Islington College

Artificial Intelligence

CU6051NI

Self-Driving Virtual Car Final Report

Submitted By:

Hrishav Tandukar (15043026)

Erin Rai (15043008)

Group: L3C4

Date: 31st January, 2018

Word Count: 4979

Submitted To:

Mr. Ashim Lamichhane

Lecturer, IT Faculty.

Semester: Autumn

Abstract

Artificial Intelligence has always been a controversial topic in real world. But being a programmer, it is an interesting topic to learn. Module called Artificial Intelligence of third year provides us opportunity to develop an artificial intelligence on our own. The module artificial intelligence provided knowledge about problem solving and heuristic search, knowledge representation, game theory, natural language processing and machine learning.

The document is based on the project, self-driving car. The car is a modelled version of a can which is virtual. It will learn how to drive itself. The key word here is learn. The car is not given any rules on how to operate on environment. It figures out everything on its own. In order to achieve this the term Deep-Q-Learning is used which is a part of Machine Learning. The term Deep-Q-Learning is very vast and includes Q-Learning and Neural Network.

The project required a lot of hard work, planning and dedication. For successful completion of project and in order to check logic of the program pseudo codes had been developed. State diagrams has also been created which shows not only algorithms that has been used in order to carry out this project but also indicates logics and whole overview of the system. Various hardware and software components are to be recognized while developing a software. Therefore, at first various hardware and software requirements were also identified in order to carry out the project.

Acknowledgement

In order to research and learn about the project, this required a lot of assistance and guidance from many people. We are extremely, thankful to them for providing us such supervision and assistance.

We respect and extend our gratitude to Islington College (London Metropolitan University), for providing us an opportunity to work on the project and providing us all the resources, supports and guidance. Especially, we extremely thankful to the supervision and guidance from the lecturer. We are thankful to and fortunate enough to be able to work for the Project and learned and gain vast knowledge.

We are thankful to and fortunate enough to get constant support, encouragement and guidance from Lecturer and out Colleague which helped us a lot in the project.

Table of Contents

Table of Figures

1. Introduction

The documentation report explains and delivers the topic introduction, research methodologies, mind map for the topic and the description of how the artificially created bot which is a car learns to self-drive from one destination to another destination in a given environment.

1.1. Preface

The topic is based on the term Machine Learning. The term Machine Learning consists of many methodologies. Deep-Q-Learning is one of the popular methodology which has been used in many sectors from industrial sector to education sector.

The project is based on the Deep-Q-Learning, which is explained in detailed - later part of the report. Self-Driving car which is one of the emerging technology in world. In order to get the idea and knowledge on how the self-driving-car actually works- the project covers the concept and methods to build simple modelled version of car which behaves like a real world self-driving car.

1.2. Present Scenario and Problem Statement

Machine Learning (ML) has revolutionized the world of computers by allowing them to learn as they progress forward with large datasets. ML builds algorithms, which when exposed to high volume of data, can self tech and evolve (Ghosh, 2017).

Machine Learning has become one of the most discussed topic all around the world. It is one of the most useful and widespread manifestation of Artificial Intelligence currently in-use. They are used in a variety of ways including cybersecurity, enhancing recommendation engines and optimizing self-driving car (Morgan, 2015). Different sectors including Financial services, Government, Health Care, Marketing and sales, oil and gas, transportation and many more. One of them is **Self-Driving Car**.

The world of self-driving autonomous cars are one of the greatest and latest topic of Machine Learning. Companies across the globe from car makers such as Tesla, Audi and Nissan, to tech firms like Google and Aurora are spending millions to get the technology get on the road. The automated Self-Driving Cars are being tested in the real world in order for them to run in the real world. The process of learning is being done for the Self-Driving Cars. There are many companies that wants to launch the Automated Cars. (Auto Express, 2018)

Mainly Google is famously working on self-driving vehicles for the open road with full autonomy. The internet giant plans to have their driverless cars on the road by 2018.

Tesla is taking a daring chance on autonomous vehicles. Instead of developing a whole new car specifically for autonomous driving (Husain, 2018).

Spurred on by tech companies entering the automobile world, most major automobile manufacturers such as Ford, General Motors, BMW, Mercedes-Benz, Nissan and Toyota are all testing driverless car systems. Some claim that as early as 2020 they will be mass producing self-driving vehicles (Husain, 2018).

Self-Driving Cars have been already build. But it has not been able to come in the market due to many factors like it needs to learn in every type of environment. It takes time to learn for the automated cars and adapt in the environment.

Even though many companies have been investing a lot and also have completed building a self-driving-car. But they have not been able to launch them, due to the different risks that may or might cause by them to the environment. Due to these reasons, the self-driving car are not launched in the market.

So, the research on the Deep-Q-Learning are still being conducted. There are various types of learning algorithms. These may differ according to the situations and the conditions of the applications. Trying different learning algorithms allows to select the optimum algorithm for the system.

1.3. Coursework as a Solution

The term Machine Learning is one of the emerging topic. The project concept is based on the Deep-Q-Learning which is a very vast topic in itself. So, the topics included in the concept were very interesting and challenging. The topic Neural Network and the Q-Learning.

The concept of the project can be used in the real world Self-Driving-Car. Even though the concepts are basics, it will be a good start for further development. Similarly, it can be used in development of robots which can save people from different situations like earthquakes, fire and many more. The concept like Neural Network and Q-Learning can also be used individually for different purposes.

1.3.1. Coursework Aim

Some of the aims of this project are as follows:

a. To delivery our final application i.e. virtual self-driving car.
b. To work in a group and complete the project
c. To develop system in which Deep-Q-Learning concept is implemented.
d. To complete the project in scheduled time
e. The coursework aims to give the concept about how the neural network and Q-Learning can be implemented in the Self-Driving car.
f. To develop an agent that learns itself in the environment and reaches the given locations.
g. To implement the python programming language for developing a system.
h. To get basic idea and knowledge on how the Machine Learning related system works through system and documentation.

1.3.2. Coursework Objectives

In order to complete the aim, following objectives should be completed:

a. Research on the topic Machine Learning
b. Learn about different methodologies of Machine Learning
c. Plan a schedule and work on project, accordingly
d. Researching the term Deep-Q-Learning.
e. Enriching knowledge on Neural-Network and Re-enforcement Learning.
f. Learning and implementing Python as programming language.
g. Learning to create a Windows Application
h. Finally, creating a documentation for the explanation of the topic.

2. Background

2.1. Requirements to run the program

The end-user will receive the following files.

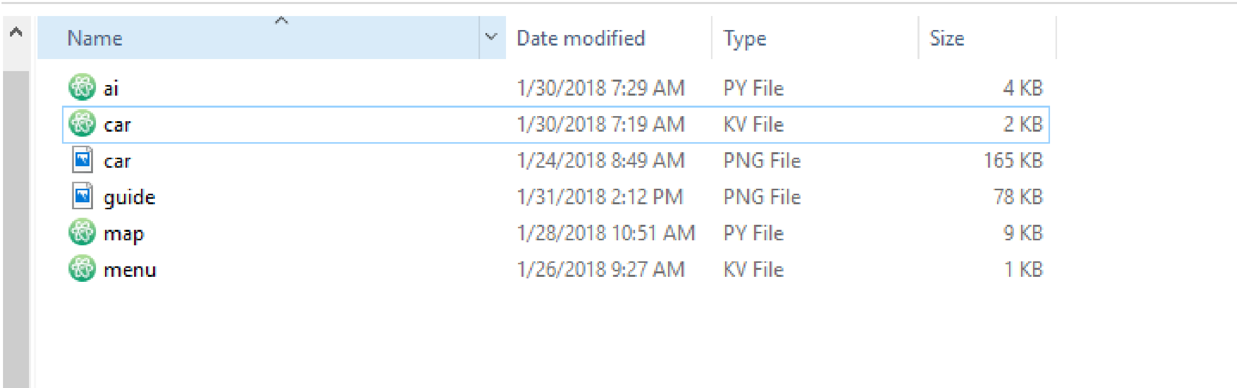

Figure 1: File Structure

In order to run the application the user should have installed the kivy, numpy and pyTorch library.

In order to install kivy and pyTorch on Windows :

Open Anaconda Prompt as Admin

Open Anaconda Prompt from your windows search by right-clicking on it and selecting Run as administrator (This is very important for package permissions)

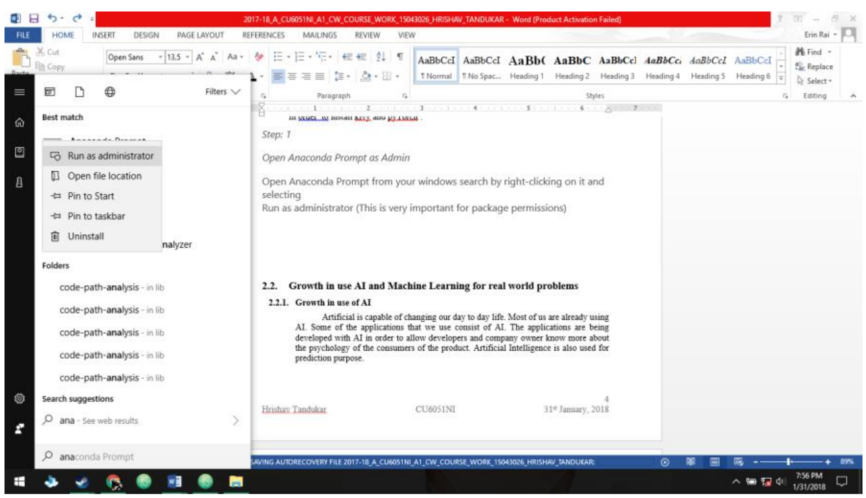

Figure 2: Anaconda Prompt

Once the Anaconda Prompt is open, type in these commands in the order specified Enter y to proceed when prompted.

