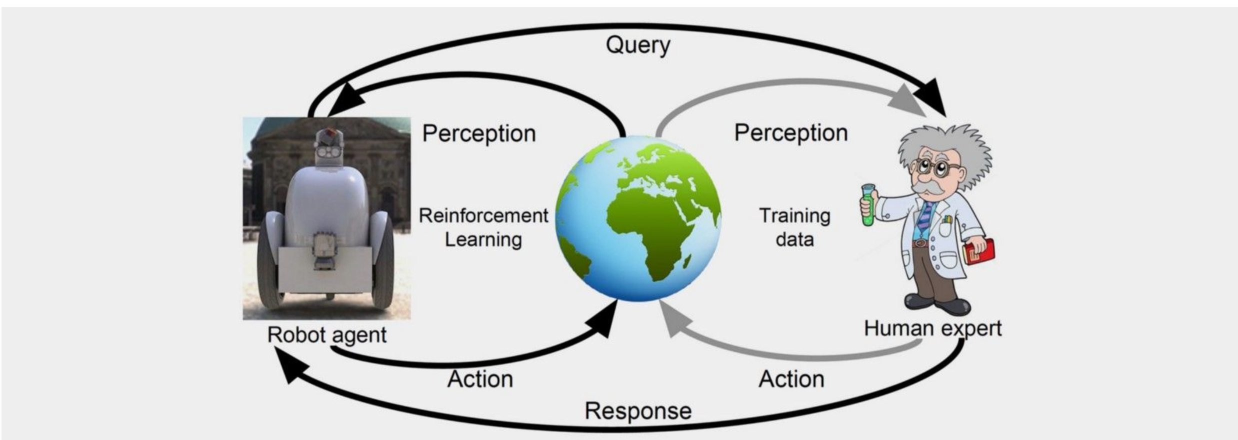
DINO AI GAME ANALYSIS

PROBLEM STATEMENT:

In this project ,we implement an end to end Deep reinforcement learning method to learn to control Chrome offline dinosaur game directly from high-dimensional game screen input. We propose special training methods after which , our **Reinforcement AI** is able to outperform normal humans.



What is Reinforcement Learning?

Reinforcement learning, in the context of artificial intelligence, is a type of dynamic programming that trains algorithms using a system of reward and punishment.
A reinforcement learning algorithm, or agent, learns by interacting with its environment. The agent receives rewards by performing correctly and penalties for performing incorrectly. The agent learns without intervention from a human by maximizing its reward and minimizing its penalty.

This automated learning scheme implies that there is little need for a human expert who knows about the domain of application.

EXPERIENCE LEARNING :

Experience stores state transitions, rewards and actions, which are necessary data to perform Reinforcement learning, and makes mini-batches to update neural networks. This technique expects the following merit:

- reduces correlation between experiences in updating Neural Network, increases learning speed with mini-batches.

SCOPE :

There are many challenges in current Reinforcement Learning research. Firstly, it is often too memory expensive to store values of each state, since the problems can be pretty complex. Solving this involves looking into value approximation techniques, such as Decision Trees or Neural Networks. There are many consequence of introducing these imperfect value estimations, and research tries to minimise their impact on the quality of the solution. Moreover, problems are also generally very modular; similar behaviours reappear often, and modularity can be introduced to avoid learning everything all over again. Hierarchical approaches are common-place for this, but doing this automatically is proving a challenge. Finally, due to limited perception, it is often impossible to fully determine the current state. This also affects the performance of the algorithm, and much work has been done to compensate this Perceptual Aliasing.
The main challenge here is the sheer volume of problems and possible actions. Hypothetically we may only think about a couple of problems or maybe ten at most. But real-life applications require millions of possible actions which create major complexities in the Large Action Space before it's rewarded. This will be a major scientific and engineering problem that needs to be solved before we can see scalable methods to train multi-purpose agents to do anything you can imagine.

The recent developments include -

- Learning from Demonstration/Imitation Learning
- Inverse Reinforcement Learning
- Safety in Reinforcement Learning
- Competitive and Cooperative Multi-agent Reinforcement Learning
- Google's AlphaGo
- Evolution Strategies, an alternative to overcome shortcomings of Reinforcement learning.
- Pathnet, a new Modular Deep Learning architecture from DeepMind.

CHALLENGES :

The goal of RL to make the agent perform many different type of tasks, rather than specializing in just one . This can be achieved by multi task learning and remembering the learning .

