Autonomous Car using Deep Q Learning

Rajeev Srinivas
Department of Computer Science and Engineering
Sapthagiri College of Engineering, Bangalore, India

Prof. Madhuri C
Department of Computer Science and Engineering
Sapthagiri College of Engineering, Bangalore, India

Rahul Prabhu
Department of Computer Science and Engineering
Sapthagiri College of Engineering, Bangalore, India

Abstract

This paper offers with the simulation outcomes of a self reliant vehicle learning to force in a simplified surroundings containing simplest lane markings and static boundaries. Learning is finished the usage of the Deep Q Network. For a given enter image of the road captured by means of the car the front digital camera, the Deep Q Network computes the Q values (rewards) corresponding to the movements available to the self reliant driving car. These moves are discrete angles via which the automobile can steer for a fixed pace. The self sustaining using device within the vehicle enforces the movement that has the best reward. Our simulation results show high accuracy in learning by means of looking at the lanes and bypassing limitations.
Keywords: Self-Driving Cars, Reinforced Learning, Deeplearning, Deep Q Network

I. INTRODUCTION

The utilization of Artificial Intelligence (AI) and Machine Learning (ML) procedures to the advancement of self-driving frameworks is as of now a hotbed of research. Although a fully practical self-driving vehicle remains an extended way off, maximum principal automobile producers worldwide have reached advanced tiers of growing self- using cars. However, the current self sustaining using may be very expensive as it tends to rely upon incredibly specialized infrastructure and device like Light Detection and Ranging (LIDAR) for navigation, Global Positioning GPS) for localization and Laser Range Finder (LRF) for impediment detection [1]. Deep Q Network (DQN) is a form of strengthened getting to know in which the output of a CNN is not class, but Q values (rewards) given to actions because of the input states. The DQN agent learns a hit rules immediately from excessive-dimensional sensory inputs using end-to-stop reinforcement mastering. Recently, DQN has proved a success in the difficult area of traditional Atari 2600 games. A massive amount of categorized information of human using is essential to educate any autonomous deriving machine. DQN studying permits the agent to study from its very own conduct in place of classified information. After several hours of education, it learns to persuade smoothly in between the lanes and avoid limitations aided simplest by means of the visual information enter. This paper is prepared as follows: Section II explains the DQL principle and Section III offers the CNN and DQN systems.

II. DEEP Q LEARNING

Q Learning is a variation of reinforcement getting to know. In Q getting to know, there is an agent having states and corresponding moves. At any moment, the agent is in a few possible states. In the following time step, the state is transformed to different states(s) with the aid of doing some action. This motion is observed either by reward or punishment. The goal of the agent is to maximize the gain. The Q mastering set of rules is represented through the subsequent update formula

$$Q_t(s , a) \leftarrow Q_t((s_t , a_t) + \alpha (r - Q(s_t , a_t)). \gamma_{MAX} Q(s_{t+1} , a')$$

where $Q(s_t, a_t)$ represents the Q value of the agent in the state s_t, and action a_t at time t, rewarded with reward r_t. α is the learning rate and γ is the discount factor. The γ parameter is in the range [0,1]. If γ is closer to 0, the agent will tend to consider only immediate rewards. On the other hand, if it is closer to 1, the agent will consider future rewards with greater weight, thus willing to delay the reward. The Learning rate and Discount factor, described below, are the two most crucial parameters influencing the performances of the Q learning algorithm.

A. *Learning Rate*

Learning Rate determines the strength with which the newly obtained information will override the vintage information. A value of 0 will make the agent not learn any new information, at the same time as a value of 1 will make the agent keep in mind handiest the maximum current statistics. In absolutely deterministic environments, a learning rate is greatest. When the problem is stochastic, the algorithm nonetheless converges under some technical conditions on the studying rate that require it to lower to zero. In exercise, frequently a constant learning rate is used.

B. Discount Factor

Discount Factor shows the significance of future rewards. A value of 0 will make the agent brief-sighted by way of most effective thinking about contemporary rewards, at the same time as a value approaching to 1 will make it attempt for a long-term excessive reward. If the bargain value meets or exceeds 1, the action values can also diverge. Even with a reduction issue best slightly lower than 1, the Q Learning leads to propagation of mistakes and instabilities while the value feature is approximated with artificial neuralnetworks. In that case, it is recognized that beginning with a lower cut price thing and increasing it closer to its final price yields improved accelerated learning.