conda install -c anaconda python=3.6.1

conda install -c peterjc123 pytorch=0.1.12

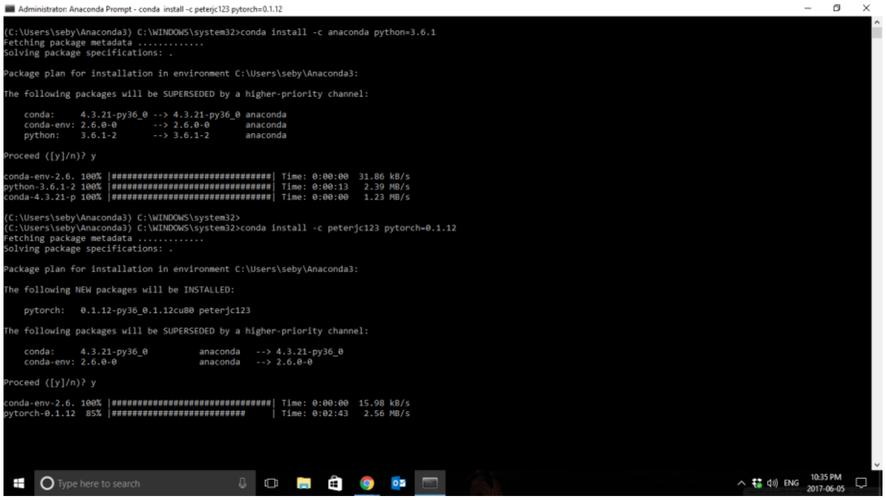

Figure 3: pytorch command install

Change the directory in the Anaconda Prompt to the known path where the kivy wheel was downloaded. (For me this path is C:\Users\seby\Downloads, so change the below command accordingly for your system)

cd C:\Users\user\Downloads

Once the path has been changed succesfully, you should now enter these commands in the Anaconda prompt to install Kivy

1. pip install docutils pygments pypiwin32 kivy.deps.sdl2 kivy.deps.glew
2. pip install kivy.deps.gstreamer
3. pip install Kivy-1.10.1.dev0-cp36-cp36m-win_amd64.whl

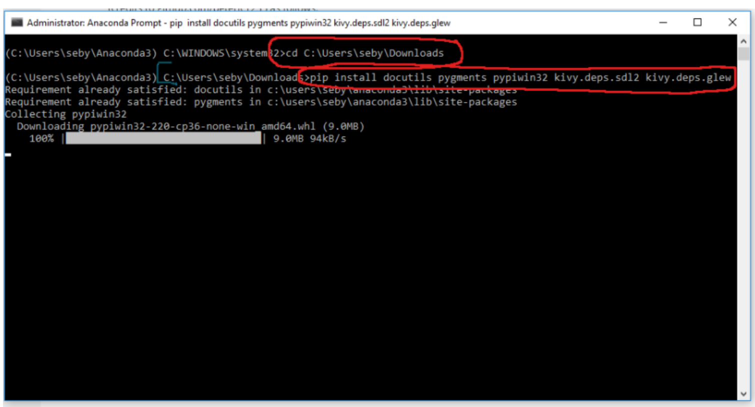

Figure 4: Kivy First Step installation

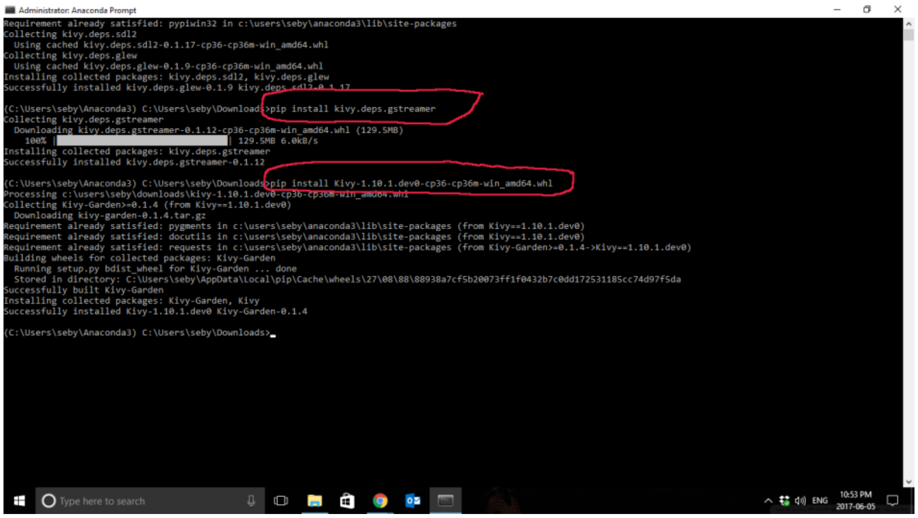

Figure 5: kivy second step installation

Kivy is now setup on your anaconda environment successfully (Eremenko, 2017)

2.2. Growth in use AI and Machine Learning for real world problems

2.2.1. Growth in use of AI

Artificial is capable of changing our day to day life. Most of us are already using AI. Some of the applications that we use consist of AI. The applications are being developed with AI in order to allow developers and company owner know more about the psychology of the consumers of the product. Artificial Intelligence is also used for prediction purpose.

Most consumers are not aware of various machine learning capabilities. According to research of Deloitte, about 68 percent of consumers are already using some form of artificial intelligence (AI)/machine learning (ML) on their smartphones. But users are not aware about how AI is being used in the application that they had been using. AI is mostly used for predictive text, driving route suggestions, and voice assistants. (deloitte, 2018)

The result obtained from survey can be used for various business purposes. AI can be used in order to extract various data and information of the interest and psychology of the customers. After, extracting interest of all the customers, company can reach to a conclusion about what can be a potential product that can be developed for its customers. Therefore, at last a suitable product is developed according to the interest of the consumers. (techrepublic, 2018)

According to a research, 75 percent of businesses are now investing in Big Data, the role of AI and machine learning is set to increase dramatically over the next five years. In year 2017, a quarter of organizations is said to be spending 15 percent or more of their IT budget on machine learning capabilities, and the number of machine learning examples is expected to rise in the near future days. (Feldman, 2018)

Over two-thirds of consumers are already using artificial intelligence (AI) (deloitte, 2018). The majority of people have already begun using the technology to make smarter decisions and become more productive in their personal lives, and that can also translate to the business world (techrepublic, 2018).

2.2.2. Machine Learning for Real World Problems

Machine Learning is emerging and evolving in a great pace. Machine Learning solves many real world problems. The project concept Deep-Q-Learning along with its sub topic Neural-Network and Q-Learning individually. Some of them are listed below:

2.2.2.1. Manufacturing

In Fanuc, a robot uses deep reinforcement learning to pick a device from one box and putting it in a container. Whether it succeeds or fails, it memorizes the object and gains knowledge and train’s itself to do this job with great speed and precision (Maruti Techlabs, 2017).

2.2.2.2. Inventory Management

Reinforcement learning algorithm can be built to reduce transit time for stocking as well as retrieving products in the warehouse for optimizing space utilization and warehouse operations (Maruti Techlabs, 2017).

2.2.2.3. Power Systems

Reinforcement Learning and optimization techniques are utilized to assess the security of the electric power systems and to enhance Microgrid performance. Adaptive learning methods are employed to develop control and protection schemes (Maruti Techlabs, 2017).

2.2.2.4. Image Compression

Neural Network can be used in the character recognition. The devices like Palm Pilot is an example. Similarly, Handwriting characters recognition also uses Neural Network (cs.stanford.edu, 2017).

2.2.2.5. Stock Market Prediction

The day-to-day business of the stock market is extremely complicated. Many factors weigh in whether a given stock will go up or down. Neural network can examine a lot of information quickly and sort it all out, they can be used to predict stock prices (cs.stanford.edu, 2017).

2.3. Technical process overview

2.3.1. Technical Terms

2.3.1.1. Machine learning

Machine learning is a sub field of artificial intelligence (AI). Machine learning allows applications to become more accurate in predicting outcomes without being explicitly programmed. The basic premise of machine learning is to develop algorithms which can receive input of various data and information and then use statistical analysis in order to predict an output value within an acceptable range. (Rouse, 2017)

Algorithms of Machine learning algorithms are often categorized as being supervised learning, unsupervised learning or re-enforcement Learning. In supervised learning, machine learns from the given input and output of data and information whereas in supervised learning data are not required in order to train the machine but an interactive approach called deep learning is used in order to train machine. Similarly, Reinforcement Learning is based on the environment and the agent which takes action and change its state and learns with the reward system. (Rouse, 2017)

Deep-Q-Learning which is the combination of two methods of Machine Learning i.e. Q-Learning and Neural Network

2.3.1.2. Neural network

The inventor of the first neurocomputer, Dr. Robert Hecht-Nielsen, defines a neural network as – "a computing system made up of a number of simple, highly interconnected processing elements, which process information by their dynamic state response to external inputs." (Tutorialspoint, 2017)

Neural networks are organized in layers. Layers of neural network consists many interconnected 'nodes' which contains 'activation function'. Patterns in network are displayed through the 'input layer', which communicates to one or more 'hidden layers' where the actual processing is carried out through an application with the help of weighted 'connections'. The hidden layers are then linked to an 'output layer'. (Lin, 2011)

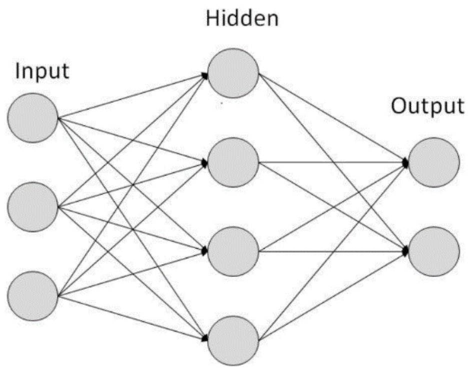

Figure 6: Graphical representation of layers of neural network.