We've seen recent work of Google Deepmind on multi task learning ,where the agent learns to recognize a digit and playing Atari . However this is really a very challenging task when you scale the process . It requires a lot of training time and huge number of iterations to learn tasks .

Another challenge comes the way agent perceives the environment . In many real world tasks agent does not have the scope to observe the complete environment .This partial observations make the agent to take the best action not just from current observation , also from the past observations .So remembering the past states and taking the best action w.r.t current observation is key for RL to succeed in solving real world problems .

RL agents always learn from exploration and exploitation .RL is a continuous trial-and-error based learning , where agent tries to apply different combination of actions on a state to find the highest cumulative reward .The exploration becomes nearly impossible in real world . Let us consider an example where you want to make the robot learn to navigate in complex environment avoiding collisions . As the robot moves around the environment to learn, it'll explore new states and takes different actions to navigate .However it is not feasible to take best actions in real world where the dynamics of the environment changes very frequently and becomes very expensive for the robot to learn .

So to avoid the above problem,different other mechanisms have been applied on RL agents to make it learn. Few approaches like learning by mimicking the desired behavior, learning through demonstrations are being tried on robots to learn the environment in simulations .However in this way the learning becomes very specific to the environment and it loses the actual goal of generalized learning .

NECESSITY :

Since reinforcement depends on maximising reward and does not need labelled data, it can be used in environments where a certain action is desired but the means and attributes associated to it are not known.

It can be used to improve the efficiency of many systems like
- Elevator Control
- Dynamic Channel Assignment for Mobile Telecommunications
- intelligently trade stocks and options
- advanced self-driving cars, autonomous robot

LITERATURE SURVEY:

Inference from the paper we analysed:

Several approaches were used to achieve the AI that can play Chrome Offline Dinosaur Game. For the feature- extraction based algorithm, computer version methods can recognize the T-Rex and obstacles from the images. Carefully designed feature extraction algorithms can successfully abstract the state and AI built upon them can improve its performance significantly compared with naive baseline. MLP learned from online training can strengthen the AI further for it refines the parameters automatically by experience. However, feature-extraction based algorithm have their limits and can not outperform the human experts. For end-to-end reinforcement learning method, our result shows that it can successfully play the game by learning straightly from the pixels without feature extraction, and is much stronger than the feature-based method. Finally, specially designed training method can help us overcome the training difficulties caused by the properties of our game, which further improves our AI's performance and helps achieve superhuman results.

Our Views :

From the paper studied ,we can come to the conclusion that using reinforcement learning we can maximise the accuracy of any reward-punishment based system as the is stated by the numbers below. Also this type of learning covers an infinite array of problems because of its flexible nature. Since this does not need any previous attributes or any previous training data, it can tackle any problem to a certain level of mastery.

This also comes with its shares of problems since it is a randomized process to an extent so it can't be efficiently applied to systems with high number of variables, also current computation power is really a huge bottleneck to this type of learning.

Another problem regarding Reinforcement Learning which can also be its advantage is its ability to innovate and try methods which no human would be capable of designing. This is an advantage as it innovates and brings forward new methods to tackle the problem.

But this is rare and most of the times the randomized process gets stuck in a local minima which greatly affects the algorithm's efficiency.

Also the table below is a good statistic of a good Reinforcement AI, since the Google Dinosaur Game had a very low number of variables to deal with, it could easily find a good minima and increased the efficiency of the program, making it better than a human expert.

But in other games like Dota, where the number of variables are very high, to the level that computation becomes a bottleneck, it gets very tough for the AI to beat a human player.

So as of now the only games/problems that the AI can truly beat human players is the one where the number of variables is within a certain range which can be computed by today's computational systems.

Algorithm	Average 20	Max	Std
Keep-jump	41	111	23
Human-expert	910	1500	420
Human-optimized	196	4670	134
MLP	469	1335	150
Deep Q-learning	1216	2501	678

Link:

http://cs229.stanford.edu/proj2016/report/KeZhaoWei-AIForChromeOfflineDinosaurGame-report.pdf

OTHER REFERENCES:

- G. Bradski et al. The opencv library. Doctor Dobbs Journal, 25(11):120–126, 2000.
- V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves,
- I. Antonoglou, D. Wierstra, and M. Riedmiller. Playing atari with deep reinforcement learning. arXiv preprint arXiv:1312.5602, 2013.
  V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu,
- J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, et al. Human-level control through deep reinforcement learning. Nature, 518(7540):529–533, 2015.
- G. Tesauro. Temporal difference learning and td-gammon. Communications of the ACM, 38(3):58–68, 1995.