III. CNN OVERVIEW

Neural Networks, in general, Imitate the characteristic of organic neurons whose excitation signals can be on/off depending at the strength of the enter stimulus. The properly-known Multi-Layer Feedforward Neural Network which includes the mixture matrix operation and nonlinear activation characteristic plays quite well in solving class issues. For instance, the error in classifying the 0-9 digit snapshots in the MINST dataset is as low as three %.

However, inside the category of the CIFAR-10 dataset which has ten lessons of photos along with dogs, birds and so forth, the test blunders increase to about 50%. The terrible performance of MLPs in classifying photographs has given upward thrust to a new form of a Neural Network, known as Convolutional Neural Nets (CNN). A CNN, for example, with 3 convolution layers interlinked with max pooling layer showed only16.6% test error in classifying the pix in the CIFAR-10 dataset [8].

IV. PROBLEM STATEMENT

To make a self driving car which moves from source to destination using Deep Q-Learning and follows the path drawn without hitting any obstacles and with relevant given speeds and also help store the previous training.

V. RESULTS

The state-action cycle is repeated until the agent reaches the goal. If in the state-action cycle, the agent finally ends up hitting an impediment or crossing one of the bounding lanes, the learning session is discontinued, and the agent starts offevolved a new and fresh session. If it continues at the direction efficaciously, it's far rewarded for each obstacle it overcomes. The agent updates its information at the trajectory every 0.15 seconds, sends it to the mastering program and takes the precise movement dictated by means of the studying software fending off all of the limitations on its path.

A. Deep Q Network

CNN is essentially a classification structure for classifying images into labeled classes. The various layers of the CNN extract image features and finally learn to classify the images. Hence, the outputs of a typical CNN represent the classes or the labels of the classes, the CNN has learnt to classify. A Deep Q Network is a variation of CNN. The outputs are not classes, but the Q values (or rewards) corresponding to each of the actions the agent has learnt to take in response to its state in the environment.

The Deep Q Network computes the Q value (reward) for a given state of the environment using the following algorithm.

```
    Initialize all Q values;
Repeat: (for each episode)
Choose a random state s;

WHILE (true) Do (Repeat for each step in the episode)
        Select action a ∈ A(s) according to the policy
        execute the action, a;

            Observe new state, s';

            Receive immediate reward, r;
        Update Q(s, a) using Equation (1);

    Until s is one of the goal states, Go to Repeat;

    Until the desired number of episodes has been
investigated; Stop.
```

VI. Conclusion

In this paper, we proposed a simulation study of an autonomous agent learning to drive in a simplified environment consisting of only lane markings and static obstacles. We used a Deep Q Network to train the agent in the simulated environment. Future prospects of our study include verifying the simulation results and further tuning them using Robocar (fully functional driving car, 1/10th the size of standard commercial car, driving on the laboratory floor. The verification results are expected to demonstrate that learning autonomous driving in a simulated environment is a step towards driving on real streets.

References

[1] M. Ali, "Detection of market streets direction,"
[2] K. Arulkumar N. Dilok anakul, M. Shaan, and A. A.Bharat, “Differnetiating options for deep reinforcement learning,”
[3] Y.Bengo, learn deep architecture for AI. Trends in Machine Learning.
[4] M. Day and Y Lin, "Deep Learning for Accurate Analysis on Google Play Consumer Review," 2018 IEEE International Conference on Information Reuse and Integration (IRI).
[5] C. Duke, A. Coheen, & M. L. Littman, An object-oriented valuation for accurate reinforcement learning. Proc. Int. Conf. Mach. Learn.
[6] G. E. Hinton, N. Srinivas, A. Krizevski, I. Sutskelever, &
[7] R. R. Salakhutdi, (2012). Upgrading neural networks by preventing co-adaptating of feature detectors.
[8] Y. Ji, E. Shelhamar, J. Donahleue, et al., “Caffe: convolutional architecture for fast feature embedding,”
[9] K. M. I. Khalilullaer, S. Ota, T. Yasuhuda and M. Jindal, "Improvement of robot navigation method using deep learning".

