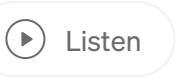
✦ Member-only story

Unlocking the Secrets of Actor-Critic Reinforcement Learning: A Beginner's Guide

Understanding Actor-Critic Mechanisms, Different Flavors of Actor-Critic Algorithms, and a Simple Implementation in PyTorch

Renu Khandelwal · Follow
6 min read · Feb 21



Concepts you should Know:

Reinforcement Learning: Temporal Difference Learning

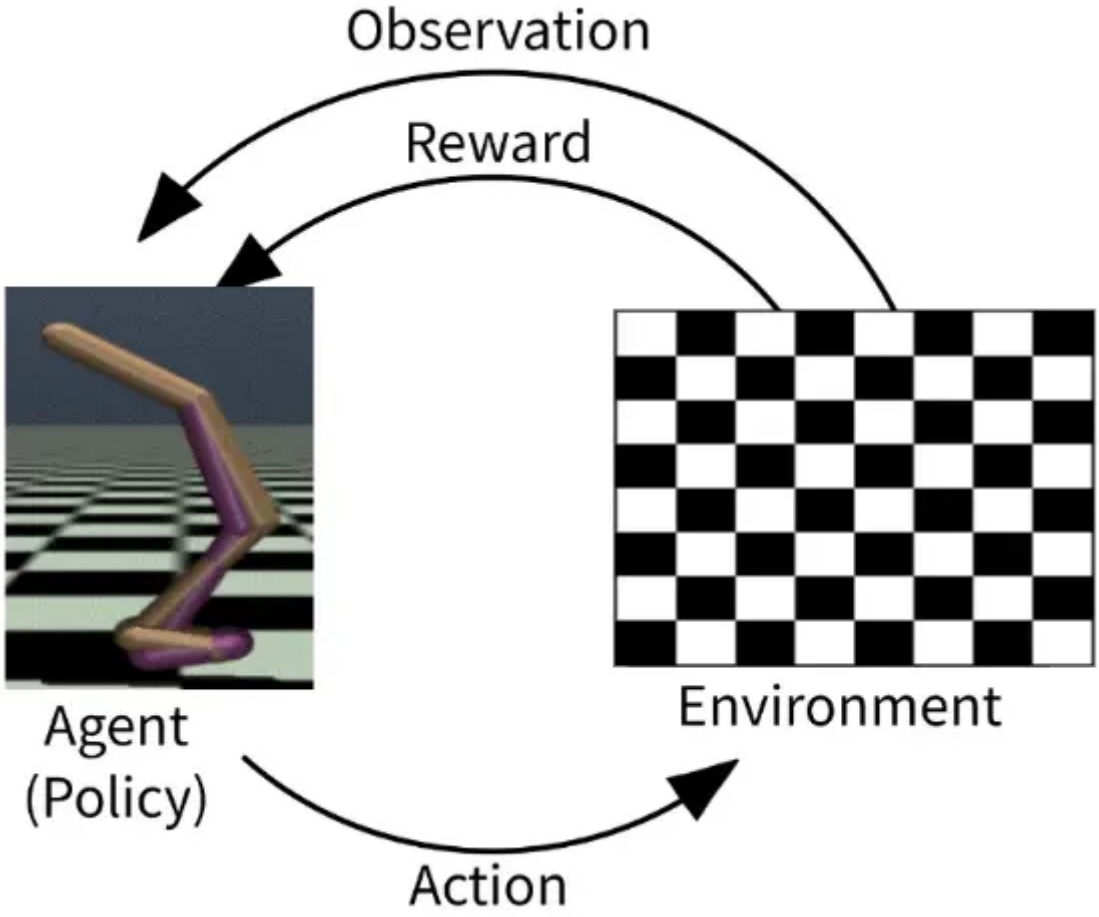
Reinforcement Learning: Q-Learning

Deep Q Learning: A Deep Reinforcement Learning Algorithm

An Intuitive Explanation of Policy Gradient

What is the Actor-Critic algorithm?

Actor-Critic is a Reinforcement Learning algorithm that optimizes the agent's actions based on the environment's feedback.



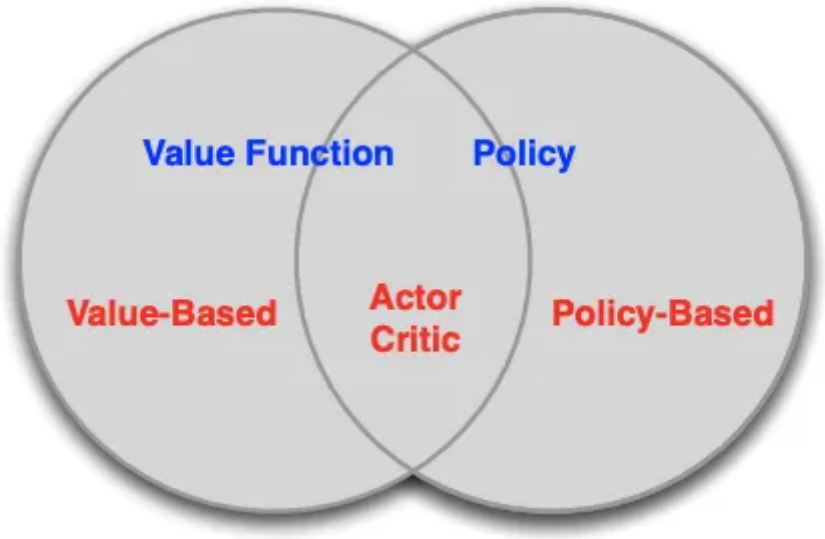
> *The Actor-Critic RL aims to find an optimal policy for the agent in an environment using two components: Actor and Critic.*

Actor: The Actor learns an optimal policy by exploring the environment

Critic: The Critic assesses the value of each action taken by the Actor to determine whether the action will result in a better reward, guiding the Actor for the best course of action to take.