Activation function plays a vital role in artificial neural network. It helps system or software to learn and develop sense of something which is complicated and non – linear in case of complex functional mappings between inputs and response of a variable. Activation function serves the purpose to convert an input signal of a node in an artificial neural network to an output signal. After carrying out this process, now output signal is used as an input in the next layer in the stack (Walia, 2017).

Backpropagation, short for "backward propagation of errors," is an algorithm for supervised learning of artificial neural networks using gradient descent. Given an artificial neural network and an error function, the method calculates the gradient of the error function with respect to the neural network's weights. It is a generalization of the delta rule for perceptron to multilayer feedforward neural networks (John, et al., n.d.).

2.3.1.3. Reinforcement learning

Reinforcement Learning is a type of Machine Learning. Reinforced learning allows machines and software agents to automatically determine the ideal behavior within a specific context, in order to maximize its performance. In reinforced learning, simple reward feedback is required for the agent in order to learn its behavior. Reward point are given for machine or accuracy of output. (Trymachinelearning, 2017)

List of reinforcement Machine learning algorithms are as follows:

- Temporal difference learning
- Q-learning
- Learning Automata
- SARSA

Bellman's equation is a dynamic programming equation which involves taking the optimal decisions across those periods (Bellman, 1954).

Markov-decision making process is used in Reinforcement learning which maps the states to action, so as to maximize a numerical reward over time. A Markov decision process (known as an MDP) is a discrete-time state transition system. It can be described formally with 4 components (ocw.mit.edu, n.d.).

a. State
b. Actions
c. Transition probabilities
d. Real Value Reward

Q-Learning is a family of RL algorithms. Q-Learning attempts to learn the value of being in a given state and taking a specific action there. In Q-Learning, an agent tries to learn the optimal policy from its history of interaction with the environment. A history of an agent is a sequence of state-action-rewards (artint.info, 2017).

Experience replay is one of the tricks that has been discovered to be one of the most important optimizations to make that will enable the neural network to learn in a reasonable time (or even converge). This is because this technique breaks the concept of continuity between any two transitions while giving the network a chance to also reinforce its knowledge of previous experiences more efficiently (Sefair, 2017).

2.3.2. Technical Process

In the project is a modelled version of a car (so it won't be driving on the streets of real cities) but still - it will learn how to drive itself. And the key word here is *learn,* because the car will not be given any rules on how to operate in the environment before hand - it will have to figure everything out on its own. And to achieve that we will be using Deep Q-Learning.

Deep Q-Learning is the result of combining Q-Learning with an Artificial Neural Network. The states of the environment are encoded by a vector which is passed as input into the Neural Network. Then the Neural Network will try to predict which action should be played, by returning as outputs a Q-value for each of the possible actions. Eventually, the best action to play is chosen by either taking the one that has the highest Q-value, or by overlaying a Softmax function.

In Neural Network, there is input layer in which the inputs are the three signals, and orientation which are the state of the car. Then, in the hidden layer the sum of input and weights are calculated and activation function which is used as rectifier function. Likewise, the hidden layer gives the possible Q-Values and the best Q value is calculated with the Softmax Function. After that experience replay which is one of the most important process involved in learning process. The memory is defined which keeps the track of the previous experiences and observation to re-train the model with those previous experiences. Then random samples are taken and neural network is trained with few samples. To make agent perform well in mid-term and long-term, not only immediate rewards but also the future rewards are taken. In order to implement gamma is passed. Thus DQN will learn to maximize the discounted future reward on the given State.

Then the agent will randomly select its action at first by a certain percentage with learning rate. At first the agent try different things and with the experiences it gets will predict the reward it gets and selects the best action possible. In this way it learns to drive itself avoiding the obstacles.

Here, Reward, action, state, orientation, Learning rate, gamma and temperature of the softmax function plays an important role in the learning of the agent.

