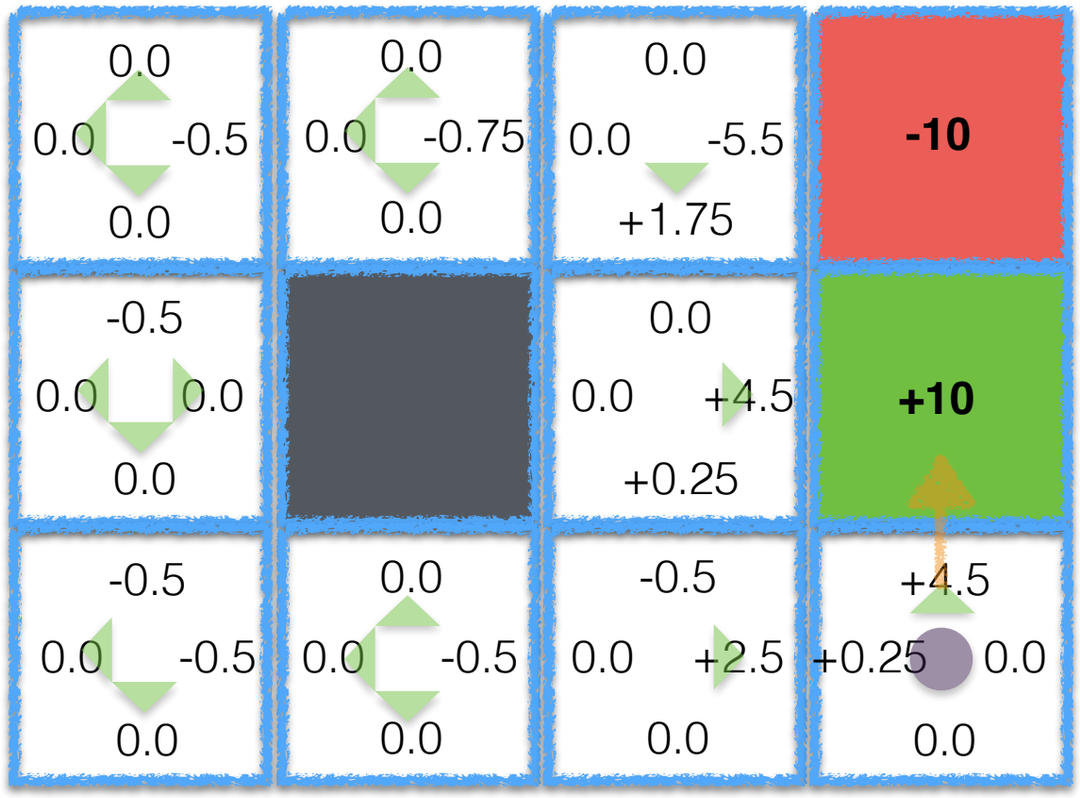
$t = 18$

Grid World Example



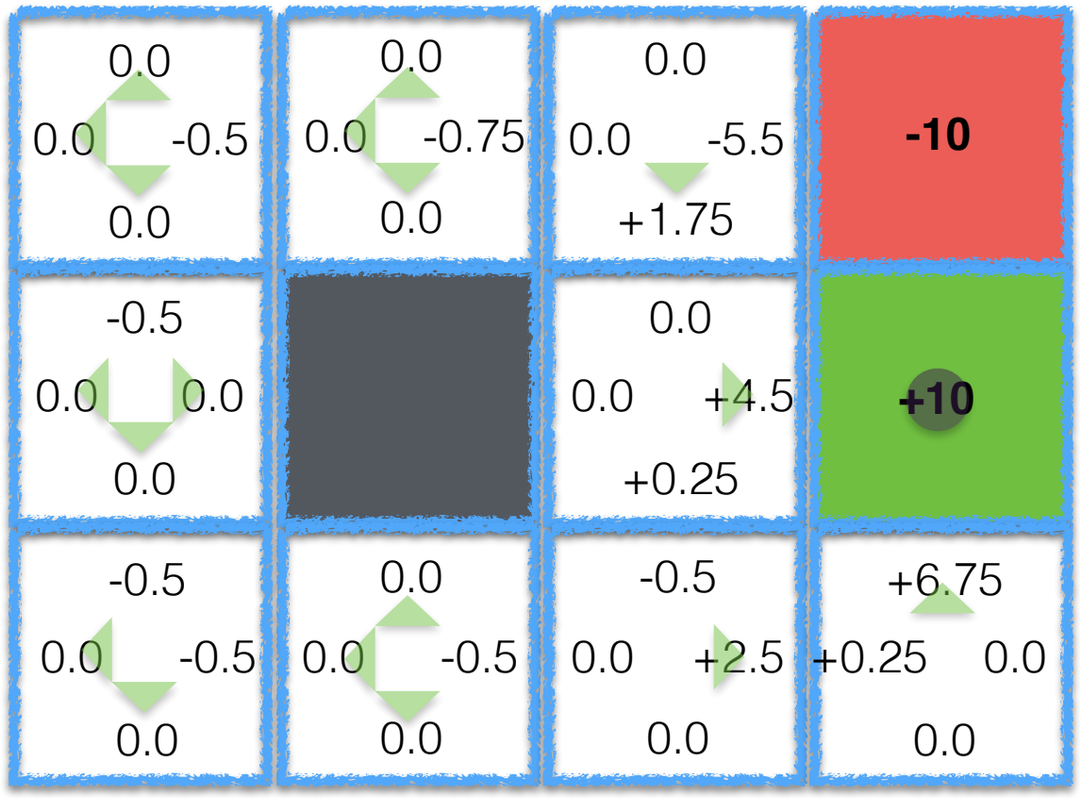
$t = 19$

Grid World Example



$t = 20$

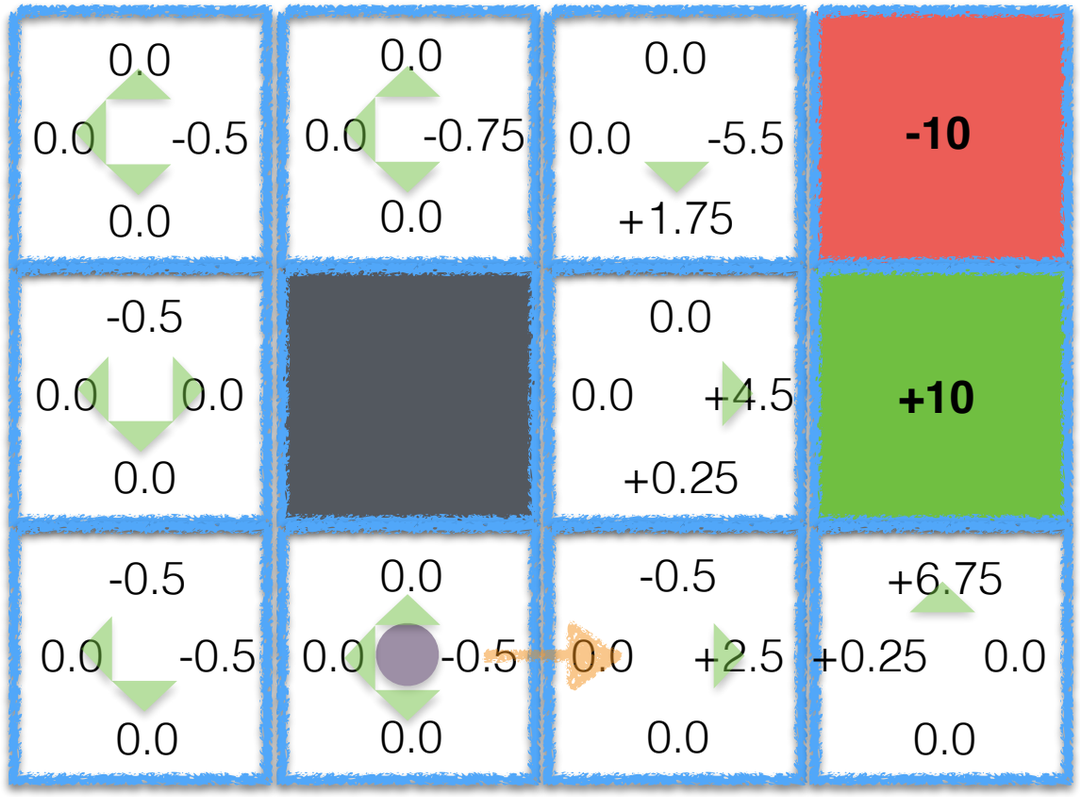
Grid World Example



$terminal = true$

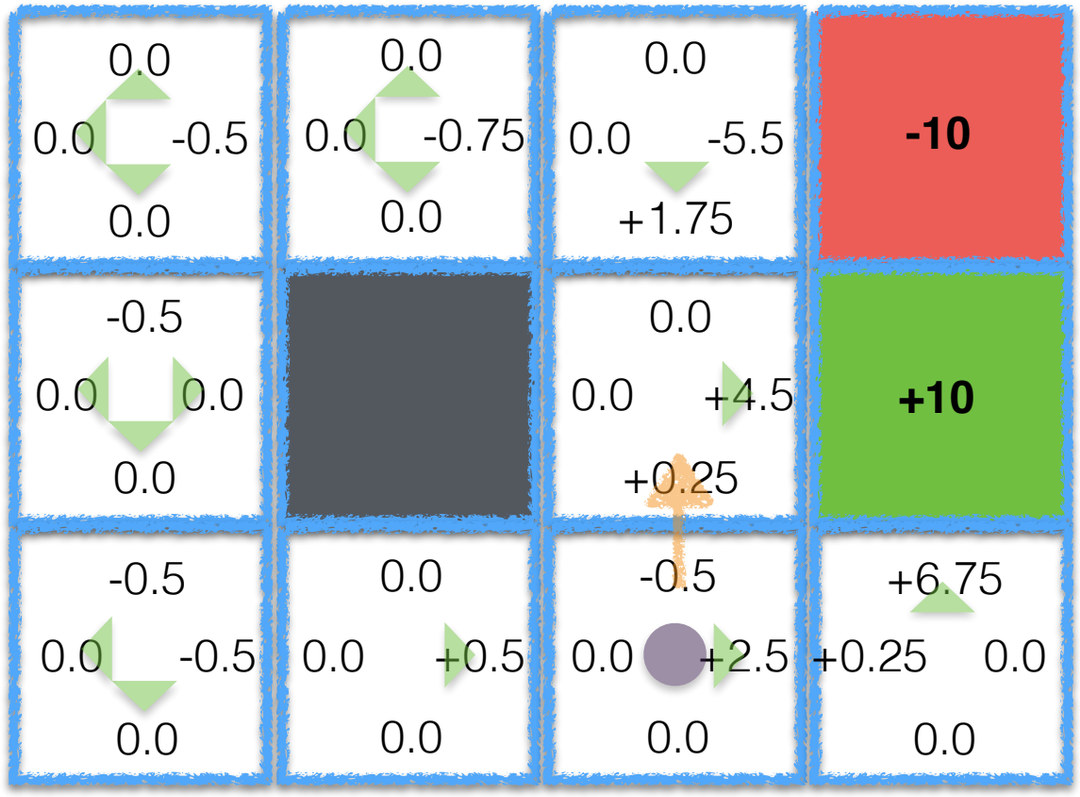
$t = 21$

Grid World Example



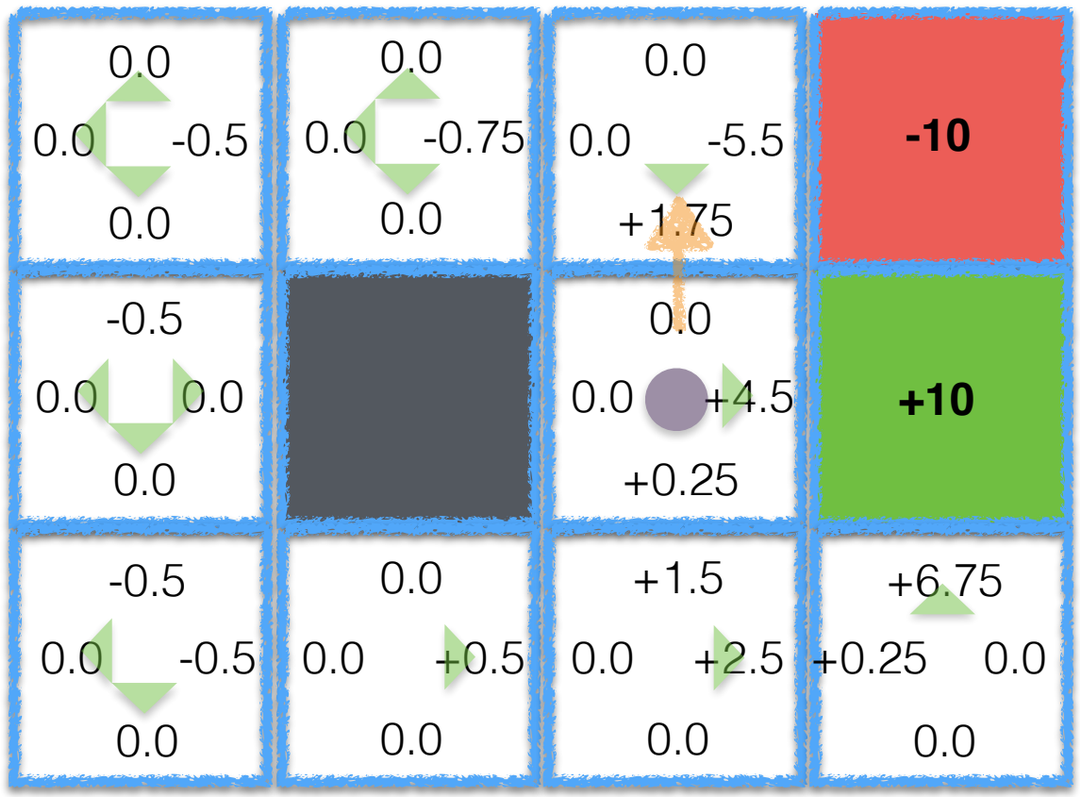
$t = 22$

Grid World Example



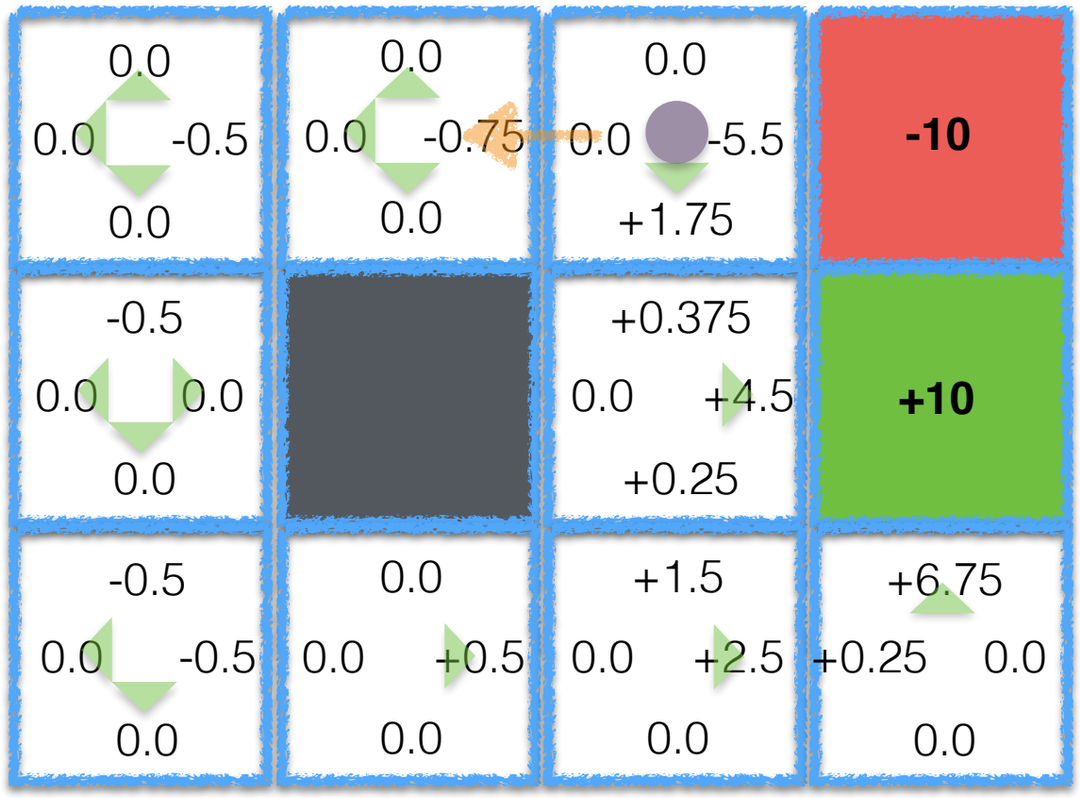
$t = 23$

Grid World Example