The Actor then uses the feedback from Critic to adjust its policy and make more informed decisions, leading to improved overall performance.

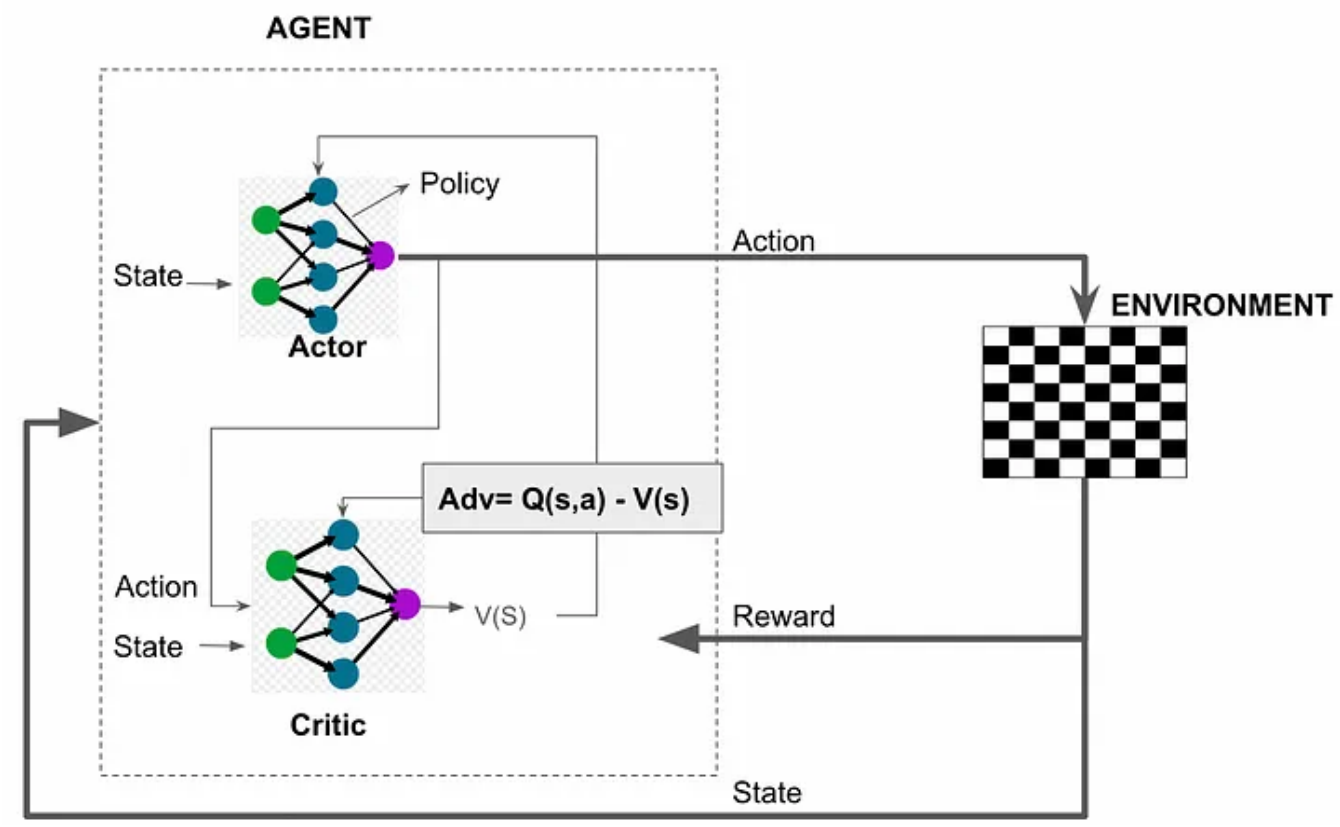
> *The Actor-Critic is a combination of value-based, and policy-based methods where the Actor controls how our agent behaves using the Policy gradient, and the Critic evaluates how good the action taken by the Agent based on value-function.*



The Actor uses policy gradient to control how Agent behaves, and Critic uses the Value-based Q function to evaluate the action taken by the Agent(source: https://www.davidsilver.uk/wp-content/uploads/2020/03/pg.pdf)

> *In value-based methods, the value function is estimated to predict the expected future reward for a given state or action.*
>
> *Policy-based methods directly map states to actions through a policy. The policy is updated using the policy gradient theorem, which updates the policy in the gradient direction to increase the expected reward.*

How does Actor-Critic Algorithm Work?



A2C-Advantage Actor Critic(image by author)

The Actor-Critic algorithm takes inputs from the environment and uses those states to determine the optimal actions.

The Actor component of the algorithm takes the current state as input from the environment. It uses a neural network, which serves as the policy, to output the probabilities of each action for the state.

The Critic network takes the current state and the Actor's outputted actions as inputs and uses this information to estimate the expected future reward, also known as the Q-value. The Q-value represents the expected cumulative reward an agent can expect to receive if it follows a certain policy in a given state.

On the other hand, the value state represents the expected future reward for a given state, regardless of the action taken. It is calculated as the average of all the Q-values for a given state over all possible actions.