2.4. Initial System Architecture

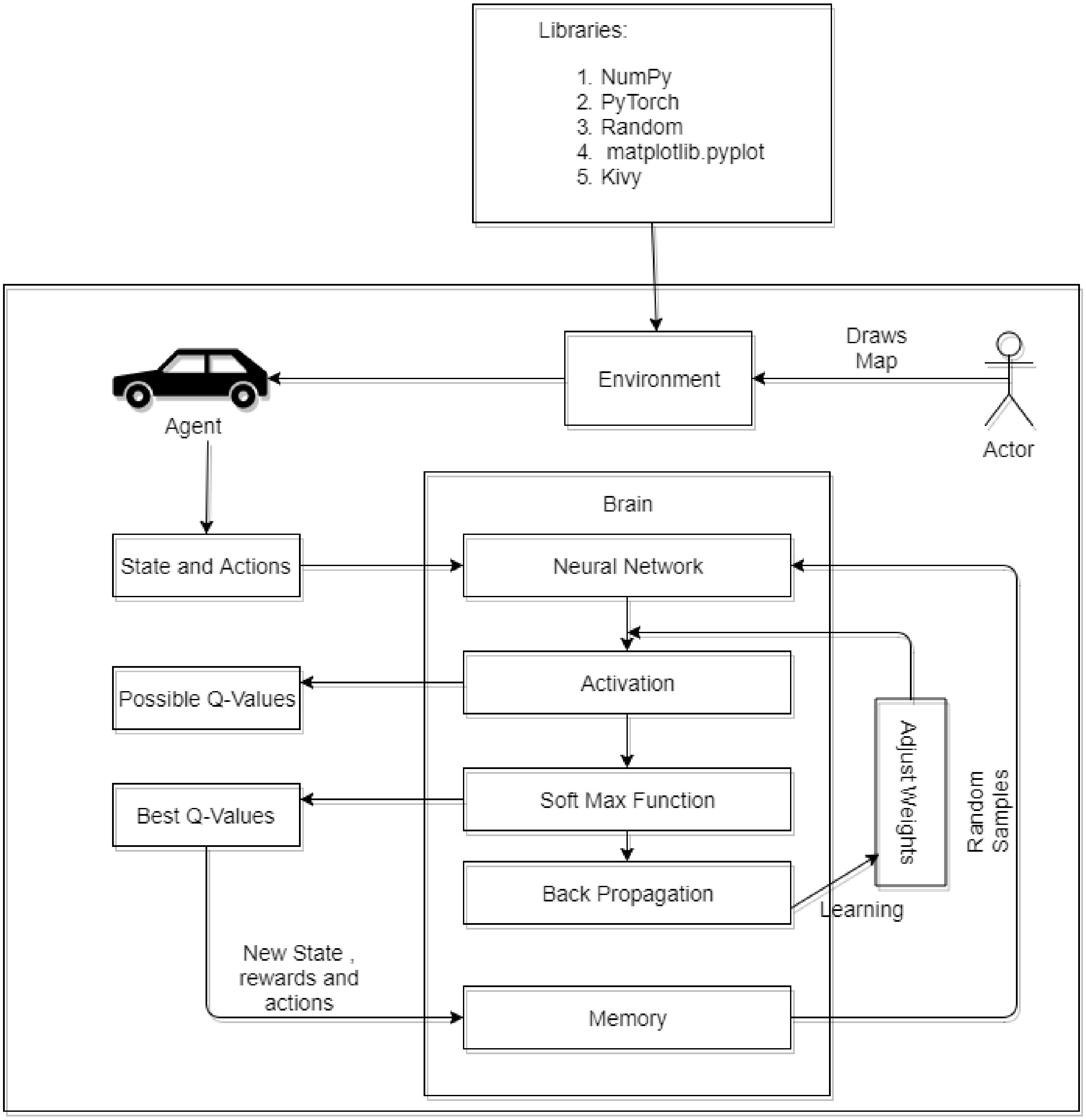

Figure 7: Complex Architecture Diagram

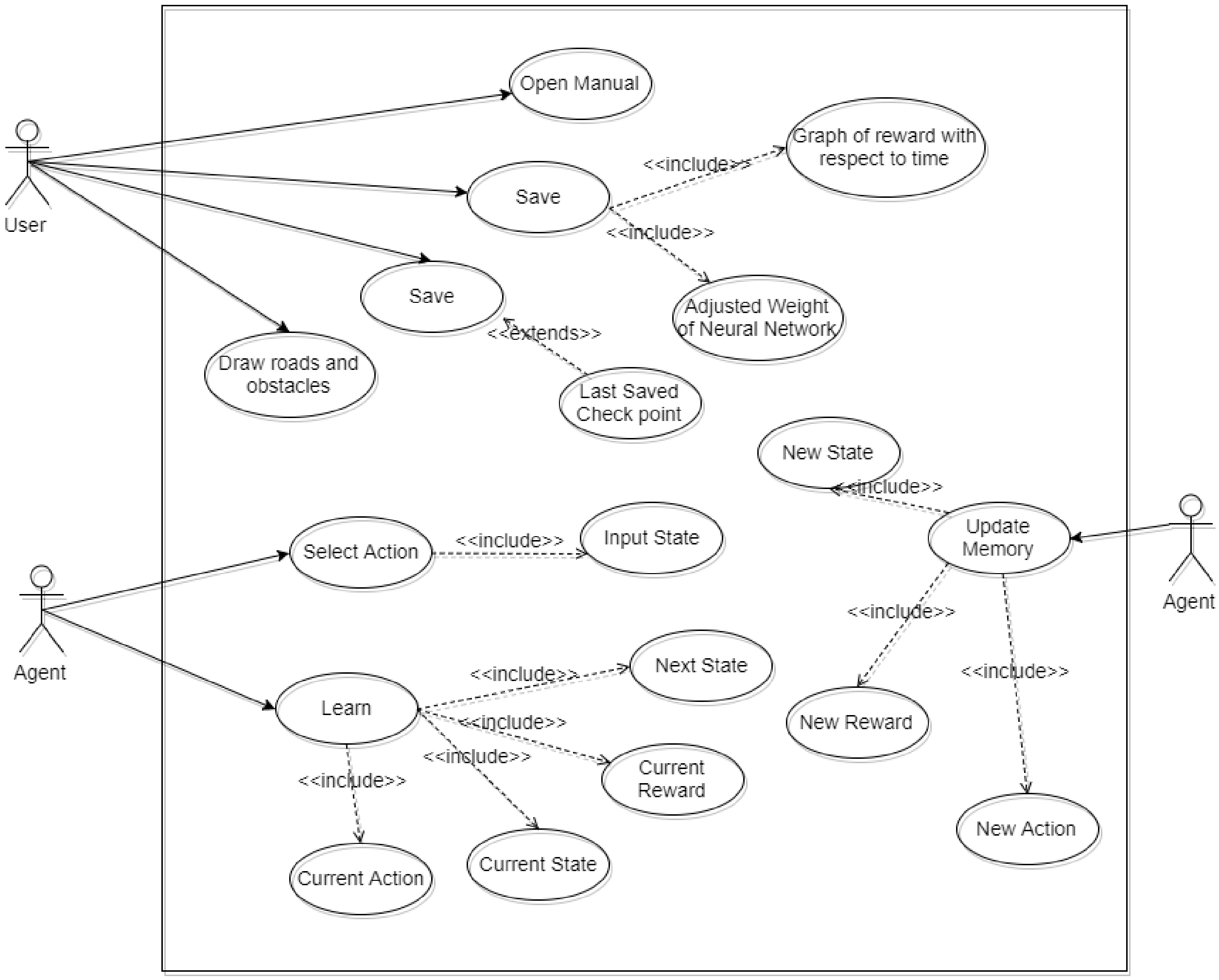

Figure 8: Use case diagram

2.5. Analysis of similar system

2.5.1. Introduction

We have carried out research on similar system that has been developed using path finding. There are various systems which implement concept of neural network. Here, we have shortlisted two major system which are similar to our project i.e. virtual self-driving car.

2.5.2. Deep Traffic

During the phase of Topic Selection for the project, while searching for the topic – the term Deep Traffic was introduced. DeepTraffic is a deep reinforcement learning competition. The goal is to create a neural network to drive a vehicle (or multiple vehicles) as fast as possible through dense highway traffic. What you see above is all you need to succeed in this competition (selfdrivingcars.mit.edu, n.d.).

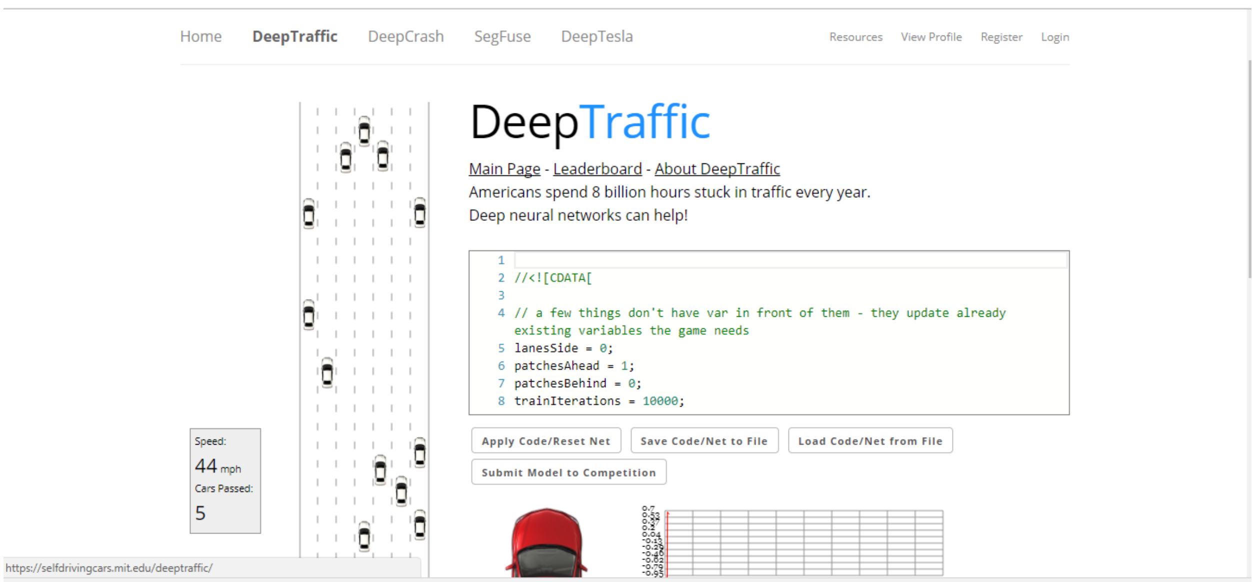

Figure 9: Deep Traffic

2.5.3. Roomba Red

Figure 10: Roomba Red.

Roomba vacuums are equipped with a remarkable artificial intelligence system that automatically adjust itself according to floor type and carpet height, and steps up the power when extra dirt is detected. The Roomba also uses the concept of deep-q-learning (Robot Store, 2017).

2.5.4. Self-driving car

Self-driving cars are being developed by Google. Partially-autonomous car may require human driver in order to control the car whereas Google is trying their best in order to develop fully-autonomous car which may not even require a sterling wheel for controlling the car. (Ucsusa, 2017)

Self-driving car uses software like radar for maintain internal map of their surroundings. Software used self-driving car of processes those inputs, plots a path, and sends instructions to the vehicle's "actuators," which control acceleration, braking, and steering. Hard-coded rules, obstacle avoidance algorithms, predictive modeling, and "smart" object discrimination (ie, knowing the difference between a bicycle and a motorcycle) help the software follow traffic rules and navigate obstacles. (Ucsusa, 2017)

Layers of autonomy defined in self-driving car

Different cars are capable of different levels of self-driving, and are often described by researchers on a scale of 0-5.

Level 0: All major systems are controlled by humans

Level 1: Certain systems, such as cruise control or automatic braking, may be controlled by the car, one at a time

Level 2: The car offers at least two simultaneous automated functions, like acceleration and steering, but requires humans for safe operation

Level 3: The car can manage all safety-critical functions under certain conditions, but the driver is expected to take over when alerted

Level 4: The car is fully-autonomous in some driving scenarios, though not all

Level 5: The car is completely capable of self-driving in every situation (Ucsusa, 2017)

2.5.5. Alpha Go

Alpha Go is a computer program which is developed by Google. This programs plays board game Go. AlphaGo has been able to defeat world champions of board game Go. AlphaGo Zero is capable of doing so due to the use of reinforcement learning which makes AlphGo Zero its own teacher. First of all, system does not know anything about Go. With the help of neural network and powerful search algorithm it then plays against itself. As this AI starts to play, the neural network is tuned and updated to predict moves, as well as the eventual winner of the games. (Ucsusa, 2017)

2.5.6. Conclusion

So, similar topics were searched. With the research, there are various algorithms and concepts used while developing virtual self-driving car. The Deep-Q-Learning concept was finalized. This concept is not only limited for this project but also feasible to be used in various other fields. The concept is very vast topic in itself. So, the topic was challenging and interesting to learn and research on.

2.6. Overview of Technical Aspects

2.6.1. Python

Python is a powerful high-level, object-oriented programming language created by Guido van Rossum. It has simple easy-to-use syntax, making it the perfect language for someone trying to learn computer programming for the first time. Spyder is used as an IDE (Integrated Development Environment) in order to carry out backend coding task for software development. Due to the easy syntax in python programming and user friendly environment provided by IDE we have used python in order to develop our system. (Programiz, 2018)

2.6.2. Kivy (Windows Application Framework)

Kivy runs on Linux, Windows, OS X, Android, iOS, and Raspberry Pi. It can natively use most inputs, protocols and devices including WM_Touch, WM_Pen, Mac OS X Trackpad and Magic Mouse, Mtdev, Linux Kernel HID, TUIO. The framework provided by Kivy is stable and has a well-documented API. The toolkit of Kivy comes with more than 20 widgets, all highly extensible. Many parts are written in C using Cython, and tested with regression tests. Therefore, Kivy is used in order to develop our virtual self-driving car. (Kivy, 2018).

So Kivy, is used in order for developing the GUI form the Windows Application Framework.

2.6.3. Libraries

a. Numpy : Working with array
b. Random : In order to generate random samples
c. Os: Load the saved model
d. pyTorch: Implementation of Neural Network
e. Kiy: For desktop application
f. Matplotlib: To plot the Graph

3. Artifacts Review

3.1. Artifacts Description

A virtual Self Driving Car is the autonomous bot that implements the Deep-Q-Learning in order to learn and improve at self-driving.

The Virtual car learns with the methodology of Deep-Q-Learning which involves Q-Learning and Neural Network. The car is implemented and trained in the virtual environment. A virtual self-driving car with three signals is created in which goals i.e. start-destination and end-destination is defined for the car. At first the car runs at random direction with random state and actions. As the car experiences the environment, the car learns to reach the goals this is due to the previous rewards, state and actions, saved in memory. The car has three possible angles with which it can perform the action. Similarly, the car chooses the best action from possible actions and reaches the destination with minimum time. This is obtained with Q-Learning and Neural Network both.

