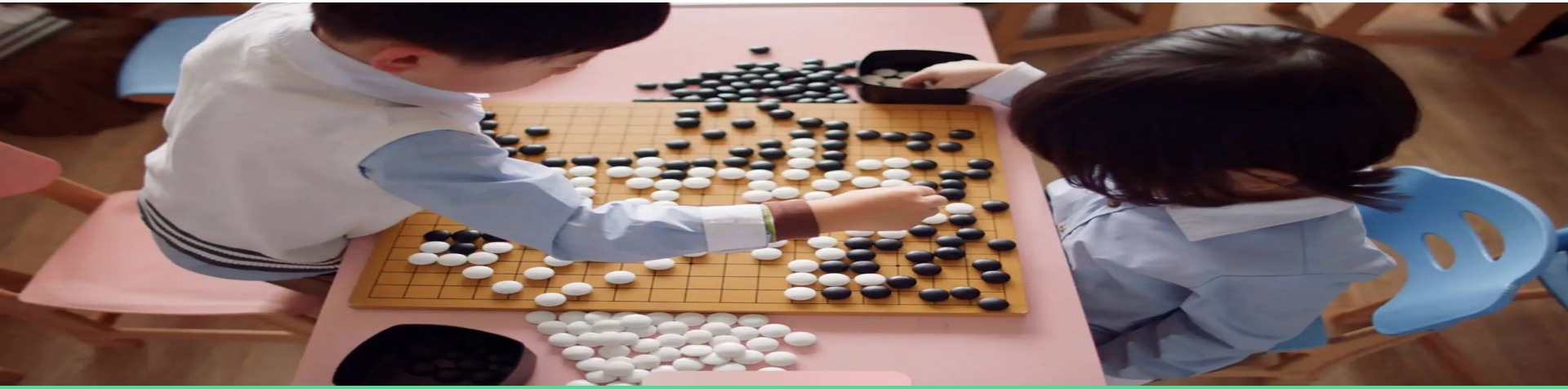
Reinforcement Learning

Fatima Loumaini

Introduction

- Go is a game originated in China over 3000 years ago. Winning the board requires multiple layers of strategic thinking.
- The goal is to surround and capture the opponent's stones and strategically creates spaces and territory.



Introduction

- As simple as the rules may seem, Go is profoundly complex. There are an astonishing 10 to the power of 170 possible board configurations - more than the number of atoms in the known universe. This makes the game of Go a googol times more complex than chess.
- Once all possible moves have been played, both the stones on the board and the empty points are tallied. The highest number wins.

Introduction

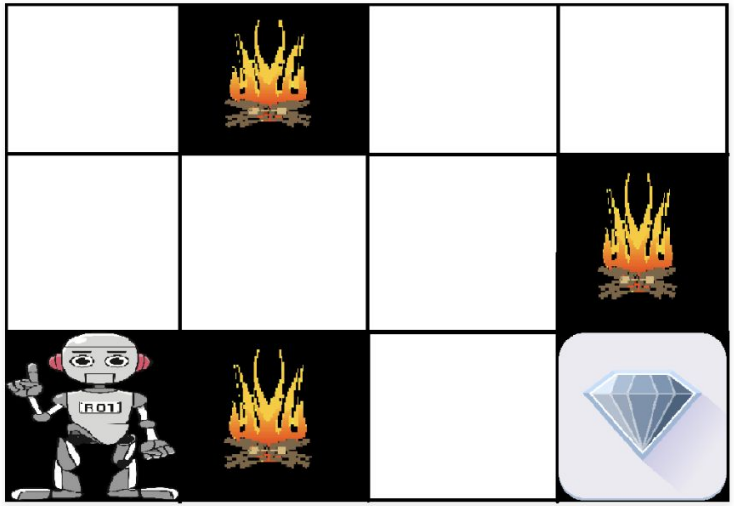
- Researchers from deepmind created AlphaGo, a computer program that combines advanced search tree with deep neural networks. These neural networks take a description of the Go board as an input and process it through a number of different network layers containing millions of neuron-like connections.
- AlphaGo Zero is a new version of AlphaGo that uses Reinforcement learning in which AlphaGo Zero becomes its own teacher and is no longer constrained by the limits of human knowledge. https://deepmind.com/blog/article/alphago-zero-starting-scratch

What is reinforcement learning?

Reinforcement learning refers to goal-oriented algorithms, which learn how to attain a complex objective (goal) or maximize along a particular dimension over many steps; for example, maximize the points won in a game over many moves. They can start from a blank slate, and under the right conditions they achieve superhuman performance. These algorithms are penalized when they make the wrong decisions and rewarded when they make the right ones – this is reinforcement.

Reinforcement Learning Example

We have an agent and a reward, with many hurdles in between. The agent is supposed to find the best possible path to reach the reward.



Main Points of Reinforcement Learning

- **Input:** The input should be an initial state from which the model will start
- **Output:** There are many possible output as there are variety of solution to a particular problem
- **Training:** The training is based upon the input, The model will return a state and the user will decide to reward or punish the model based on its output.
- The model keeps continuing to learn.
- The best solution is decided based on the maximum reward.

Use Cases for (un)supervised and reinforcement learning

	Supervised Learning	Unsupervised Learning	Reinforcement Learning
Banking	**Predict credit worthiness of credit card holders**: Build a machine learning model to look for delinquency attributes by providing it with data on delinquent and non-delinquent customers	**Segment customers by behavioral characteristics**: Survey prospects and customers to develop multiple segments using clustering	**Create a 'next best offer' model for the call center group**: Build a predictive model that learns over time as users accept or reject offers made by the sales staff
Healthcare	**Predict patient readmission rates**: Build a regression model by providing data on the patients' treatment regime and readmissions to show variables that best correlate with readmissions	**Categorize MRI data by normal or abnormal images**: Use deep learning techniques to build a model that learns different features of images to recognize different patterns	**Allocate scarce medical resources to handle different types of ER cases**: Build a Markov Decision Process that learns treatment strategies for each type of ER case
Retail	**Analyze products customers buy together**: Build a supervised learning model to identify frequent item sets and association rules from transactional data	**Recommend products to customers based on past purchases**: Build a collaborative filtering model based on past purchases by "customers like them"	**Reduce excess stock with dynamic pricing**: Build a dynamic pricing model that adjusts the price based on customer response to offers