> *The difference between the expected reward and the average reward for the action is referred to as the advantage function or temporal difference.*

Adv. = Q(s,a) — V(s)

The advantage function provides valuable information to guide the Actor's policy, allowing it to determine which actions will lead to the best outcomes and adjust its policy accordingly.

> *If the advantage function for a particular state-action pair is positive, taking that action in that state is expected to yield a better outcome than the average action taken in that state.*
>
> *The negative value of the advantage function indicates that the current action is less advantageous than expected, and the agent needs to explore other actions or update the policy to improve the performance.*

As a result, the advantage function is backpropagated to both the Actor and the Critic, allowing both components to continuously update and improve their respective functions. This results in improved overall performance, as the Actor becomes more effective at making decisions that lead to better outcomes. Ultimately, the Actor-Critic algorithm learns an optimal policy that maximizes the expected future rewards.

The Actor-Critic algorithm is like a framework that forms as the base for several other algorithms like A2C, ACER, A3C, TRPO, and PPO.

Different Actor-Critic based RL algorithms

- **A2C- Advantage Actor Critic: The Critic of Advantage Actor-Critic(A2C) methods is trained to predict $V(s)$ so that it can be used to estimate the advantage function $A(s,a)=Q(s,a)-V(s)$ for its bootstrapping.** The Actor is trained using the advantage function as the guidance signal to update its policy.
- **ACER- Actor Critic with Experience Replay**: ACER is sample-efficient actor-critic algorithm that uses experience replay, trust region policy optimization method to improve its performance.
- **A3C- Asynchronous Advantage Actor Critic: A parallel, asynchronous multi-threaded implementation of the actor-critic algorithm. Multiple agents are trained in parallel in their own environment, exploring different parts of the state spaces simultaneously.** Agents calculate policy gradients and periodically send updates to a global network or when a terminal state is reached. The global network then propagates new weights to the agents at each update to guarantee they share a common policy.
- **TRPO- Trust Region Policy Optimization:** Uses the actor-critic algorithm and **Trust region to constrain the policy update.** The policy update is measured using the KL divergence between the old policy and the updated policy, which is used as a measure for the trust region at each iteration.
- **PPO- Proximal Policy Optimization:** PPO is based on an actor-critic algorithm using multiple epochs of stochastic gradient ascent to perform each policy update. **It improves the training stability of the policy by limiting the change to the policy by avoiding too-large policy updates at each training epoch.**

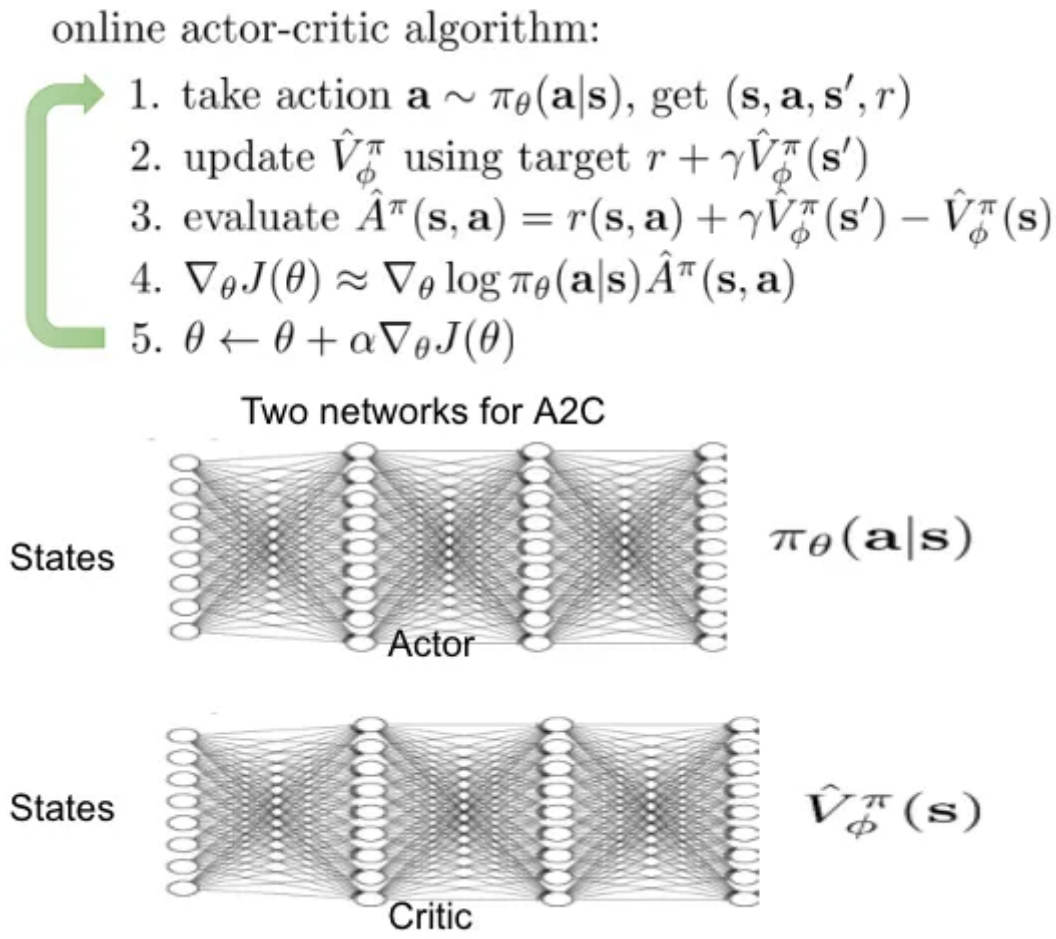
What are the applications for the Actor-Critic algorithm?