Virtual car is tested in three phases. At first stage, car will go to the destination make a round and come back. Likewise, in second stage, a simple road is developed, through which the car will have to reach destination along the road and reach the destination. In third stage, many obstacles is set and the car has to pass through obstacle and reach the destination. Finally, a zig zag road is set and the car has to reach the destination through the road.

So, the concept of Deep-Q-Learning is used in the development of the System. The concept includes the topic Neural Network and Q-Learning algorithm of Reinforcement Learning.

Figure 11: Manual for the Game

3.2. Artifacts training data sources

There are no external training data sources as the artifacts is based on the Deep-Q-Learning methodology. This methodology involves the re-enforcement learning which learns from the experience in the environment.

As, the training data sources are fed into the Neural Network, the inputs for the artifacts are the three signals which are given by the three sensors of the car and two orientation. The sensor senses and responses accordingly. If the any obstacles comes in front of the car then the sensor senses it due to the reward system we have provided for every state and action it takes. Similarly, positive orientation is for the orientation of car from the goal. Negative orientation is for exploration of the map.

So, three signals and two orientations are training data source of the Artifact. These are inputs for the Neural Network. The Neural Network update the weights accordingly with the back propagation and selects the best action to perform with respect to the reward given to the action.

3.3. Graphical Representation

3.3.1. Week 1 Mind Map (Scenario)

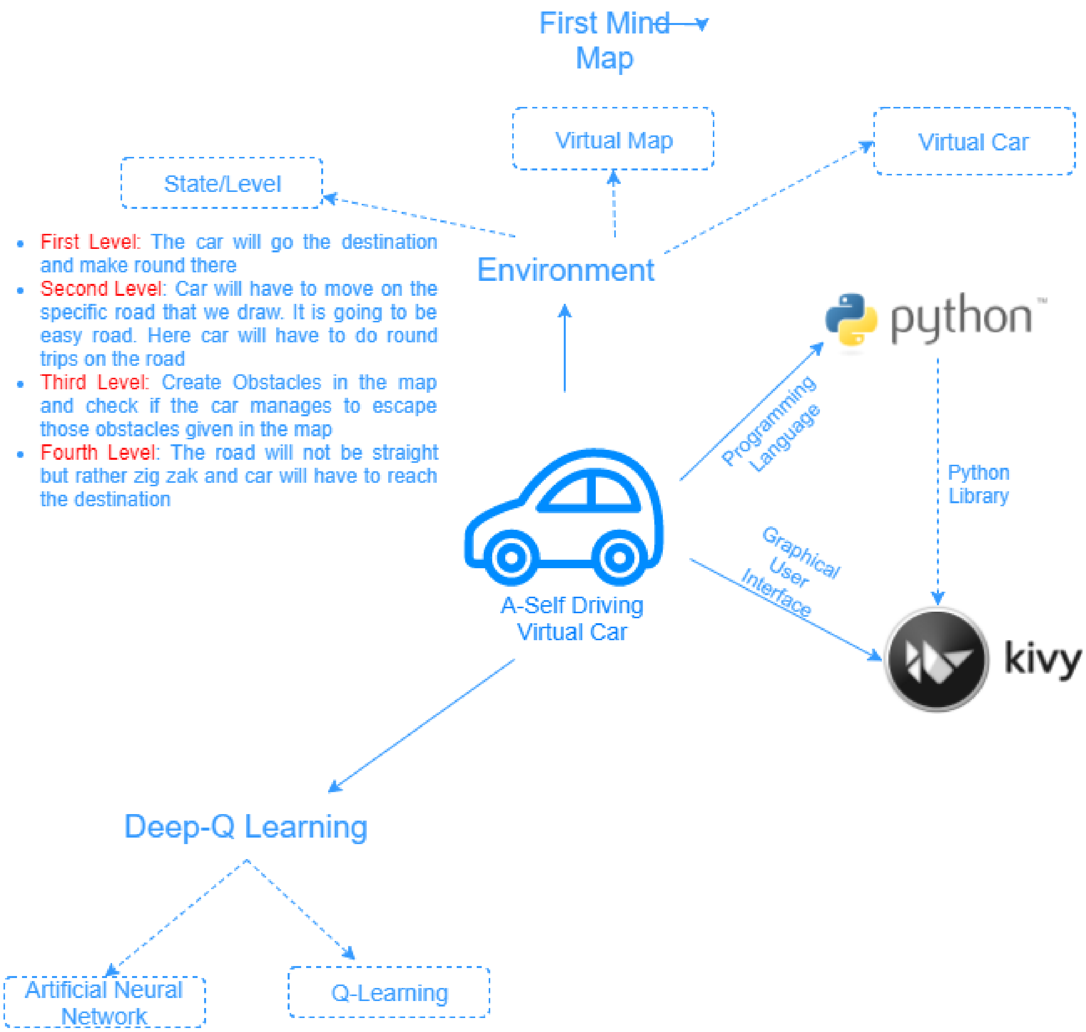

Figure 12: Week 1 Mind Map

In the mind map, the basic structure of the system is mapped. There is going to be a virtual car and a virtual environment. There are different level for checking the car capability of learning. Here, Deep-Q-Learning method is mentioned as the concept of Deep-Q-Learning is used in the system development. Likewise, Python is decided as programming language and with its library kivy for making the Graphic User Interface part.

3.3.2. Week 2 Mind Map (Deep-Q-Learning)

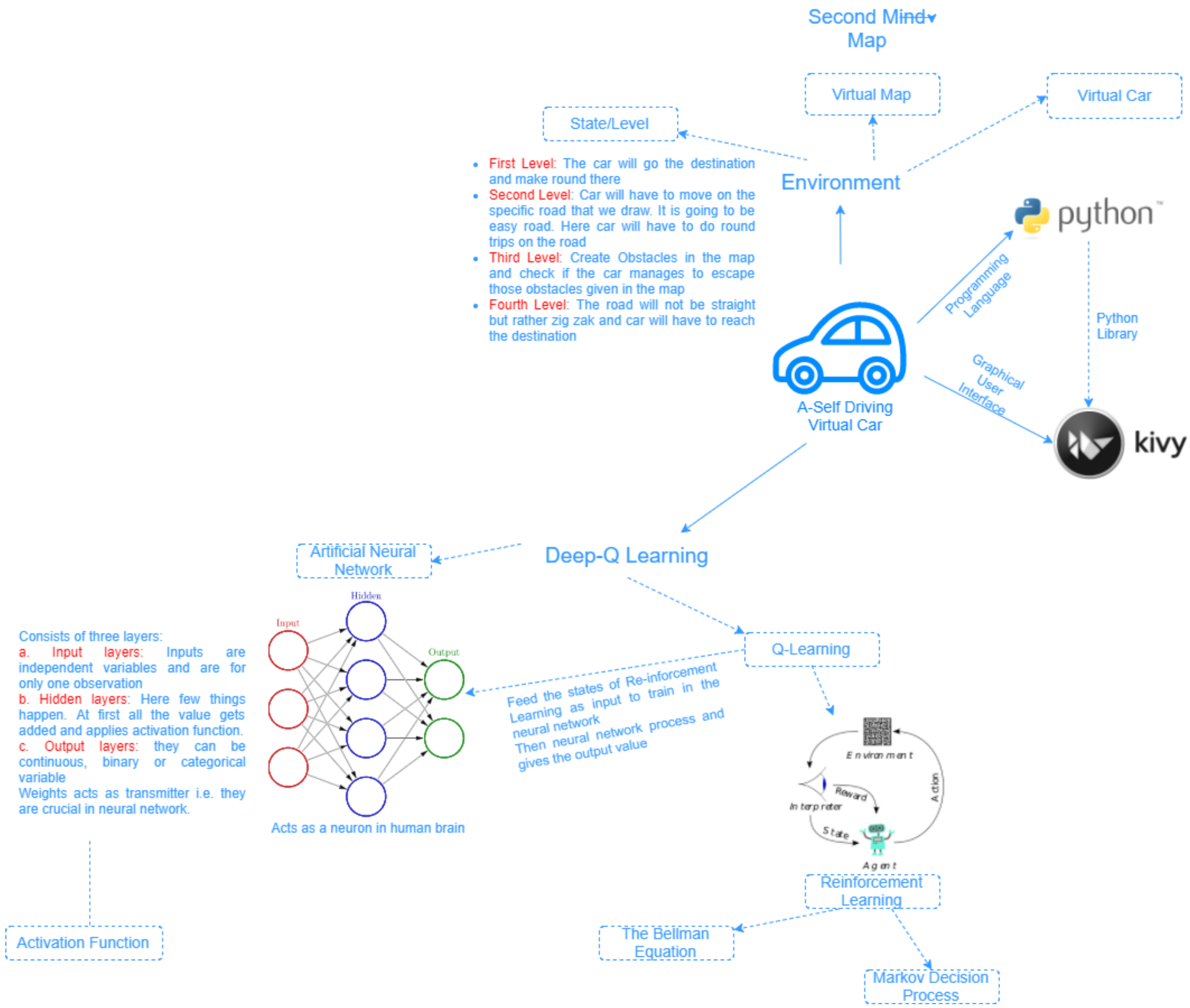

Figure 13: Week 2Mind Map

In the mind map, Deep-Q-Learning which consists of Artificial Neural Network and Q-Learning are described briefly. In artificial Neural Network there are three layers which are input, output and hidden layer. Hidden layer consists of Activation function. In Q-Learning, Reinforcement, Bellman Equation and Markov Decision Process are covered. The best Q-values from the Q-Learning is obtained with the help of Neural Network.