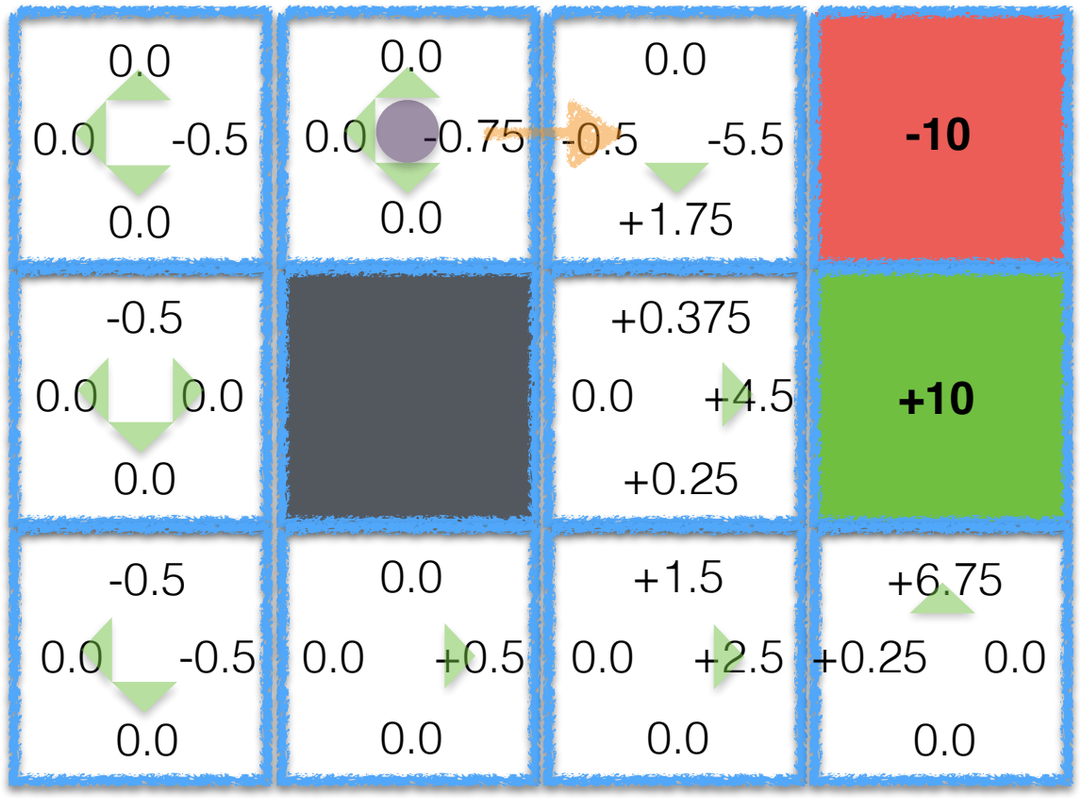
$t = 24$

Grid World Example



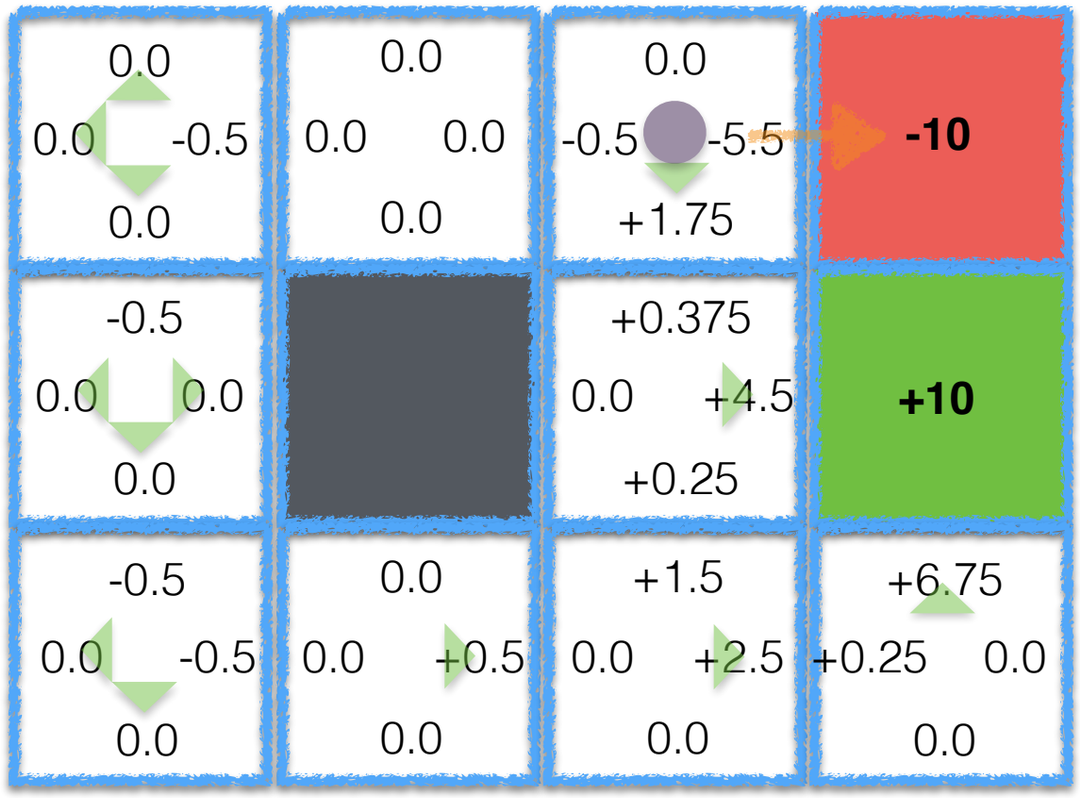
$t = 25$

Grid World Example



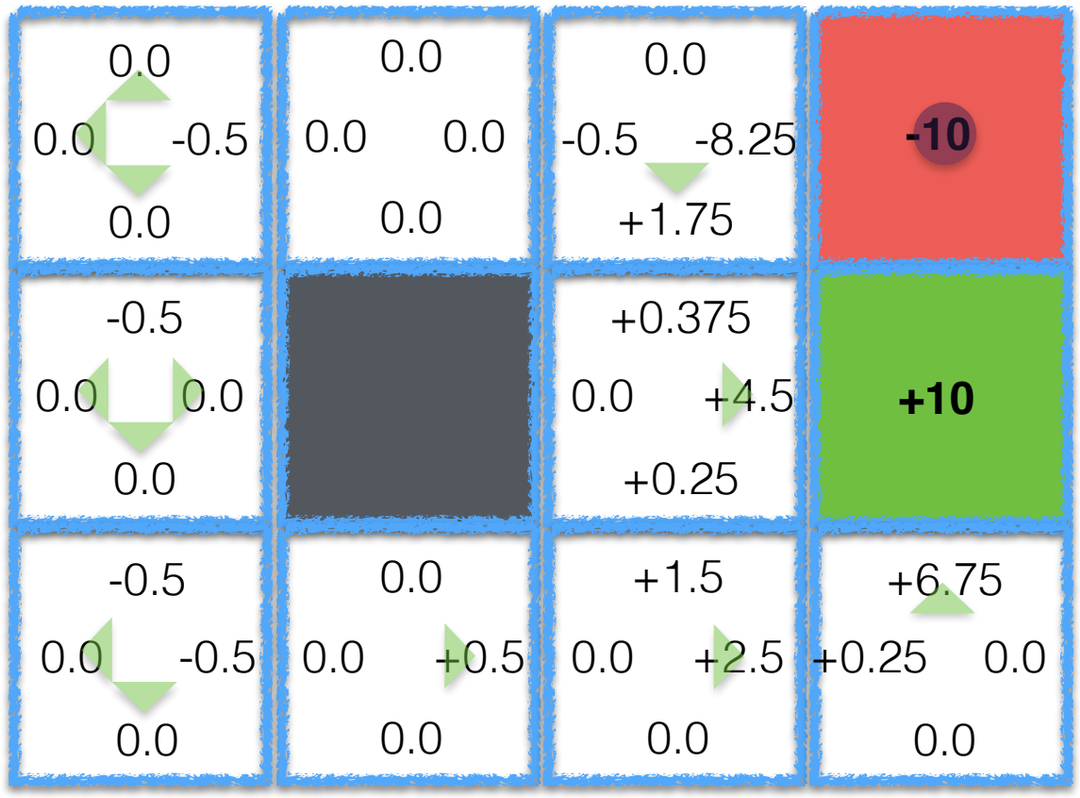
$t = 26$

Grid World Example



$t = 27$

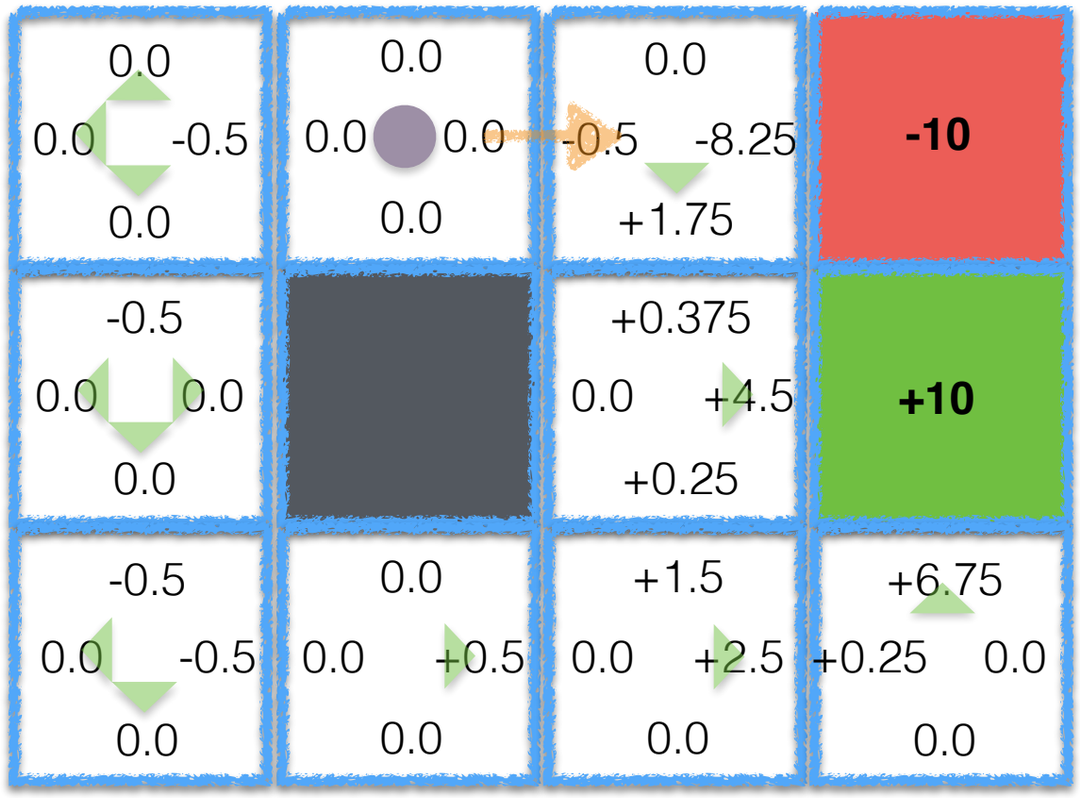
Grid World Example



$$terminal = true$$

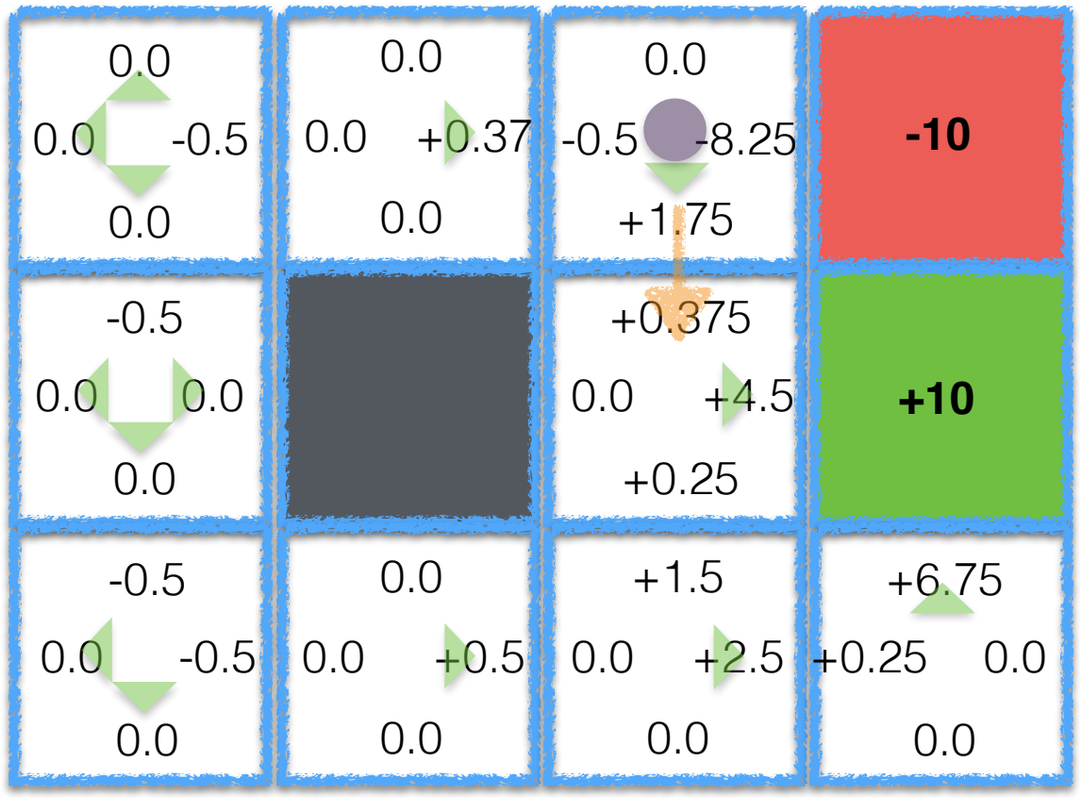
$$t = 28$$

Grid World Example



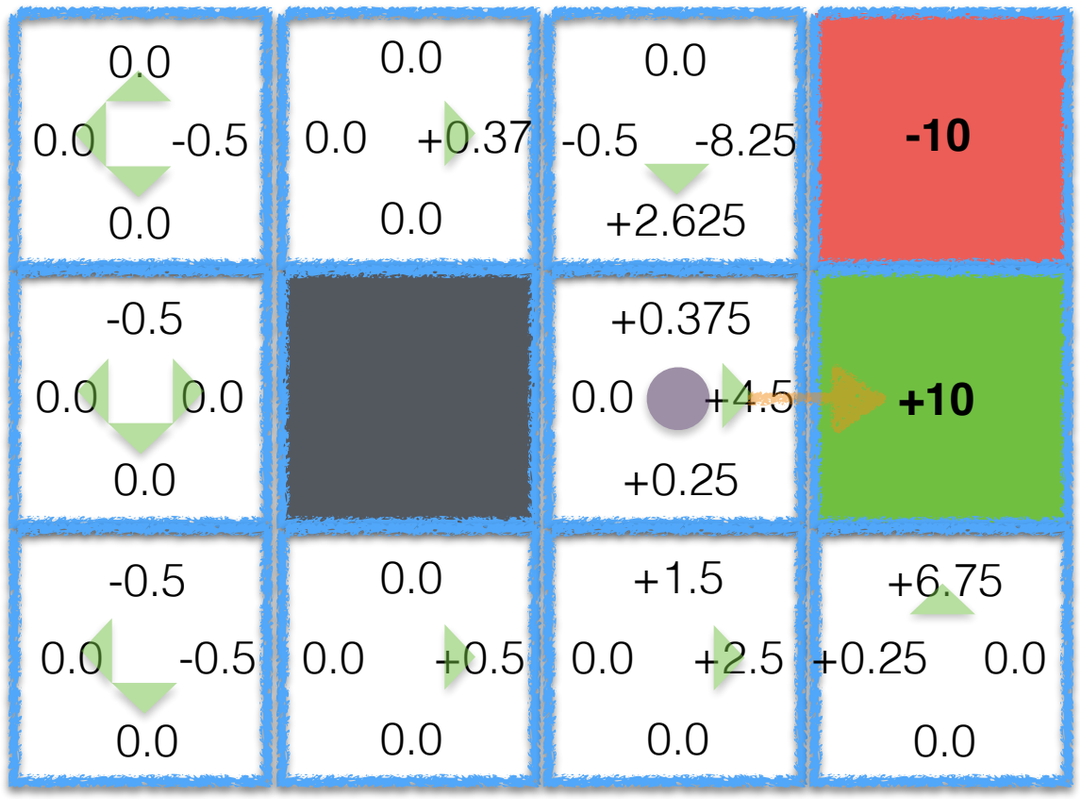
$t = 29$

Grid World Example



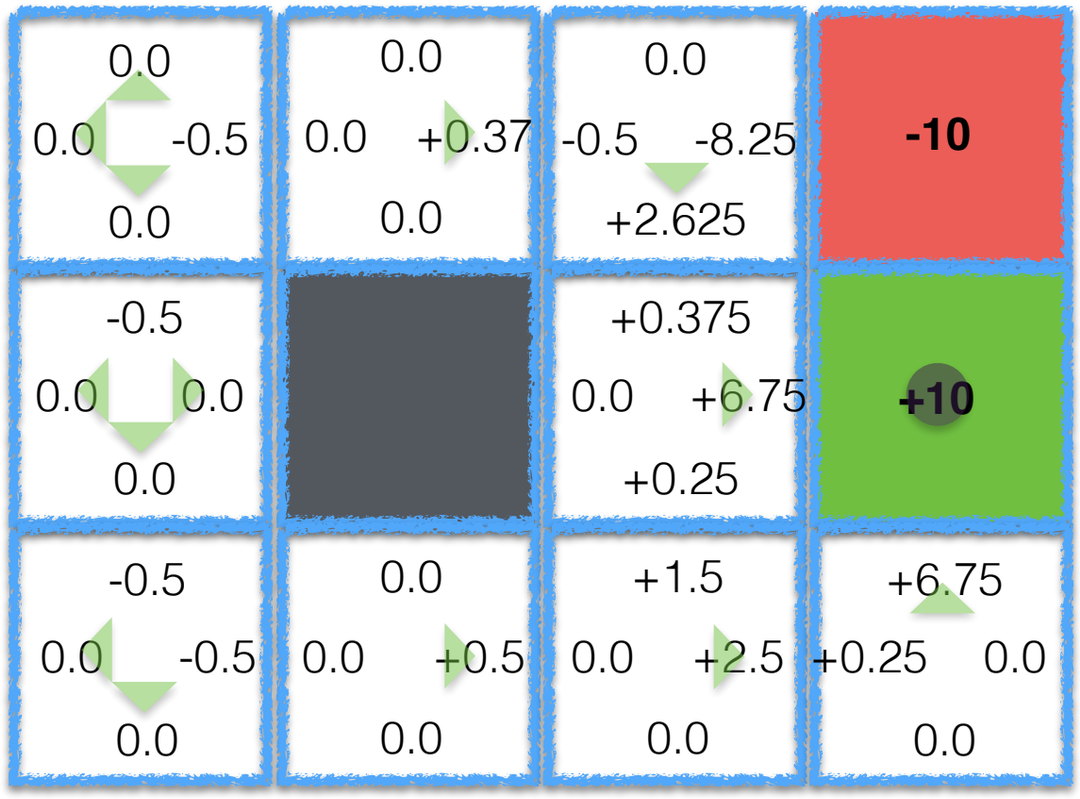