The Actor-Critic algorithm is widely utilized in

- **Control systems**, particularly for robots in **manufacturing or service industries,**
- **Gaming** to optimize the game strategy,
- Complex systems such as **power grids, autonomous vehicles, and industrial processes.**

Code Implementation

Here we will use two neural networks: Actor and Critic.



source:http://rail.eecs.berkeley.edu/deeprlcourse-fa17/f17docs/lecture_5_actor_critic_pdf

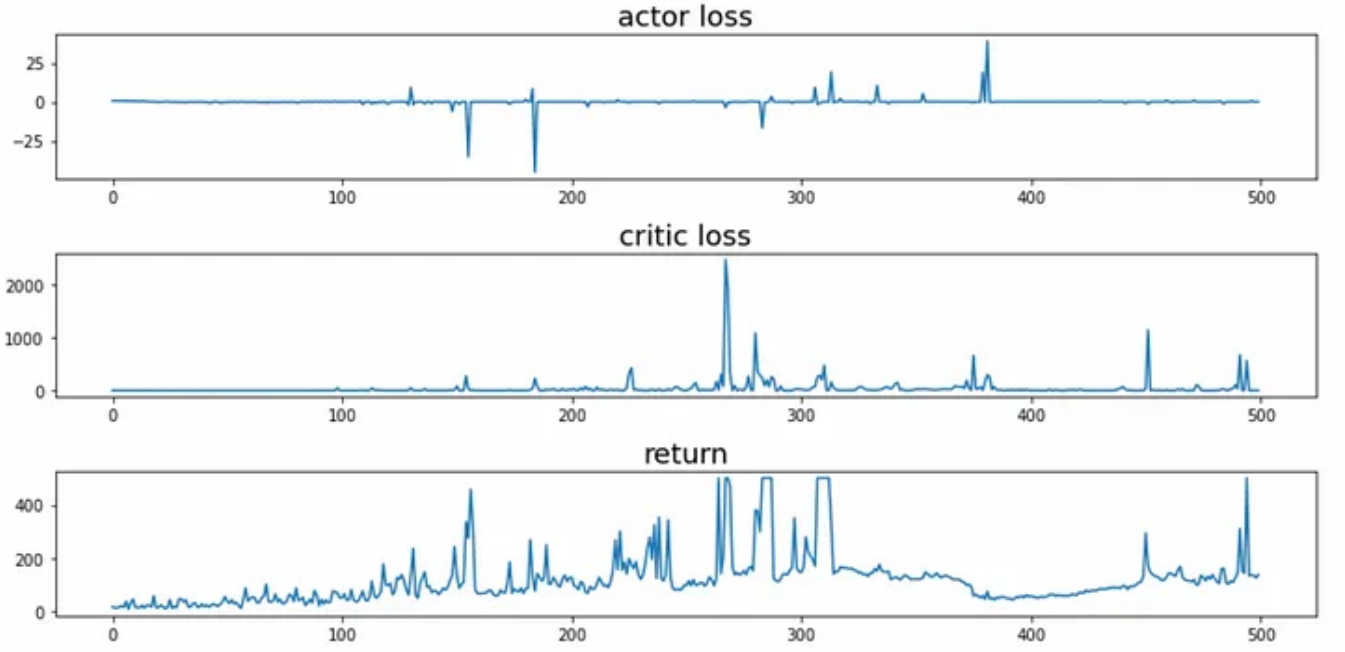
Using the Actor network for each episode step, the agent will take action in the current state, land into the next state, and get the reward from the environment. The Actor's neural network serves as the policy to output the probability of taking each possible action in that state.

The reward and the estimated value of the next state are used to calculate the advantage function, which is the expected return of taking action minus the estimated value of the current state.

$$\hat{A}^\pi(\mathbf{s}, \mathbf{a}) = r(\mathbf{s}, \mathbf{a}) + \gamma \hat{V}_\phi^\pi(\mathbf{s}') - \hat{V}_\phi^\pi(\mathbf{s})$$

Updates the actor network by computing the actor loss, which is the negative of the log probability of the action taken, multiplied by the advantage.

$$\nabla_\theta J(\theta) \approx \nabla_\theta \log \pi_\theta(\mathbf{a}|\mathbf{s}) \hat{A}^\pi(\mathbf{s}, \mathbf{a})$$



Code https://github.com/arshren/Reinforcement_Learning/blob/main/Advantage_Actor_Critic.ipynb

The A2C is fast and efficient and can learn quickly and efficiently from large amounts of data. It Actor explores the environment, and the Critic helps the Actor by providing feedback to exploit the best actions that the actor can take, thus trying to achieve an optimal policy over time.

A2C works well for continuous action spaces but not so well for discrete action spaces. It is sensitive to the hyper-parameters for optimal performance; incorrect hyper-parameters can make the train unstable.

Conclusion:

The Actor-Critic algorithm uses two components: the Actor to learn an optimal policy through exploration, and the Critic, to evaluate the actor's action to determine the best actions for a state. The Critic does this by giving the Actor feedback that would result in improved performance. The Actor-Critic algorithm works well for continuous action spaces, and hyper-parameters for training. The Actor-Critic model needs to be experimented well to avoid instability.