3.3.3. Wek 3 Mind Map (Q-Learning)

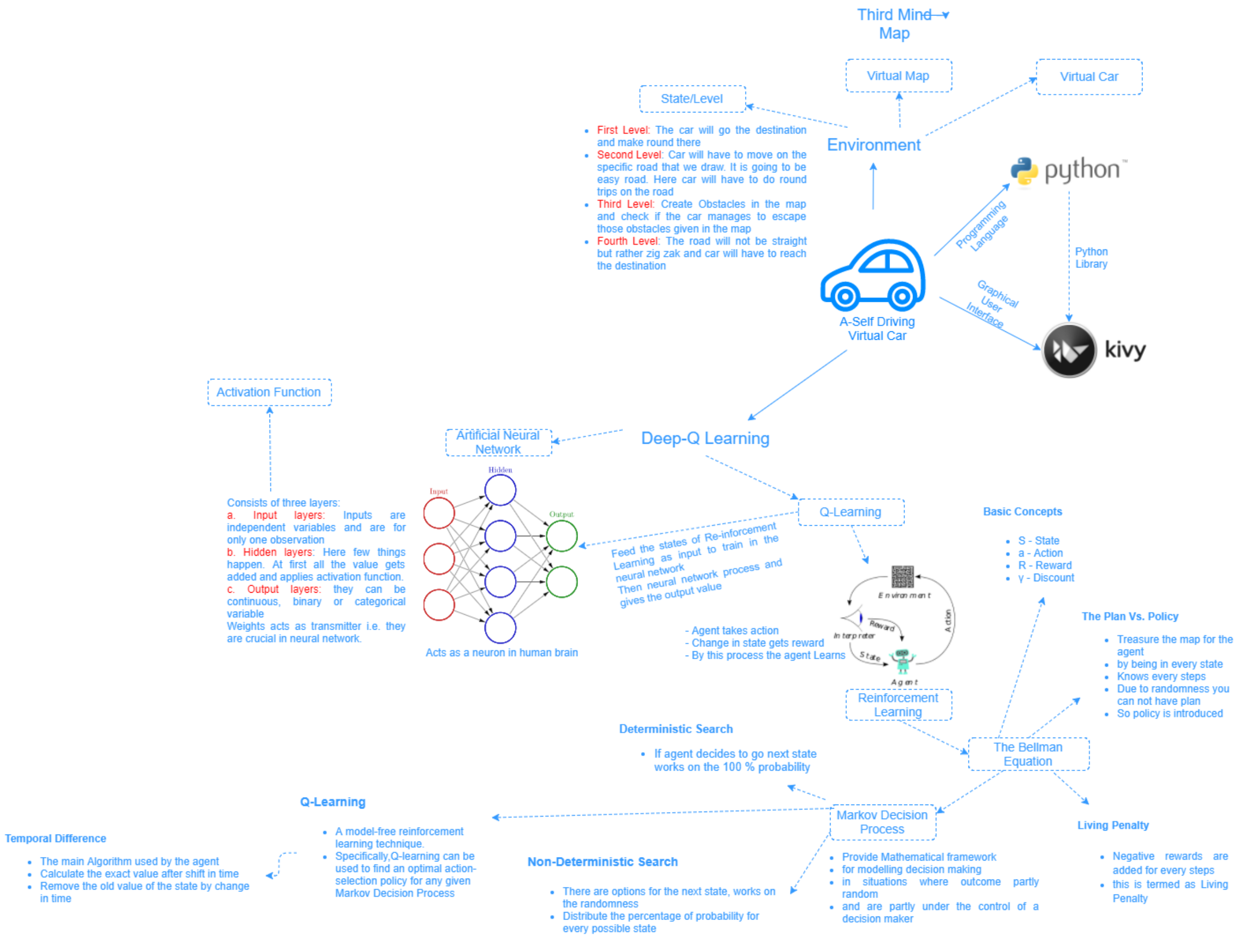

Figure 14: Week 3 Mind Map

In the mind map, the detailed information of Q-Learning is mentioned. The topics like Bellman Equation with its basic concept and policy vs plan is covered. The Markov Decision Process which has two branches i.e. Deterministic and Non-Deterministic Search is covered. Likewise, concept of Q-Learning and Temporal Difference is covered too.

3.3.4. Week 4 Mind Map (Neural Network)

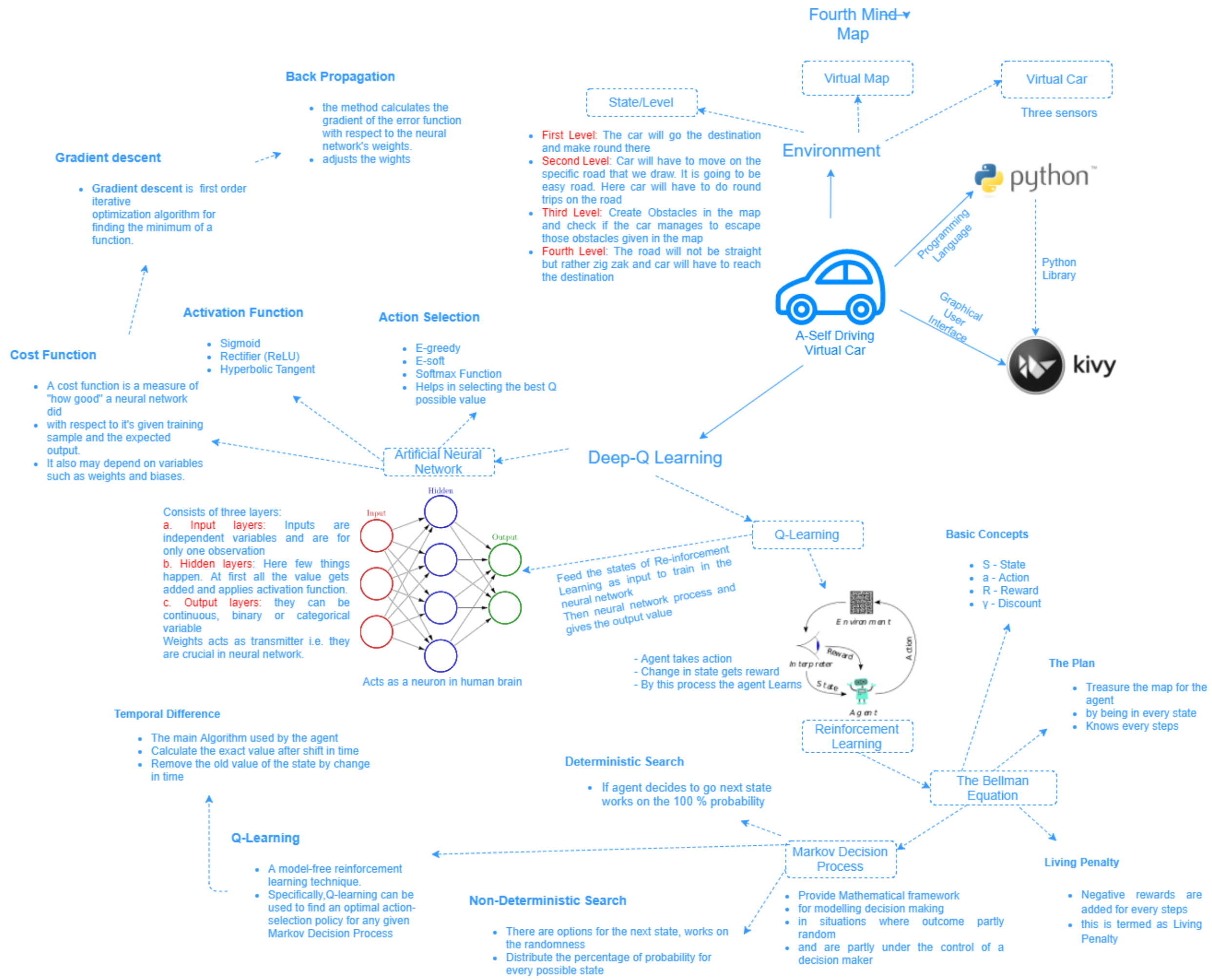

Figure 15: Week 4 Mind Map

In the mind map, the further elaboration of Artificial Neural Network is covered. The Neural Network has Activation function and terms like Cost Function, Action Selection, Gradient Descent and Back Propagation are covered.