$t = 30$

Grid World Example



$t = 31$

Grid World Example



$terminal = true$

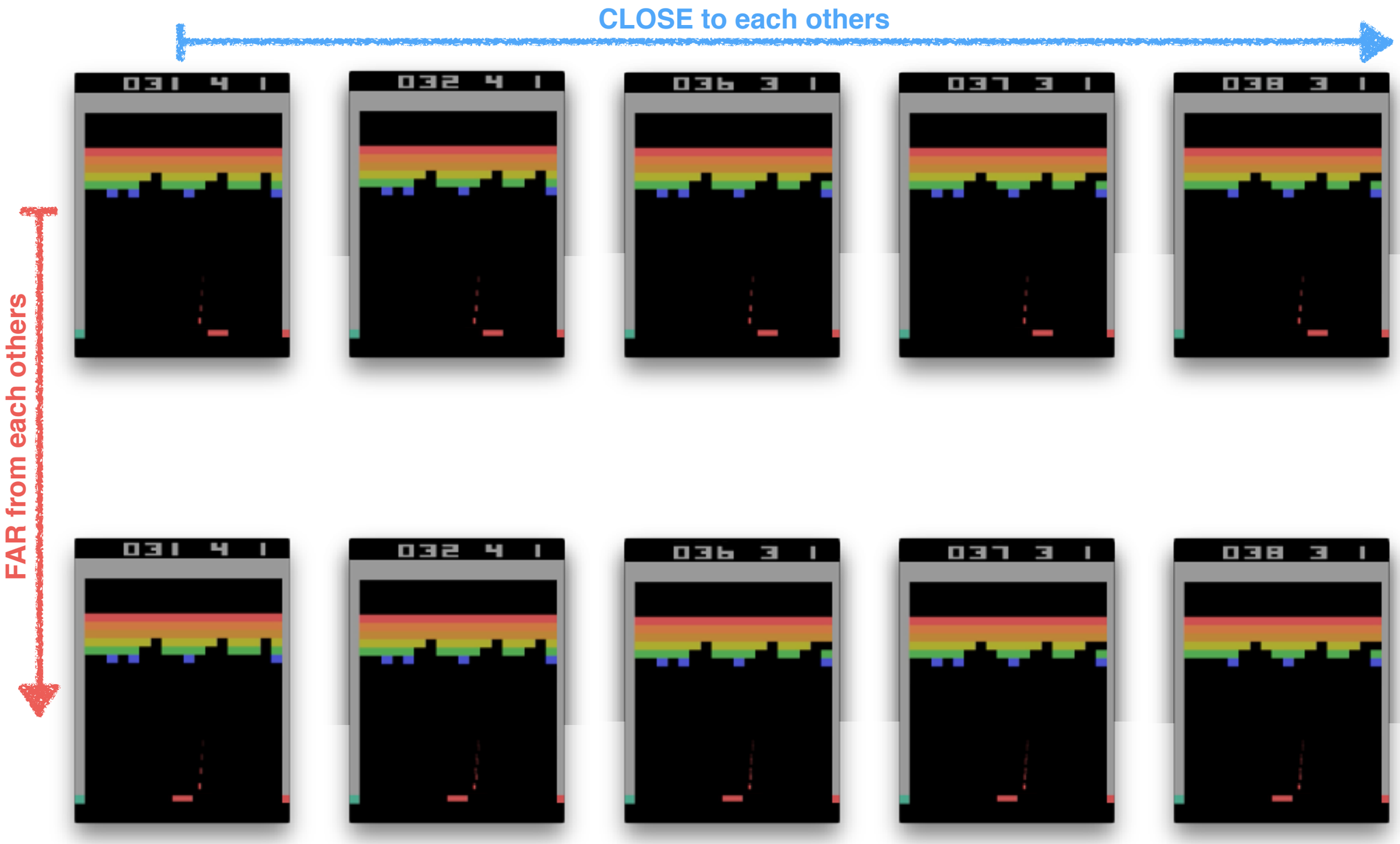
.....

$t = 32$

Q-Learning Properties

- Information propagates backwards from the terminal states.
- Converges to optimal policy even if the agent is not acting optimally
- Updates with the values of what the agent should do, not the values of what it is doing
  - => Off-policy learning
- You have to explore enough visitting most of the states multiple times.

Approximate Q-learning



Q-network

Approximating the action-value function with Neural Networks

$$Q(s, a; \theta) \approx Q^*(s, a)$$

A Q-network can be trained by adjusting the parameter θ_i to minimize

$$L_i(\theta_i) = \mathbb{E}_{s,a,r}[(\mathbb{E}_{s'}[y|s, a] - Q(s, a; \theta_i))^2]$$

where

$$y = \begin{cases} r & \text{if } terminal = true \\ r + \gamma \max_{a' \in \mathcal{A}} Q(s', a'; \theta) & \text{otherwise} \end{cases}$$