References:

REINFORCEMENT LEARNING THROUGH ASYNCHRONOUS ADVANTAGE ACTOR-CRITIC ON A GPU

Asynchronous Methods for Deep Reinforcement Learning

https://www.davidsilver.uk/wp-content/uploads/2020/03/pg.pdf

http://rail.eecs.berkeley.edu/deeprlcourse-fa17/f17docs/lecture_5_actor_critic_pdf

https://ai.stackexchange.com/questions/7390/what-is-the-difference-between-actor-critic-and-advantage-actor-critic

Artificial Intelligence | Robotics | Reinforcement Learning | Actor Critic | Policy Gradient

Follow

Written by Renu Khandelwal

6.3K Followers

A Technology Enthusiast who constantly seeks out new challenges by exploring cutting-edge technologies to make the world a better place!

More from Renu Khandelwal

```
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   PassengerId  418 non-null    int64
 1   Survived     418 non-null    int64
 2   Pclass       418 non-null    int64
 3   Name         418 non-null    object
 4   Sex          418 non-null    object
 5   Age          332 non-null    float64
 6   SibSp        418 non-null    int64
 7   Parch        418 non-null    int64
 8   Ticket       418 non-null    object
```

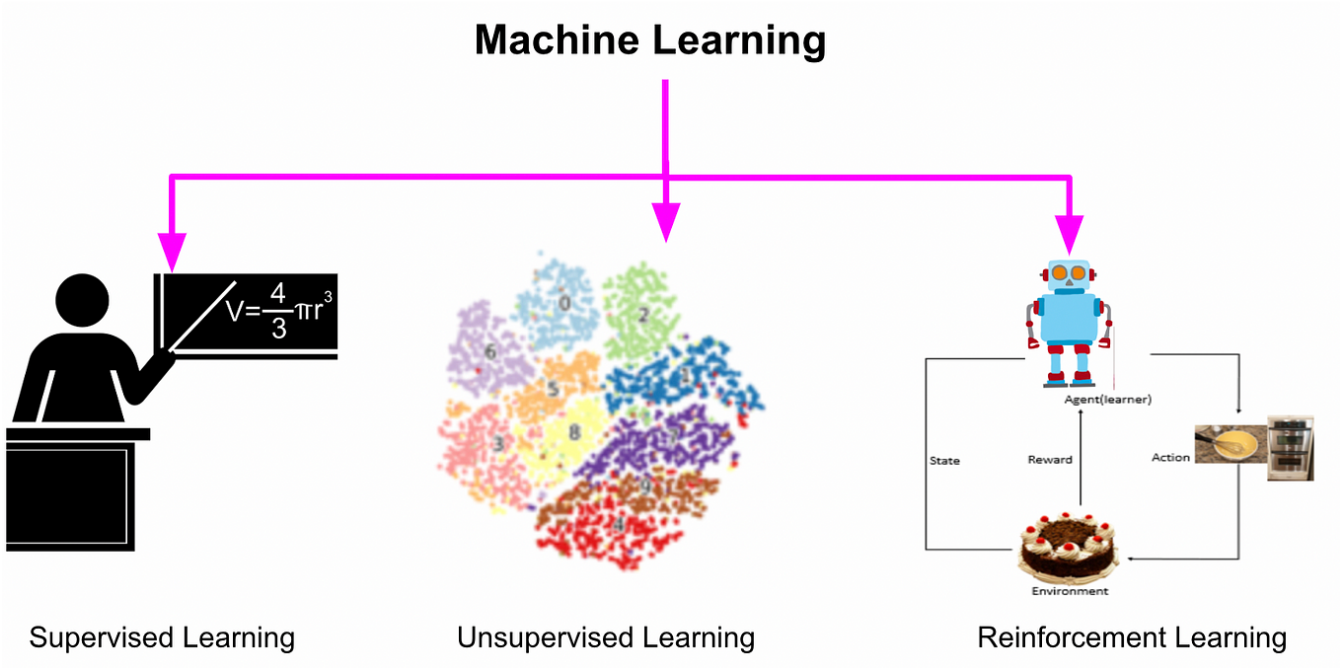
Renu Khandelwal in AI Mind

Mastering Pandas DataFrames for Machine Learning

A Comprehensive Guide for Exploratory Data Analysis(EDA) with Pandas

✨ · 12 min read · Aug 15

144



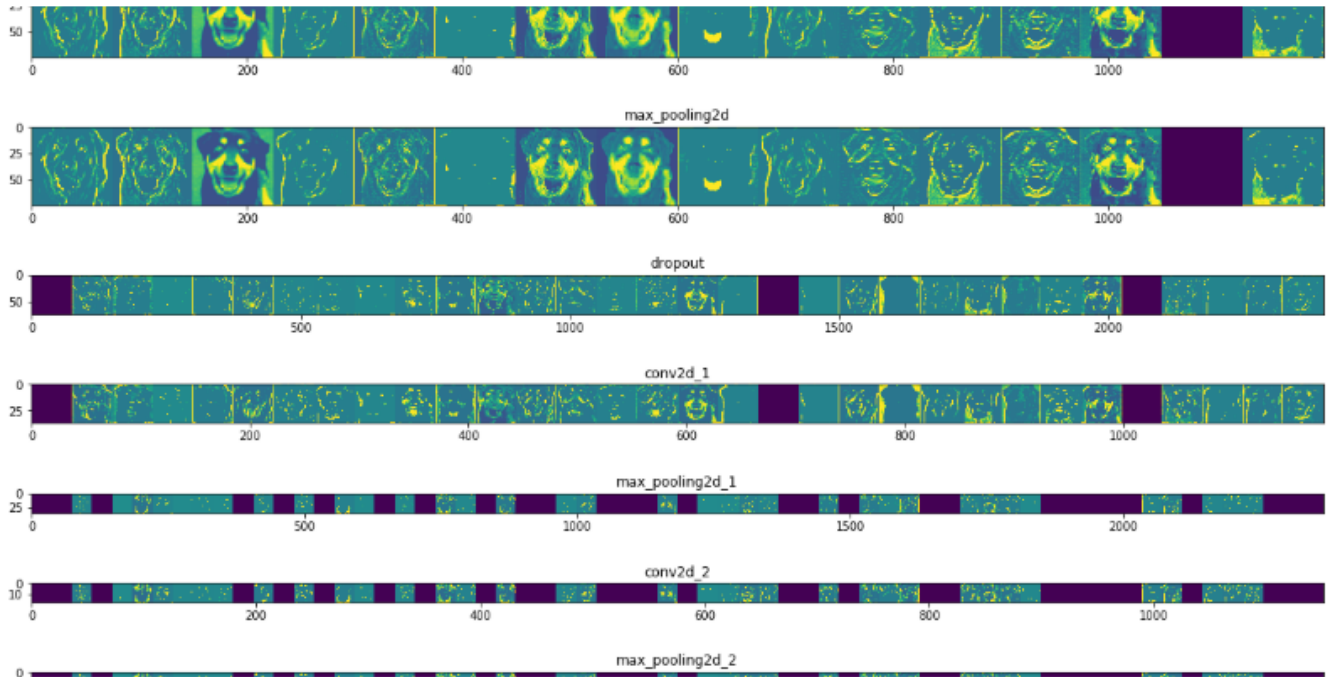
Renu Khandelwal

Supervised, Unsupervised, and Reinforcement Learning

An Intuitive explanation of Supervised, Unsupervised, and Reinforcement learning along with the differences

✨ · 5 min read · Jul 20, 2022

187



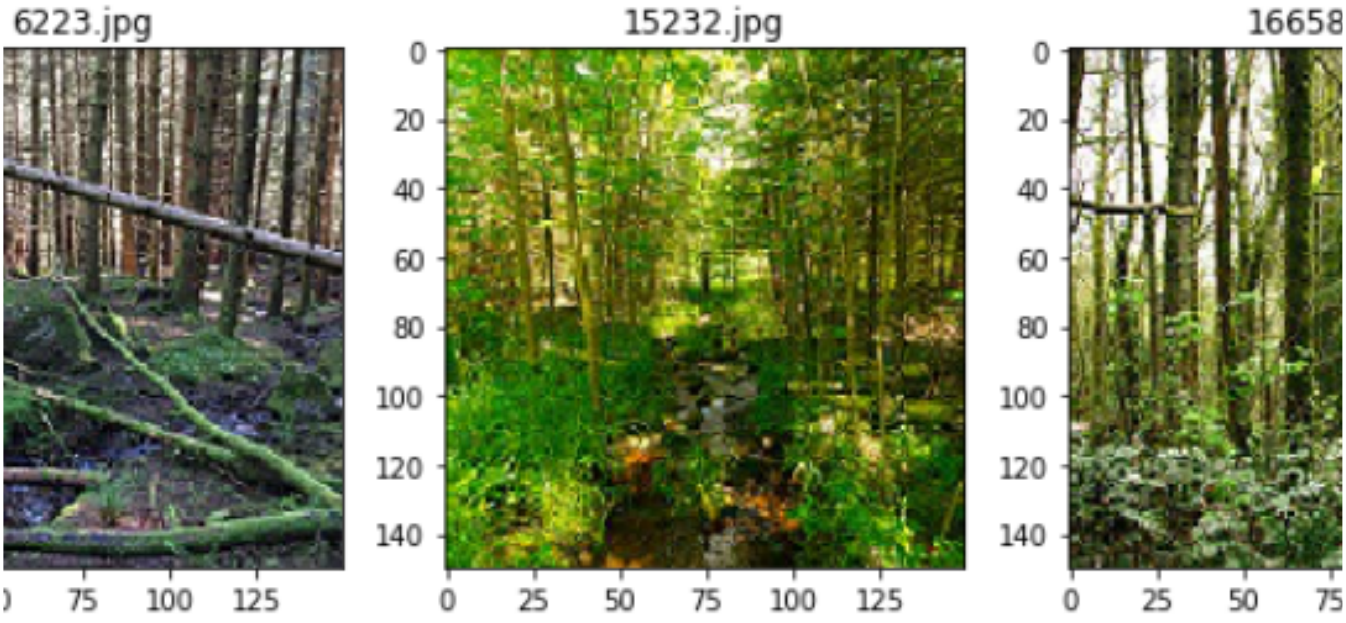
Renu Khandelwal in Towards Data Science

Convolutional Neural Network: Feature Map and Filter Visualization

Learn how Convolutional Neural Networks understand images.

· 8 min read · May 18, 2020

480 6



Renu Khandelwal in Towards Data Science

Loading Custom Image Dataset for Deep Learning Models: Part 1

A simple guide to different techniques for loading a custom image dataset into deep learning models.