3.3.5. Week 5 Mind Map (Final Outcome)

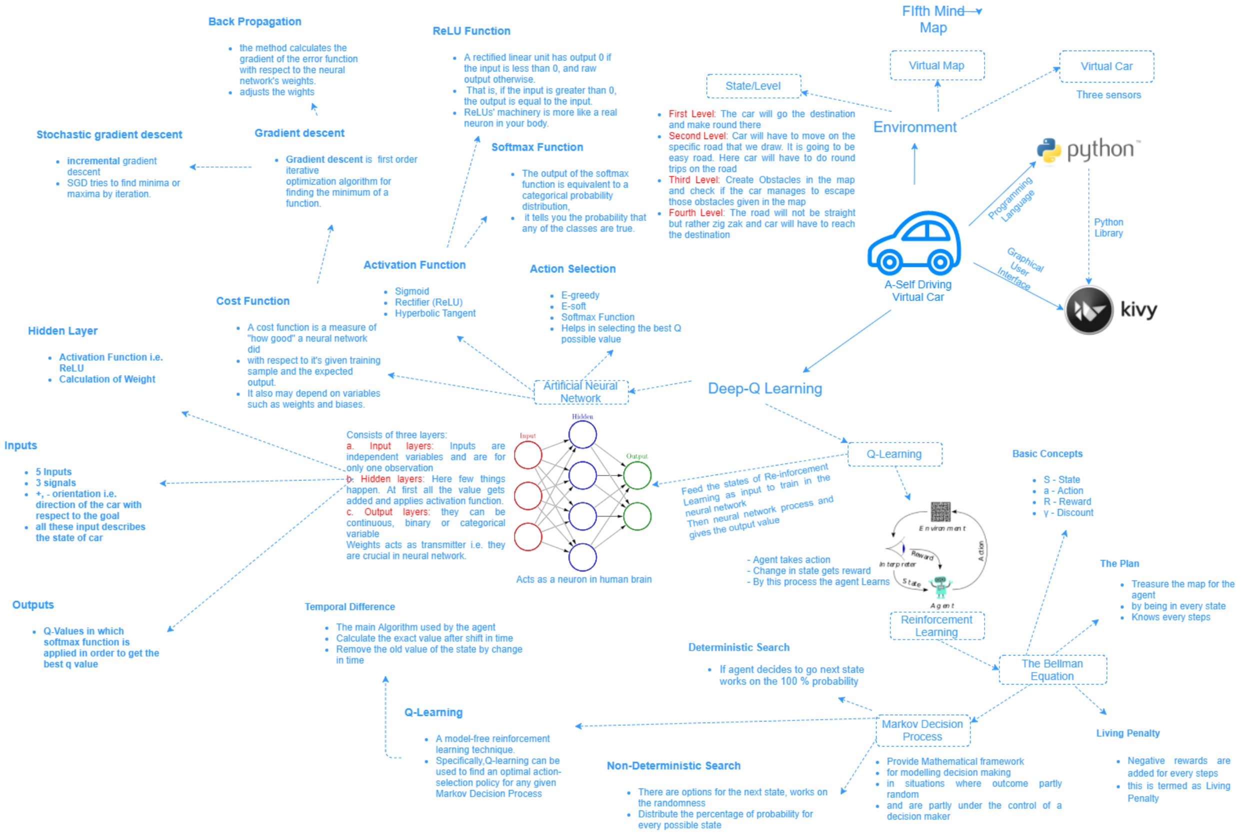

Figure 16: Week 5 Mind Map

In the mind map, the activation function, Action Selection, Cost Function that are used in the neural network of the system are mentioned. And also, the inputs, output and the Hidden layers are also mentioned. The finalized version of the mind map is completed.

3.3.6. State Transition Diagram

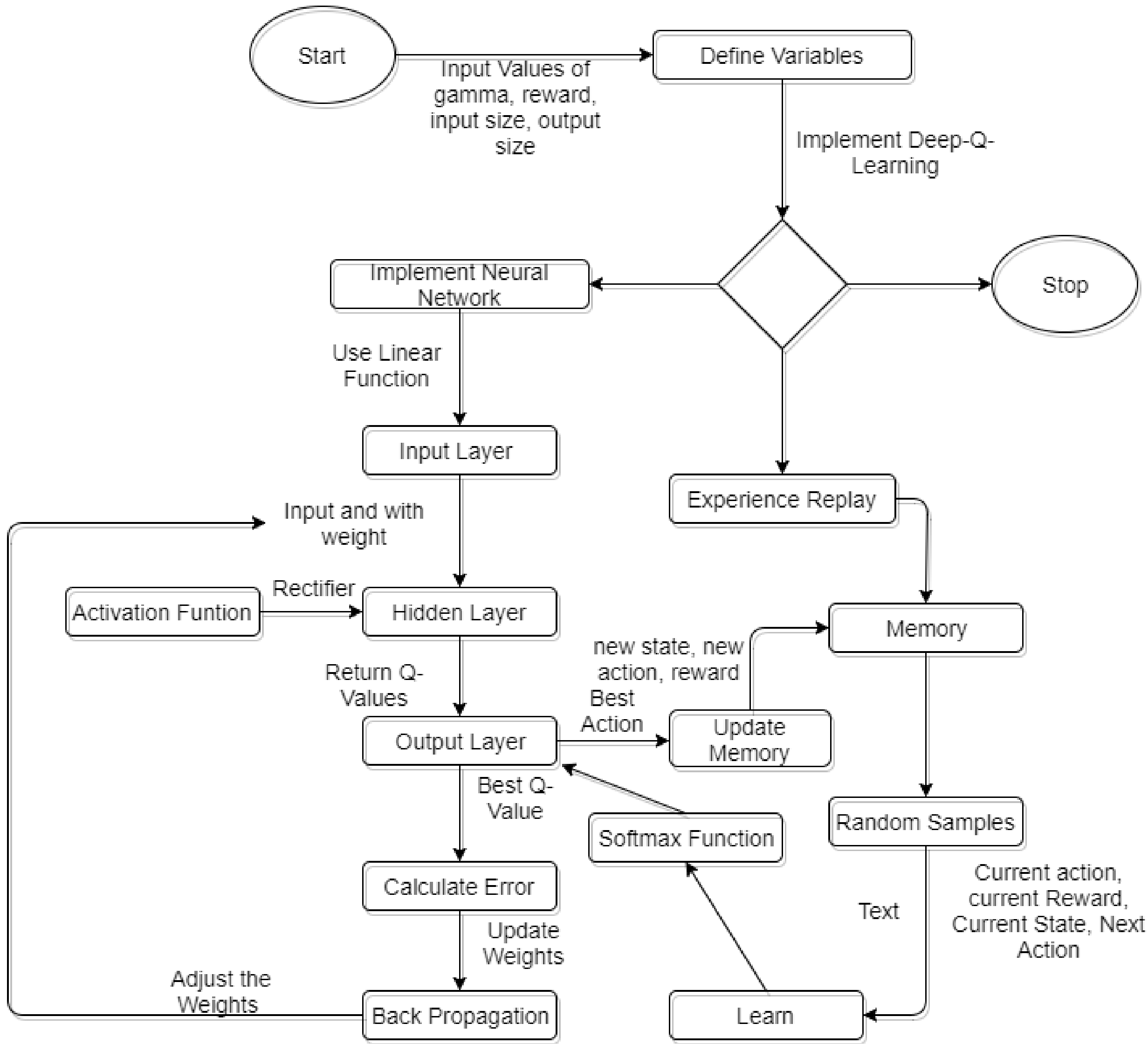

Figure 17: State Transition Diagram of the Project.

3.4. Scalability of the artifact

As, the project is only based on the modelled version of car i.e. it does not run in the real environment. This can be considered as the basic and starting point for the real world self-driving car. So, with the further research and funding the project can be used in the real-world self-driving car.

Similarly, this concept can also be used in the developing an intelligent and automated bot in different kinds of car-racing or path finding games. The concept of project can be further used in the developing automated systems like vacuum cleaner, rescue robots and many more.

3.5. Problems

Artificial Intelligence, being new topic-the coursework was very challenging. As, the coursework was based on the Deep-Q-Learning which is a huge topic in itself. To learn the concept of Deep-Q-Learning, different terms like Neural Network, Q-Learning, Experience Replay, Markov's Decision Process, and Bellman's Equation involves. To understand these Topics was very challenging. Especially, the Mathematical equations and their derivation. The concepts were clear but there is still confusions in the Mathematical equations.

In the system, the reward for the actions, learning rate for neural network and temperature in the softmax function for selecting the best action. These plays a vital role in the learning to drive safely to reach the destination. Likewise, the hidden layers, error function that is chosen also plays a vital role in the learning of the model. So, there are few problems in the system as these things takes longer time to determine and give the system with 100 percent accuracy. In the system sometimes the agent go through the wall, sometimes it gets stuck in one place for longer time. Similarly, the goal and the destination cannot be decided by the user as the goal and destination are dynamic only in the same two places.

So, the agent does not perform the 100 percent accurate actions in order to reach the destination. There are still problem in the system.

4. Conclusion

The coursework of Artificial Intelligence has been very much helpful for us. We have been able to obtain various new concepts of artificial intelligence after carrying out this coursework. Furthermore, we also learned to collaborate and successfully complete our group task.

Coursework of this module provided us knowledge about Machine Learning and use of Machine Learning. We also obtained knowledge about sub-topic Deep-Q-Learning. The topic included Neural-Network and Q-Learning concepts. We were also made familiar to various Machine Learning application.

This module also improved our analysis skill since, separate marks had been awarded for the unique topic. So, we had to think of a topic which would be different from others. In order to make our topic unique, we had to carry out research on various algorithms of AI that could be used in our application and would make our application different from the application of our other classmates. This further provided us knowledge about algorithms that could be used while developing a smart application.

Hence, this module has been fruitful for us due to the various knowledge that we could obtain while carrying out this coursework. Our group coursework also delivers all the group task that we had carried out as a team. The main concept learned were Neural Network and Q-Learning which are one of the vast and most discussed topic in Artificial Intelligence.