https://storage.googleapis.com/deepmind-media/dqn/DQNNaturePaper.pdf

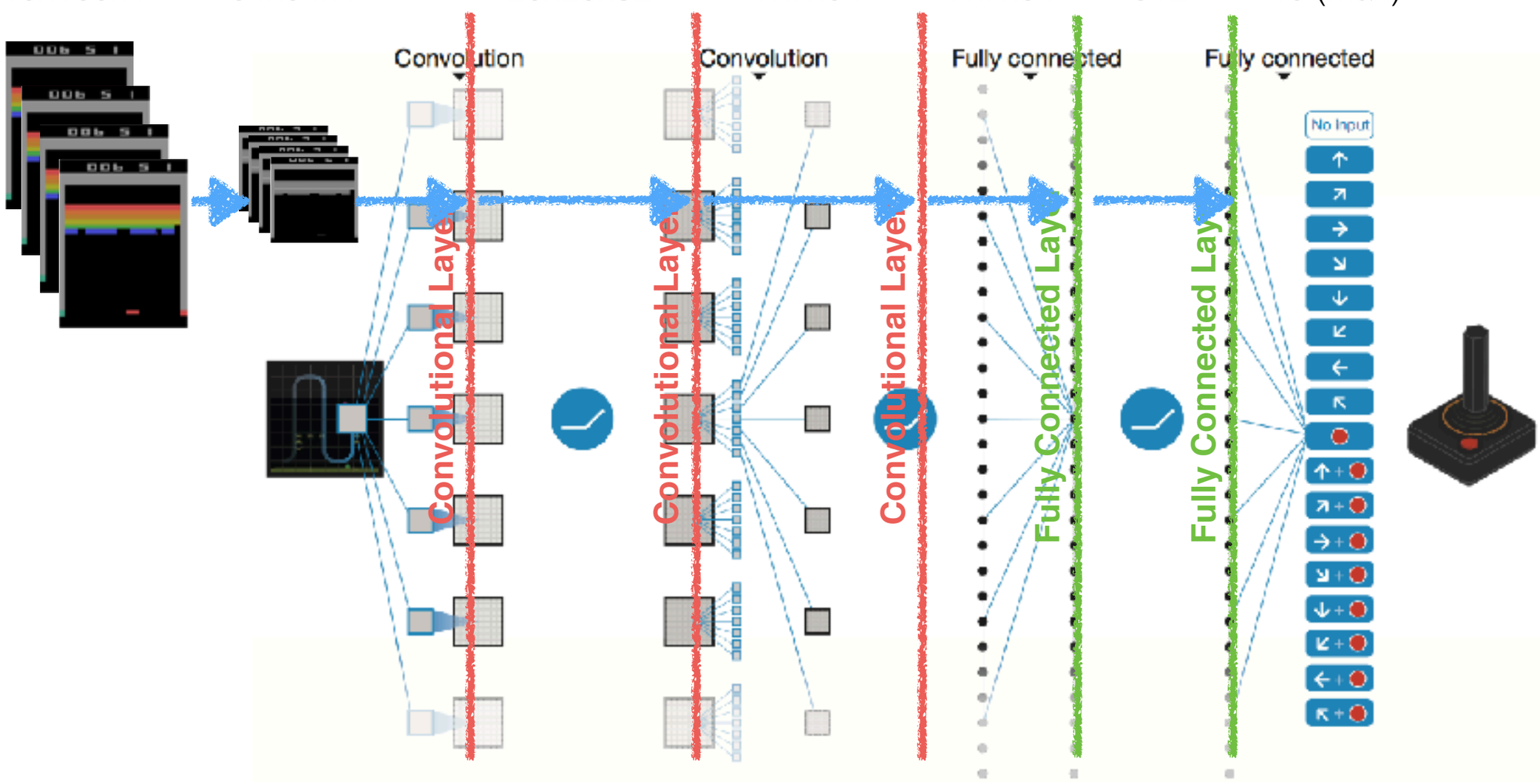


Figure 1 | Schematic illustration of the convolutional neural network. The details of the architecture are explained in the Methods. The input to the neural network consists of an $84 \times 84 \times 4$ image produced by the preprocessing map ϕ, followed by three convolutional layers (note: snaking blue line symbolizes sliding of each filter across input image) and two fully connected layers with a single output for each valid action. Each hidden layer is followed by a rectifier nonlinearity (that is, $\max(0,x)$).

(2015)

https://storage.googleapis.com/deepmind-media/dqn/DQNNaturePaper.pdf

Replay Memory

Replay memory stores the agent's experience at each time-step

$$e_t = (s_t, a_t, r_t, s_{t+1})$$

in a data set

$$D = \{e_1, ..., e_t\}$$

Q-learning update is performed on minibatch drawn at random from the replay memory.

$$(s, a, r, s') \sim U(D)$$

Target Network

Identical structure with the online network

$\hat{Q}(s, a; \theta^-)$ initialized with $\theta^- = \theta$

Target network generates the target y_j in the Q-learning update

$$y_j = \begin{cases} r_j & \text{if } terminal_{j+1} = true \\ r_j + \gamma \max_{a' \in \mathcal{A}} Q(\phi_{j+1}, a'; \theta_i^-) & \text{otherwise} \end{cases}$$

Every C steps reset $\hat{Q} = Q$ by setting $\theta^- \leftarrow \theta$

ϵ-greedy Policy

Exploration-Exploitation Tradeoff

During training select $a \sim U(\mathcal{A})$ With probability ϵ

otherwise $a = \arg\max_{a' \in \mathcal{A}} Q(s, a'; \theta)$

Annealing ϵ linearly from 1.0 to 0.1 over the first million frames and fixed at 0.1 thereafter.

Algorithm 1: deep Q-learning with experience replay.

Initialize replay memory D to capacity N

Initialize action-value function Q with random weights θ

Initialize target action-value function $\hat{Q}$ with weights $\theta^{-} = \theta$

For episode = 1, M **do**

Initialize sequence $s_1 = \{x_1\}$ and preprocessed sequence $\phi_1 = \phi(s_1)$

For t = 1,T **do**

With probability ε select a random action a_t

otherwise select $a_t = \text{argmax}_a Q(\phi(s_t), a; \theta)$

Execute action a_t in emulator and observe reward r_t and image x_{t+1}

Set $s_{t+1} = s_t, a_t, x_{t+1}$ and preprocess $\phi_{t+1} = \phi(s_{t+1})$

Store transition $(\phi_t, a_t, r_t, \phi_{t+1})$ in D

Sample random minibatch of transitions $(\phi_j, a_j, r_j, \phi_{j+1})$ from D

Set $y_j = \begin{cases} r_j & \text{if episode terminates at step j+1} \\ r_j + \gamma \max_{a'} \hat{Q}(\phi_{j+1}, a'; \theta^{-}) & \text{otherwise} \end{cases}$

Perform a gradient descent step on $\left(y_j - Q(\phi_j, a_j; \theta)\right)^2$ with respect to the network parameters θ