· 4 min read · Aug 20, 2020

See all from Renu Khandelwal

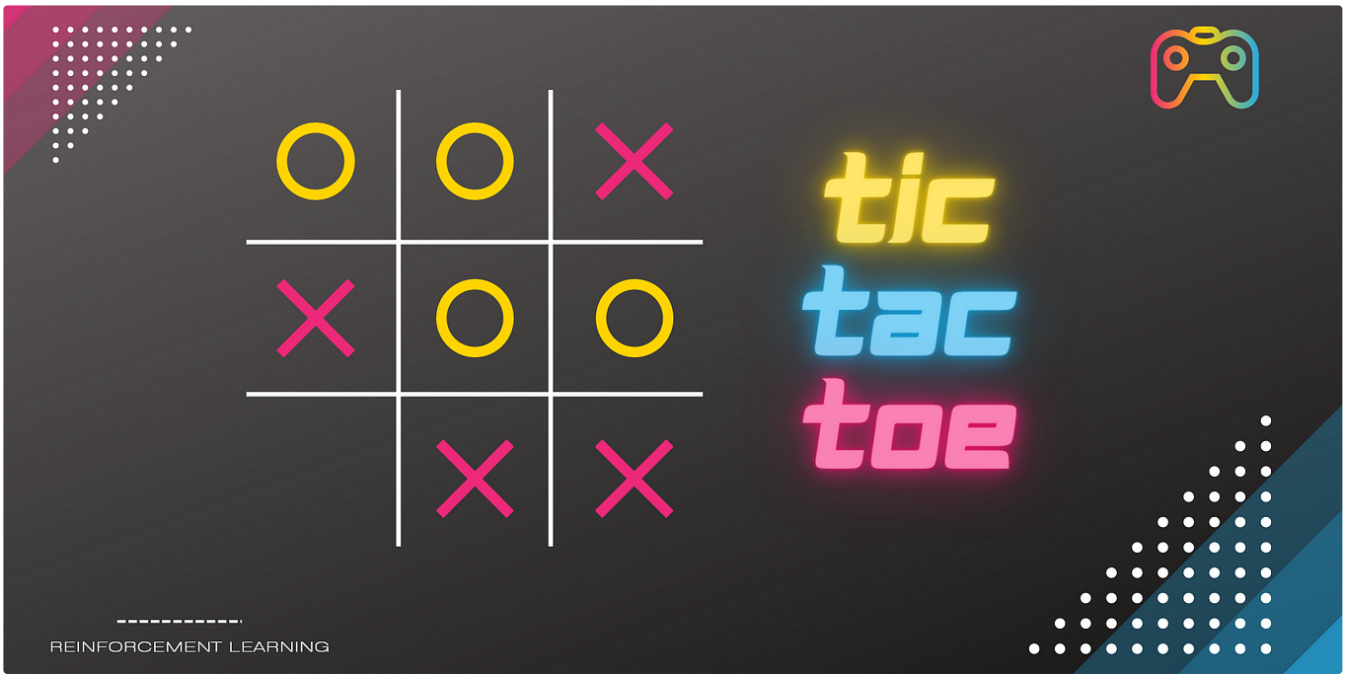
Recommended from Medium

Siwei Causevic in Towards Data Science

Generalized Advantage Estimation in Reinforcement Learning

Bias and Variance tradeoff in Policy Gradient

· 6 min read · Mar 27



Waleed Mousa in Artificial Intelligence in Plain English

Building a Tic-Tac-Toe Game with Reinforcement Learning in Python: A Step-by-Step Tutorial

Welcome to this step-by-step tutorial on how to build a Tic-Tac-Toe game using reinforcement learning in Python. In this tutorial, we will...