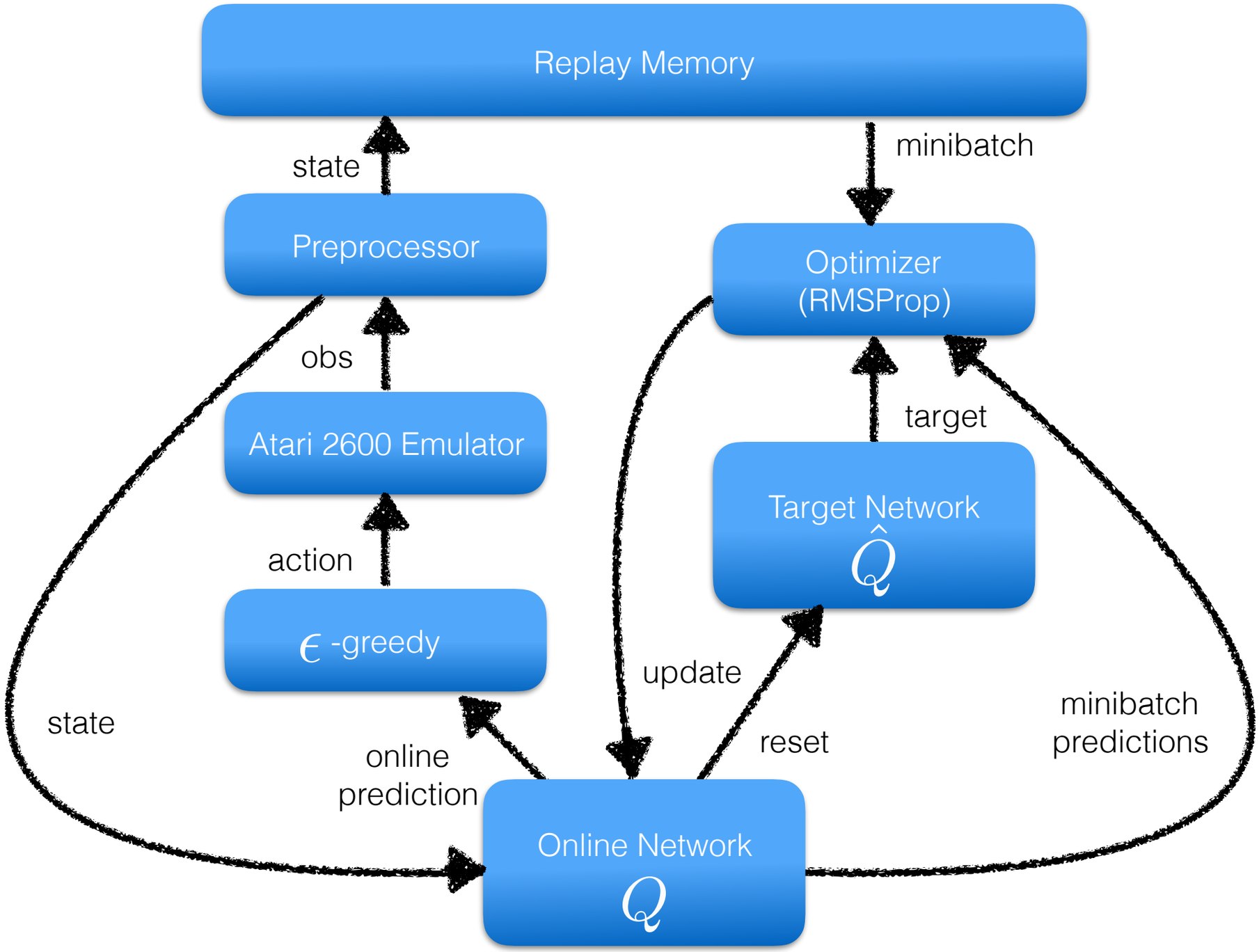
Every C steps reset $\hat{Q} = Q$

End For

End For

https://storage.googleapis.com/deepmind-media/dqn/DQNNaturePaper.pdf

Deep Q-Learning



https://storage.googleapis.com/deepmind-media/dqn/DQNNaturePaper.pdf

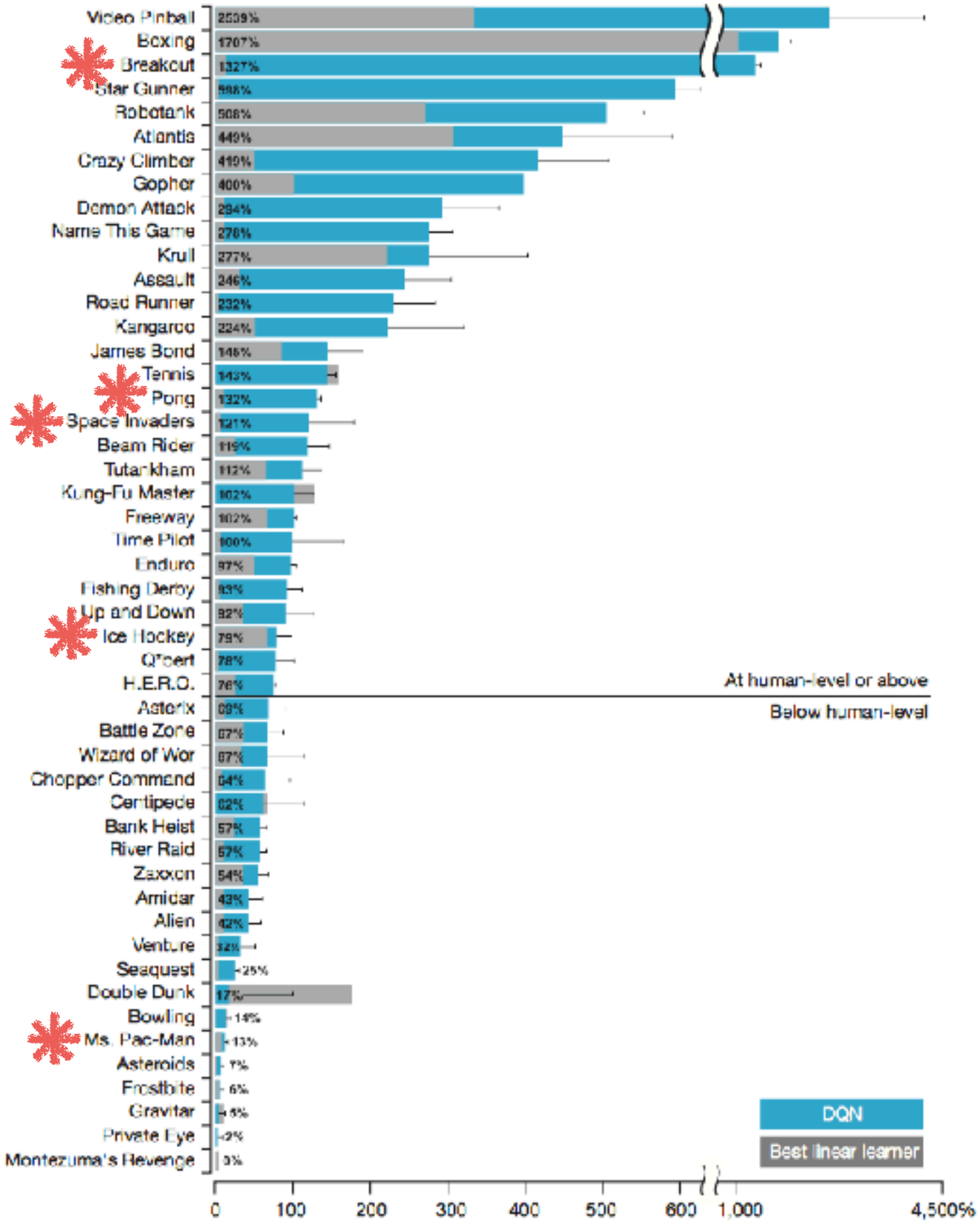
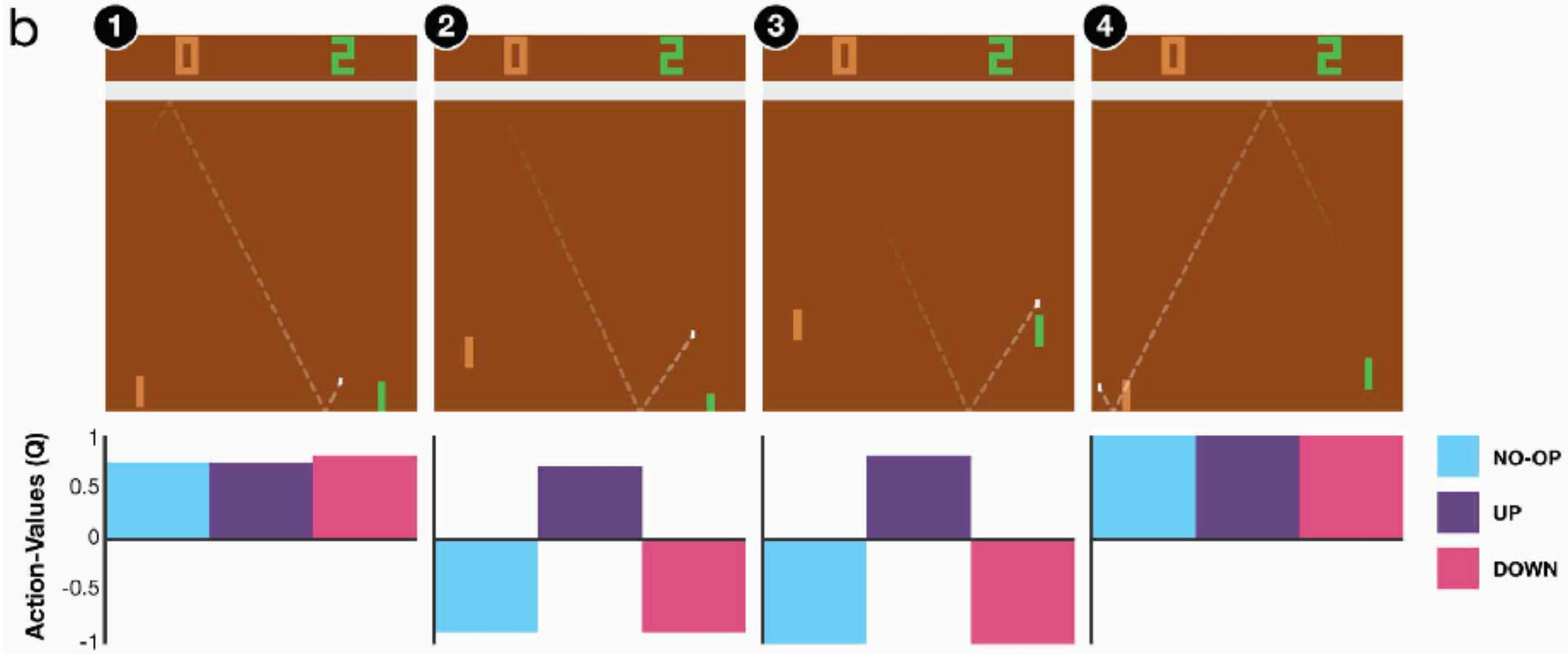


Figure 3 | Comparison of the DQN agent with the best reinforcement learning methods[15] in the literature. The performance of DQN is normalized with respect to a professional human games tester (that is, 100% level) and random play (that is, 0% level). Note that the normalized performance of DQN, expressed as a percentage, is calculated as: 100 × (DQN score − random play score)/(human score − random play score). It can be seen that DQN outperforms competing methods (also see Extended Data Table 2) in almost all the games, and performs at a level that is broadly comparable with or superior to a professional human games tester (that is, operationalized as a level of 75% or above) in the majority of games. Audio output was disabled for both human players and agents. Error bars indicate s.d. across the 30 evaluation episodes, starting with different initial conditions.



b, A visualization of the learned action-value function on the game Pong. At time point 1, the ball is moving towards the paddle controlled by the agent on the right side of the screen and the values of all actions are around 0.7, reflecting the expected value of this state based on previous experience. At time point 2, the agent starts moving the paddle towards the ball and the value of the 'up' action stays high while the value of the 'down' action falls to −0.9. This reflects the fact that pressing 'down' would lead to the agent losing the ball and incurring a reward of −1. At time point 3, the agent hits the ball by pressing 'up' and the expected reward keeps increasing until time point 4, when the ball reaches the left edge of the screen and the value of all actions reflects that the agent is about to receive a reward of 1. Note, the dashed line shows the past trajectory of the ball purely for illustrative purposes (that is, not shown during the game). With permission from Atari Interactive, Inc.

https://storage.googleapis.com/deepmind-media/dqn/DQNNaturePaper.pdf

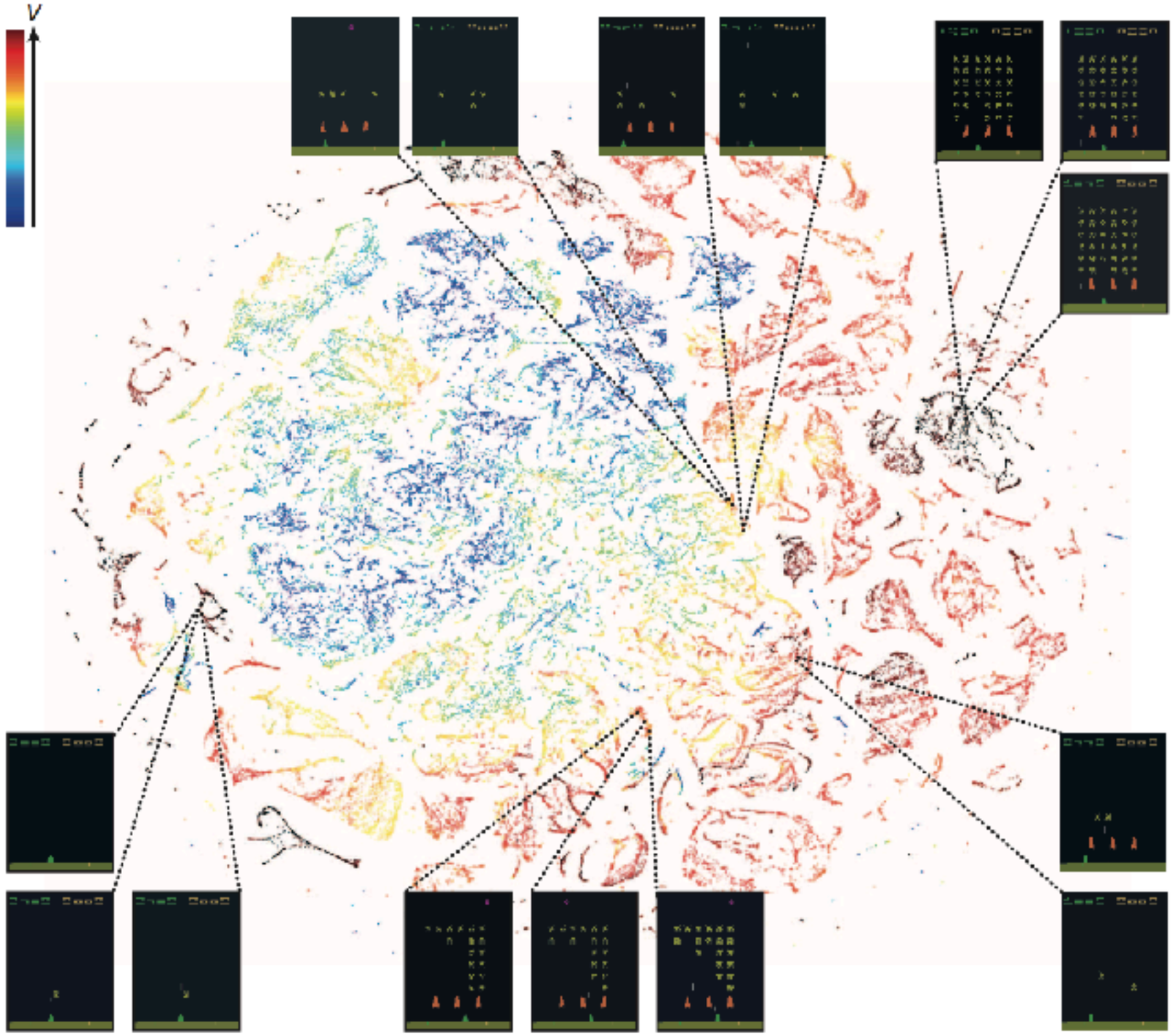


Figure 4 | Two-dimensional t-SNE embedding of the representations in the last hidden layer assigned by DQN to game states experienced while playing Space Invaders. The plot was generated by letting the DQN agent play for 2 h of real game time and running the t-SNE algorithm[25] on the last hidden layer representations assigned by DQN to each experienced game state. The points are coloured according to the state values (V, maximum expected reward of a state) predicted by DQN for the corresponding game states (ranging from dark red (highest V) to dark blue (lowest V)). The screenshots corresponding to a selected number of points are shown. The DQN agent predicts high state values for both full (top right screenshots) and nearly complete screens (bottom left screenshots) because it has learned that completing a screen leads to a new screen full of enemy ships. Partially completed screens (bottom screenshots) are assigned lower state values because less immediate reward is available. The screens shown on the bottom right and top left and middle are less perceptually similar than the other examples but are still mapped to nearby representations and similar values because the orange bunkers do not carry great significance near the end of a level. With permission from Square Enix Limited.

Effects of Components

The whole is greater than the sum of its parts

Extended Data Table 3 | The effects of replay and separating the target Q-network

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

DQN agents were trained for 10 million frames using standard hyperparameters for all possible combinations of turning replay on or off, using or not using a separate target Q-network, and three different learning rates. Each agent was evaluated every 250,000 training frames for 135,000 validation frames and the highest average episode score is reported. Note that these evaluation episodes were not truncated at 5 min leading to higher scores on Enduro than the ones reported in Extended Data Table 2. Note also that the number of training frames was shorter (10 million frames) as compared to the main results presented in Extended Data Table 2 (50 million frames).

https://storage.googleapis.com/deepmind-media/dqn/DQNNaturePaper.pdf

Code & Demo

https://github.com/kerawits/dqn

Deck

https://goo.gl/WP1tkC

Recent Developments

- Double Q-Learning, 2015

  https://arxiv.org/abs/1509.06461

- Dueling Q-Network, 2015

  https://arxiv.org/abs/1511.06581

- Prioritized Experienced Replay, 2015

  https://arxiv.org/abs/1511.05952

- Neural Episodic Control, 2017

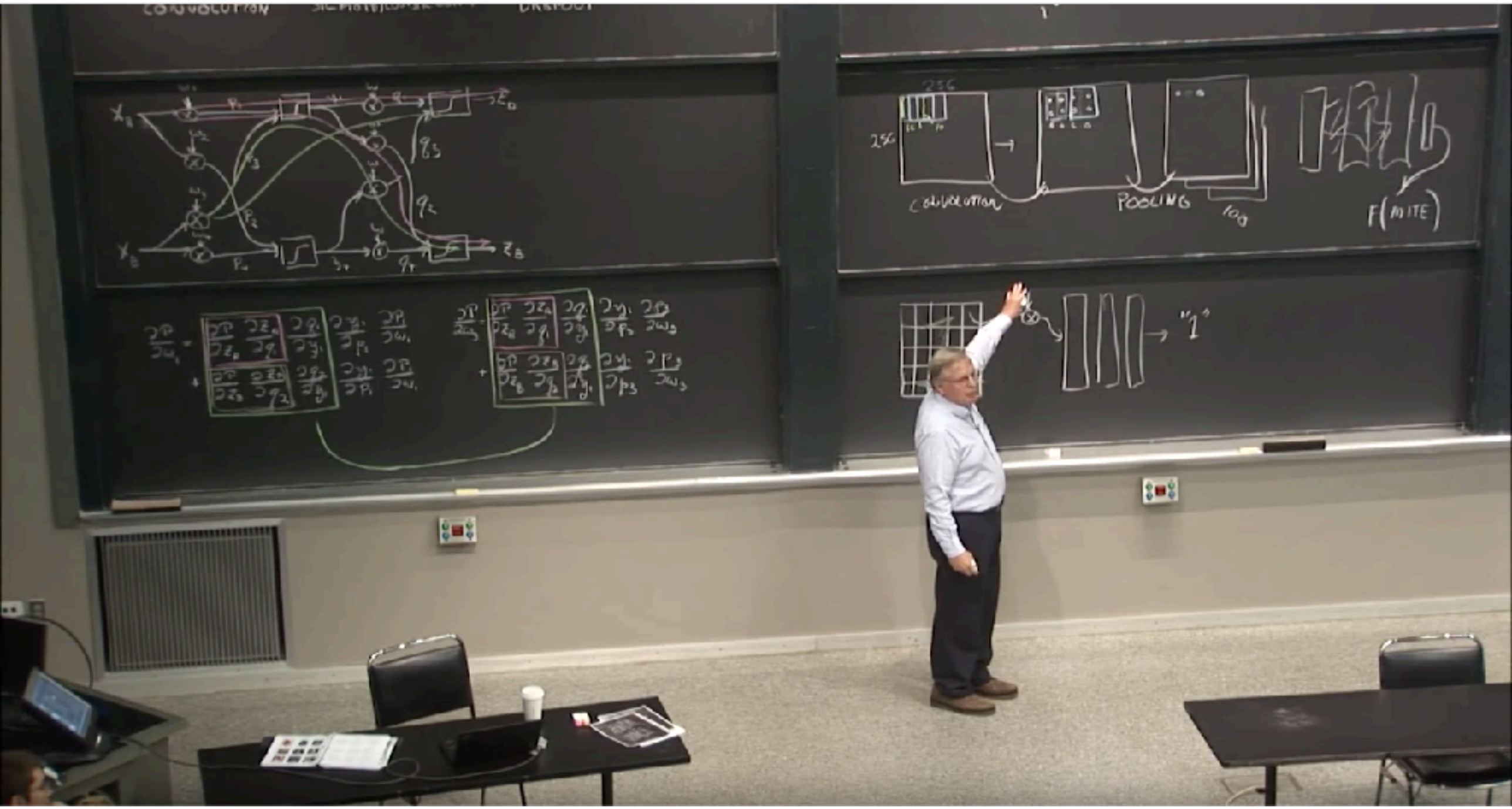
  https://arxiv.org/abs/1703.01988

- Evolution Strategies as a Scalable Alternative to Reinforcement Learning, 2017

  https://arxiv.org/abs/1703.03864

Lecturn 12a & Lecture12b

Prof. Patrick H. Winston, MIT



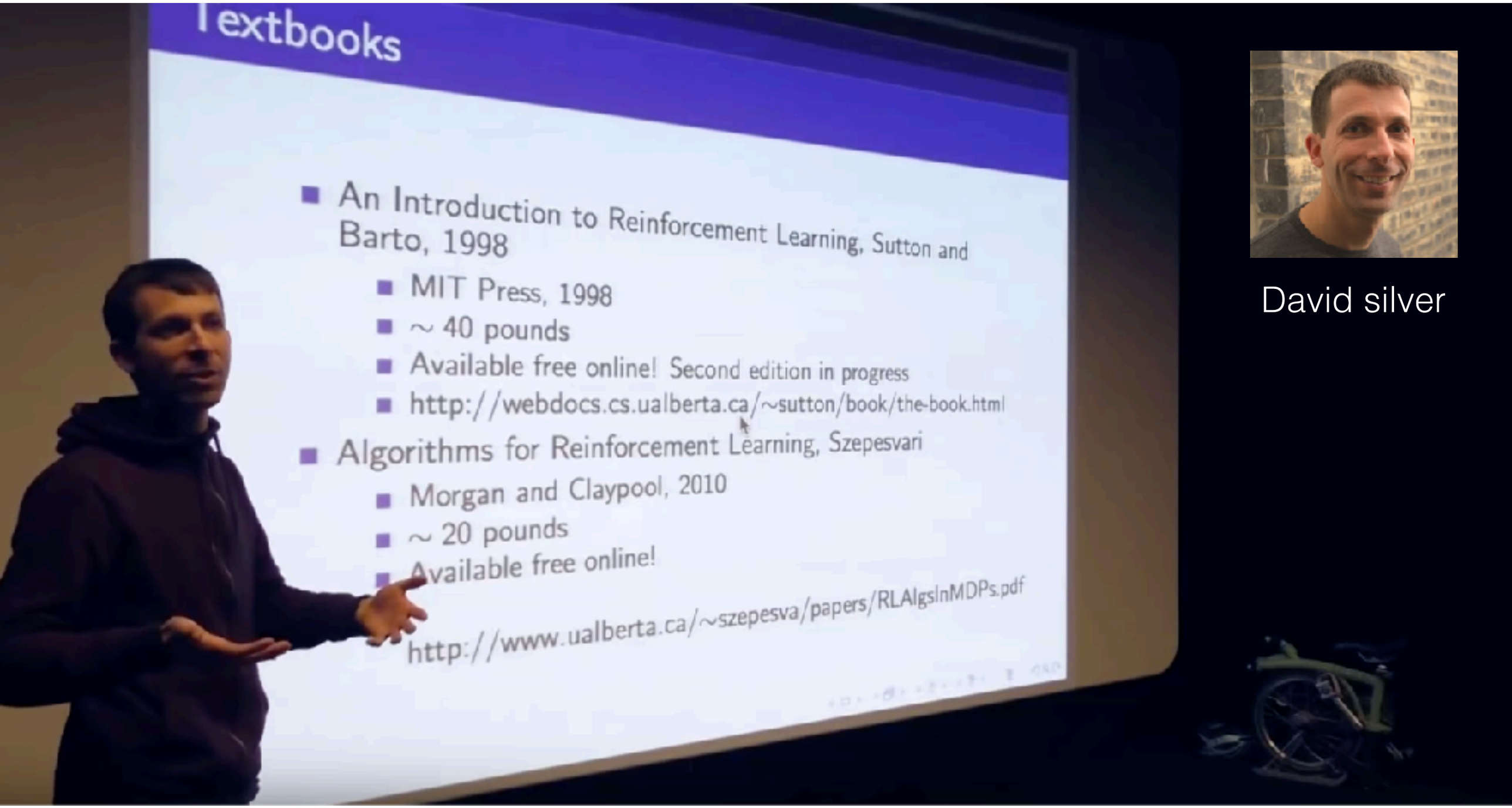
https://www.youtube.com/watch?v=uXt8qF2Zzfo https://www.youtube.com/watch?v=VrMHA3yX_QI

http://artificialbrain.xyz/artificial-intelligence-complete-lectures-01-23/

Peng Brothers Co., Ltd.