9 min read · Mar 13

39 1

Lists

AI Regulation

6 stories · 90 saves

ChatGPT

21 stories · 124 saves

ChatGPT prompts

24 stories · 298 saves

Generative AI Recommended Reading

52 stories · 184 saves

---

cal Programmatic Reinforcement
ia Learning to Compose Program

---

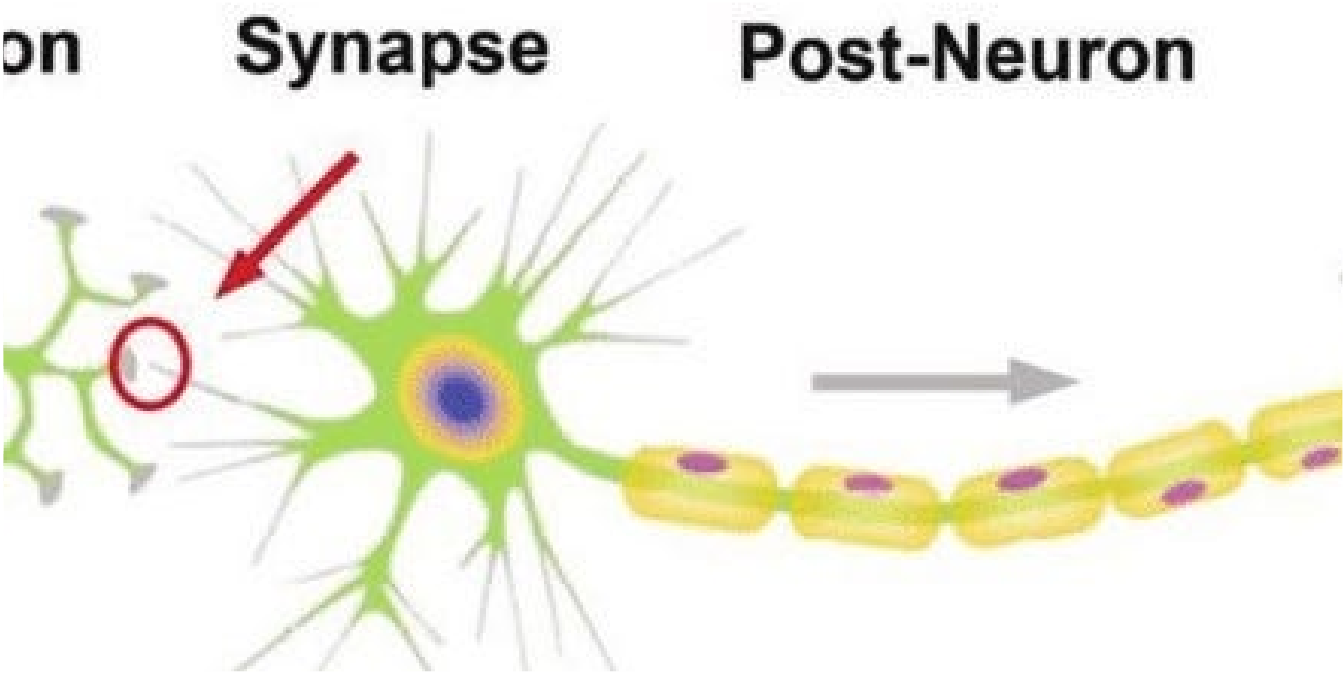
[1] En-Pei Hu [*1] Pu-Jen Cheng [1] Hung-Yi Lee [1]

Ming-Hao Hsu

[RL] Hierarchical Programmatic Reinforcement Learning via Learning to Compose Programs (ICML23)

Paper Link: Hierarchical Programmatic Reinforcement Learning via Learning to Compose Programs

4 min read · Jul 21



Ved Prakash

Liquid Neural Network : A adaptive way to train ML model

This is my 1st article in series of articles where I will review different research papers from the field of AI, machine learning, deep...

8 min read · Aug 7

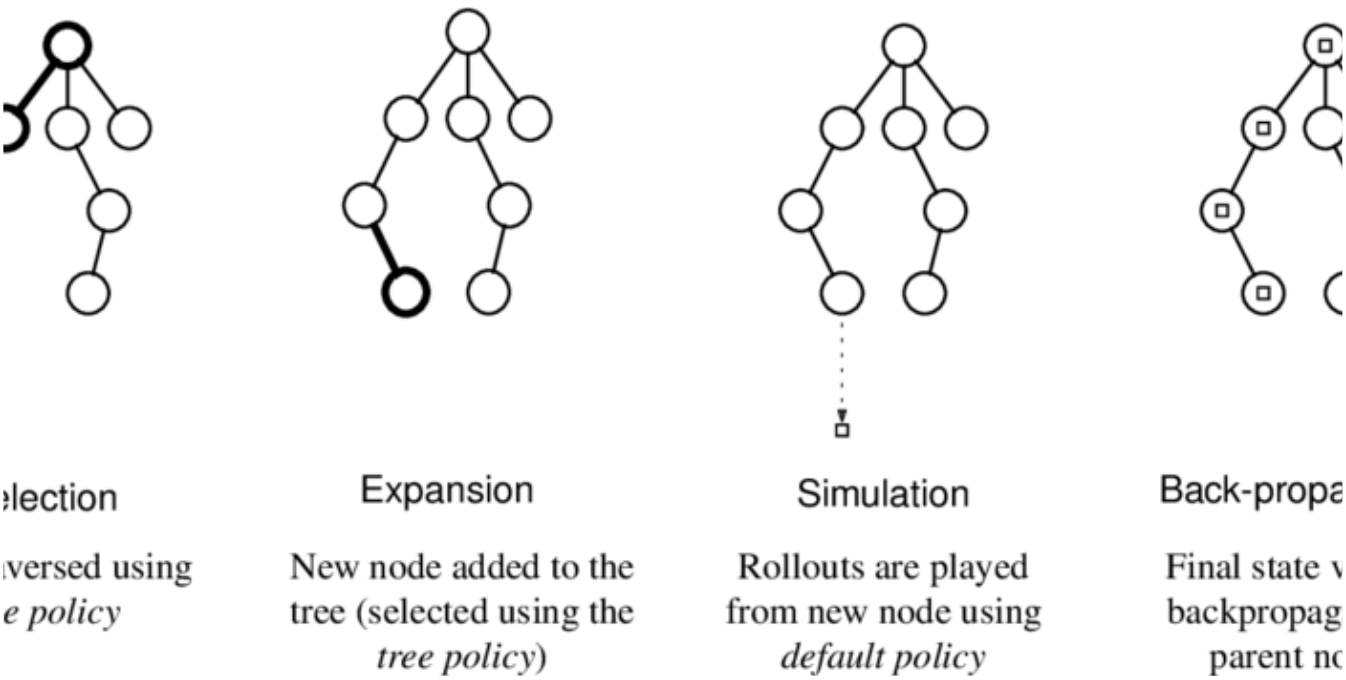
87 1

Jerry John Thomas

RL Series—Ep 1

This is actually a Reinforcement Learning series of short notes for me to revise when needed. Not intended for noobs but it will be...

6 min read · Jul 27



David Brown

Mastering Chess with Deep Learning: A Monte Carlo Tree Search and PPO Loss Approach

I. Introduction

· 6 min read · May 6