Reinforcememt Learning

David Silver, DeepMind

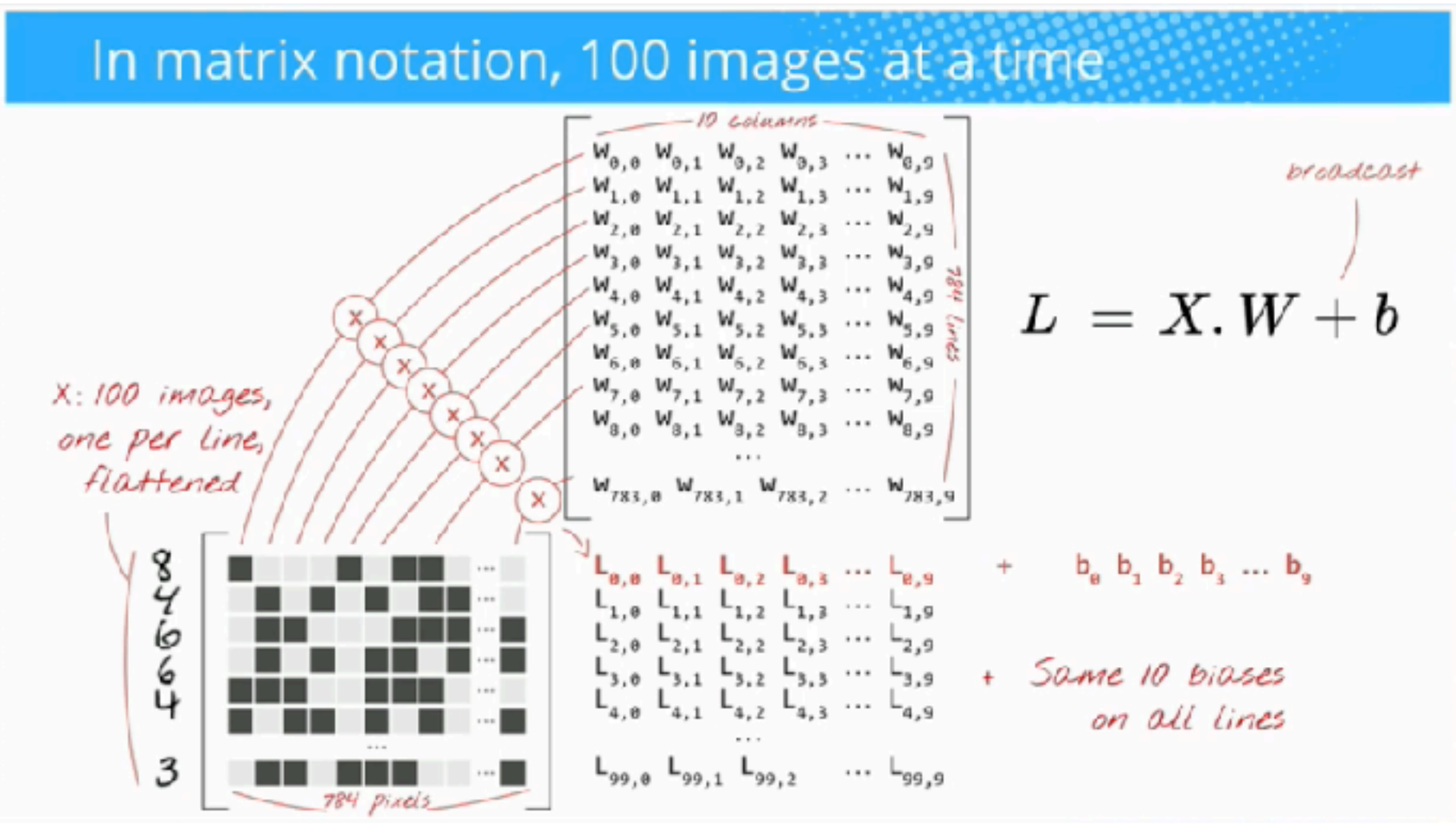


David silver

https://www.youtube.com/playlist?list=PLLHj4l_rxKAV_F-rJGl_NQmpdcFwFi-Z7

Tensorflow & Deep Learning without a PhD

Martin Görner



https://www.youtube.com/watch?v=vq2nnJ4g6N0

Peng Brothers Co., Ltd.

Deep Learning

Yann Lecun, Yoshua Bengio & Geoffrey Hinton

REVIEW

doi:10.1038/nature14539

Deep learning

Yann LeCun[1,2], Yoshua Bengio[3] & Geoffrey Hinton[4,5]

Deep learning allows computational models that are composed of multiple processing layers to learn representations of data with multiple levels of abstraction. These methods have dramatically improved the state-of-the-art in speech recognition, visual object recognition, object detection and many other domains such as drug discovery and genomics. Deep learning discovers intricate structure in large data sets by using the backpropagation algorithm to indicate how a machine should change its internal parameters that are used to compute the representation in each layer from the representation in the previous layer. Deep convolutional nets have brought about breakthroughs in processing images, video, speech and audio, whereas recurrent nets have shone light on sequential data such as text and speech.

Machine-learning technology powers many aspects of modern society: from web searches to content filtering on social networks to recommendations on e-commerce websites, and it is increasingly present in consumer products such as cameras and smartphones. Machine-learning systems are used to identify objects in images, transcribe speech into text, match news items, posts or products with users' interests, and select relevant results of search. Increasingly, these applications make use of a class of techniques called *deep learning*.

Conventional machine-learning techniques were limited in their ability to process natural data in their raw form. For decades, constructing a pattern-recognition or machine-learning system required careful engineering and considerable domain expertise to design a feature extractor that transformed the raw data (such as the pixel values of an image) into a suitable internal representation or feature vector from which the learning subsystem, often a classifier, could detect or classify patterns in the input.

Representation learning is a set of methods that allows a machine to be fed with raw data and to automatically discover the representations intricate structures in high-dimensional data and is therefore applicable to many domains of science, business and government. In addition to beating records in image recognition[1–4] and speech recognition[5–7], it has beaten other machine-learning techniques at predicting the activity of potential drug molecules[8], analysing particle accelerator data[9,10], reconstructing brain circuits[11], and predicting the effects of mutations in non-coding DNA on gene expression and disease[12,13]. Perhaps more surprisingly, deep learning has produced extremely promising results for various tasks in natural language understanding[14], particularly topic classification, sentiment analysis, question answering[15] and language translation[16,17].

We think that deep learning will have many more successes in the near future because it requires very little engineering by hand, so it can easily take advantage of increases in the amount of available computation and data. New learning algorithms and architectures that are currently being developed for deep neural networks will only accelerate this progress.

Supervised learning

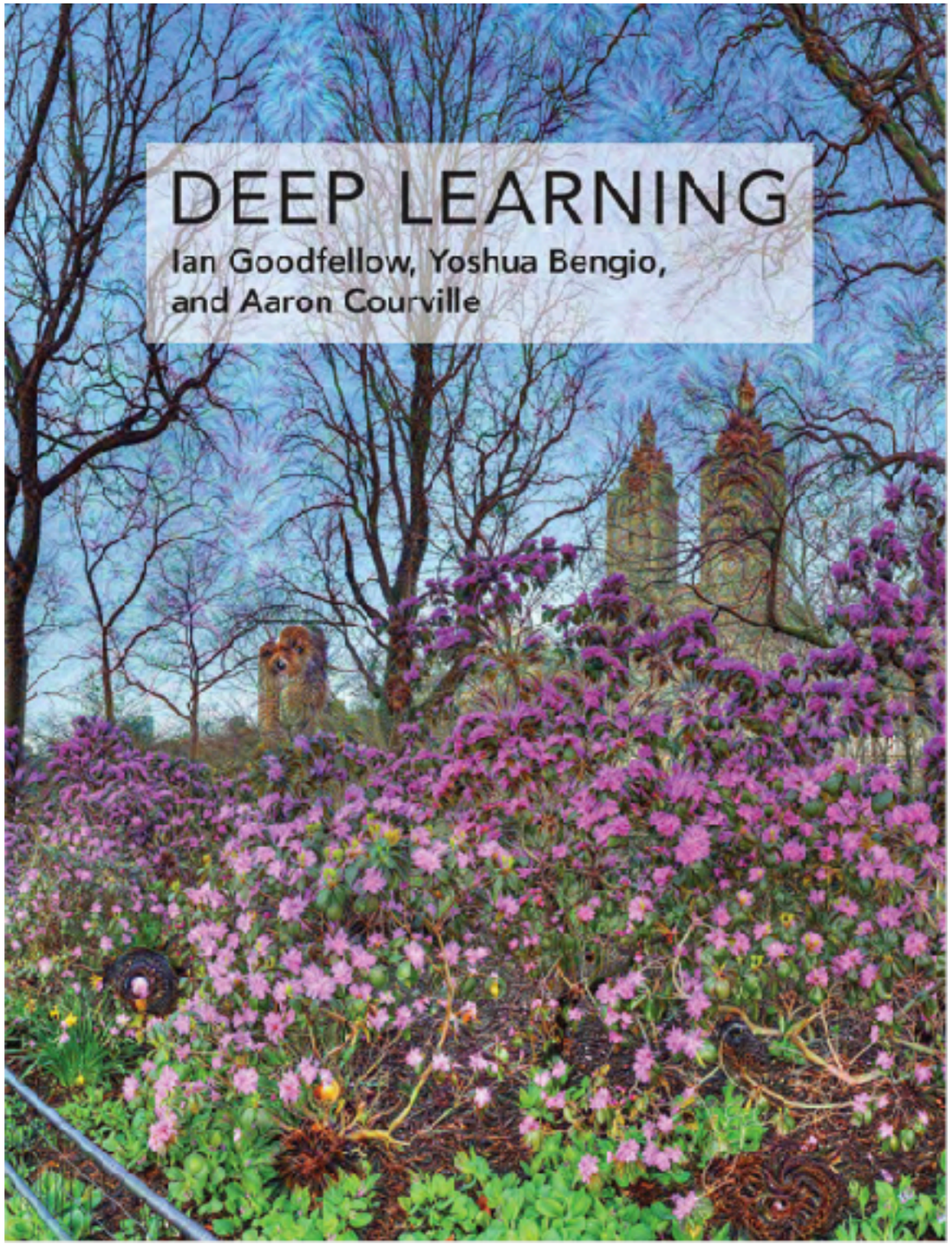
http://www.nature.com/articles/nature14539.epdf?referrer_access_token=K4awZz78b5Yn2_AoPV_4Y9RgN0jAjWel9jnR3ZoTv0PU8PImtLRceRBJ32CtadUBVOwHuxbf2QgphMCsA6eTOw64kccq9ihWSKdxZpGPn2fn3B_8bxaYh0svGFqgRLgaiyW6CBFAb3Fpm6GbL8a_TtQQDWKuhD1XKh_wxLReRpGbR_NdccoaiKP5xvzbV-x7b_7Y64ZSpqG6kmfwS6Q1rw%3D%3D&tracking_referrer=www.nature.com

Deep Learning

Ian Goodfellow, Yoshua Bengio & Aaron Courville

Ian Goodfellow

Aaron Courville



http://www.deeplearningbook.org

Yoshua Bengio