5. Appendix

5.1. Pseudocode

5.1.1. Pseudocode for creating map

import libraries
import kivy package for GUI

import class from ai.py by making object which is the main brain of the car

//getting AI, which we call brain and contains the neural network that represents Q-function
passing the input values by creating object of Deep Q Learning class from ai.py which are
5 state which are vector with 5 dimensions, 3 possible actions and gamma parameter value
giving values which are vectors for possible 3 actions with respect to time,

for action 0 no rotation, for action 1 rotate 20 degree, and for action 2 rotate -20 degrees
intitialise last last_reward at each state we get last reward, if car goes to the line then the reward is in negative,

intitialise the score which is a vector that contains the reward, reward on to the sliding window

define function which start the car we launh in the application
intialise the initial state of the car
intialise the direction and the speed of the car

define function update which updates everything that needs to be updated at each discrete time t when reaching a new state
initialise the global variables:
brain of the car, last reward, mean of the rewards which is score, last distance of car from the car to the goal,
x and y co-ordinate of goal and the height and width of the map

get the difference of x and y cordinates between the goal and the car
get the direction of the car with respect to the goal
get the input state vector which are three signals recieved by the three sensors plus the orientation and -orientation
//here oreintation is the oreintation of car with respect to the goal
play the action from the ai

```
append the score i.e. the mean of the last 100 rewards
move the car according to the last rotation angle
get the new distance between the car and the goal right after the car moves
update the position of the sensors right after the car moved

giving the rewards:
    if car is on the line then
        slow the speed of the car
         assign the negative reward
    else
        make car speed normal
        and assign it slightly negative reward
        if the car goes close to the goal
            assign slightly positive reward
     if car is in the left, right, bottom and upper edge of the frame
         it is not slowed down
        assign negative reward
     if the car reaches its goal
         switch the goal and update the distance from car to the new
        goal
```

5.1.2. Pseudocode for main brain of AI

```
import the libraries

create the architecture of the neural network

define class for neural network
    define function __init__ to define variables of neural network
        specify the variable input size and output size
            define the full connection which will be in between neurons of input
            layer and hidden layer then hidden layer and output layer with linera
            function

  define forward propagation function which will activate the neuron
            activate the hidden neurons with rectifier activation function
            return the output neurons which are Q-values

#Implementing Deep Q Learning
 define class for Deep-Q Learning
        define __init__ function to initialise all the variables required for the class
```

define variable gamma, input size, output size, mean reward for last 100 reward

define neural network for deep q learning model

define a variable for memory which is the object of experience replay class with capacity is equal to 100000

define a optimiser variable with good learning rate value

define last state, last action and last reward

define function that selects the best action

use softmax for best action to play on basis of probability i.e. to select the best Q-value from the Q-values

return the action

define function to train the deep-neural network

compute output of the input state

compute next output which is result of our neural network because of the target as we need the value for target

compute target

compute loss

perform the back propagation to update the weight

define function to update everything as soon as it reaches the new state

update new state with help of new signal

update the memory

play action by using select action function

if number of elements of memory > 100 then

learn from 100 random transition memory using learn function

update last action to the recently performed action

update last state to new state

update the last reward to new reward

6. References

artint.info, 2017. *Artificial Intelligence - foundations of computational agents -- 11.3.3 Q-learning.* [Online]
Available at: http://artint.info/html/ArtInt_265.html
[Accessed 08 12 2017].

Auto Express, 2018. *What is Audi's Virtual Cockpit?.* [Online]
Available at: http://www.autoexpress.co.uk/audi/102388/what-is-audis-virtual-cockpit
[Accessed 28 01 2018].

Bellman, R., 1954. Bellman Equation. In: *The theory of dynamic programming* . California : The RAND Corporation.

Bellman, R., 1954. *THE THOERY OF DYNAMIC PROGRAMMING,* California: SANTA MONICA.

cs.stanford.edu, 2017. *Neural Network.* [Online]
Available at: https://cs.stanford.edu/people/eroberts/courses/soco/projects/neural-networks/Applications/index.html
[Accessed 29 01 2018].

deloitte, 2018. *deloitte.* [Online]
Available at: https://www2.deloitte.com/us/en/pages/technology-media-and-telecommunications/articles/global-mobile-consumer-survey-us-edition.html
[Accessed 28 01 2018].

D, S., 2017. *Teaching a Neural Network to play a game using Q-learning.* [Online]
Available at: https://www.practicalai.io/teaching-a-neural-network-to-play-a-game-with-q-learning/
[Accessed 06 12 2017].

Eremenko, K., 2017. *PYTORCH WINDOWS INSTALLATION WALKTHROUGH.* [Online]
Available at: https://www.superdatascience.com/pytorch/
[Accessed 10 12 2018].

Feldman, M., 2018. *Redpixie.* [Online]
Available at: https://www.redpixie.com/blog/examples-of-machine-learning
[Accessed 27 01 2018].

Ghosh, P., 2017. *2017 Machine Learning Trends.* [Online]
Available at: http://www.dataversity.net/2017-machine-learning-trends/
[Accessed 28 01 2018].

Husain, A., 2018. *Self-Driving Cars: Past, Present and Future.* [Online]
Available at: https://www.witi.com/articles/493/Self-Driving-Cars:-Past,-Present-and-Future/
[Accessed 29 01 2018].

John, M., George, S. & Andrew, H., n.d. *Backpropagation.* [Online]
Available at: https://brilliant.org/wiki/backpropagation/
[Accessed 04 12 2017].

Juliani, A., 2016. *Medium.* [Online]
Available at: https://medium.com/emergent-future/simple-reinforcement-learning-with-tensorflow-part-0-q-learning-with-tables-and-neural-networks-d195264329d0
[Accessed 13 01 2018].

Juliani, A., 2016. *Medium.* [Online]
Available at: https://medium.com/@awjuliani/simple-reinforcement-learning-with-tensorflow-part-4-deep-q-networks-and-beyond-8438a3e2b8df
[Accessed 14 01 2018].

Khabar, G., 2018. *Glocal Khabar*. [Online]
Available at: https://glocalkhabar.com/featured/leaving-ibm-produce-artificial-intelligence-engineers-nepal/
[Accessed 29 01 2018].

Kivy, 2018. *Kivy*. [Online]
Available at: https://kivy.org/#home
[Accessed 10 01 2018].

Lin, S., 2011. Artificial Neural Network. In: X. Huang, ed. *Advanced Research on Computer Education, Simulation and Modeling.* Wuhan, China: Springer , p. 346.

Magazine, C., 2018. *Chatbots Magazine.* [Online]
Available at: https://chatbotsmagazine.com/reinforcement-learning-and-its-practical-applications-8499e60cf751
[Accessed 29 01 2018].

Maruti Techlabs, 2017. *Reinforcement Learning and Its Practical Applications.* [Online]
Available at: https://chatbotsmagazine.com/reinforcement-learning-and-its-practical-applications-8499e60cf751
[Accessed 28 01 2018].

Morgan, L., 2015. *11 Cool Ways to Use Machine Learning.* [Online]
Available at: https://www.informationweek.com/strategic-cio/executive-insights-and-innovation/11-cool-ways-to-use-machine-learning/d/d-id/1323375?
[Accessed 28 01 2018].

Nielsen, M., 2017. *How the backpropagation algorithm works.* [Online]
Available at: http://neuralnetworksanddeeplearning.com/chap2.html
[Accessed 10 12 2017].

ocw.mit.edu, n.d. *Markov Decision Processes.* [Online]
Available at: https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-825-techniques-in-artificial-intelligence-sma-5504-fall-2002/lecture-notes/Lecture20FinalPart1.pdf
[Accessed 24 12 2017].

Otterlo, M. v., 2009. *Markov Decision Processes: Concepts and Algorithms.* [Online]
Available at: http://pdfs.semanticscholar.org/968b/ab782e52faf0f7957ca0f38b9e9078454afe.pdf
[Accessed 13 01 2018].

Programiz, 2018. *Programiz.* [Online]
Available at: https://www.programiz.com/python-programming
[Accessed 14 01 2018].

Richard S. Sutton, A. G. B., 1998. *Citeseer.* [Online]
Available at: http://citeseer.ist.psu.edu/viewdoc/summary?doi=10.1.1.32.7692
[Accessed 13 01 2018].

Robot Store, 2017. *iRobot Roomba Red Intelligent vacuum 4100.* [Online]
Available at: http://www.therobotstore.com/ProductDetails.asp?ProductCode=VR-ROOMBARED
[Accessed 20 01 2018].

Rouse, M., 2017. *Whatis.com.* [Online]
Available at: http://whatis.techtarget.com/definition/machine-learning
[Accessed 14 01 2018].

Sefair, D. M., 2017. *My first experience with deep reinforcement learning.* [Online]
Available at: https://medium.com/ai-society/my-first-experience-with-deep-reinforcement-learning-1743594f0361
[Accessed 29 14 2017].

selfdrivingcars.mit.edu, n.d. *MIT 6.S094: Deep Learning for Self-Driving Cars.* [Online]
Available at: https://selfdrivingcars.mit.edu/deeptraffic/
[Accessed 02 12 2018].

StackExchange, 2016. *machine learning - A list of cost functions used in neural networks, alongside applications - Cross Validated.* [Online]
Available at: https://stats.stackexchange.com/questions/154879/a-list-of-cost-functions-used-in-neural-networks-alongside-applications
[Accessed 05 12 2017].

Sutton, R. S., 1988. *Springer.* [Online]
Available at: https://link.springer.com/article/10.1007/BF00115009
[Accessed 13 01 2018].

Tagliaferri, L., 2017. *Digitalocean.* [Online]
Available at: https://www.digitalocean.com/community/tutorials/an-introduction-to-machine-learning
[Accessed 15 01 2018].

techrepublic, 2018. *techrepublic.* [Online]
Available at: https://www.techrepublic.com/article/how-68-of-consumers-are-already-using-ai-without-knowing-it/
[Accessed 28 01 2018].

Tokic, M., 2010. *Tokic.* [Online]
Available at: http://www.tokic.com/www/tokicm/publikationen/papers/AdaptiveEpsilonGreedyExploration.pdf
[Accessed 15 01 2018].

Tom Schaul, J. Q. I. A. a. D. S., 2016. *Reinforcement learning (RL).* [Online]
Available at: https://arxiv.org/pdf/1511.05952.pdf
[Accessed 14 01 2019].

Touzet, C. F., 1999. *Neural Networks and Q-Learning for Robotics.* [Online]
[Accessed 10 12 2017].

Trymachinelearning, 2017. *trymachinelearning.* [Online]
Available at: http://trymachinelearning.com/training-method-classification/
[Accessed 14 01 2018].

Tutorialspoint, 2017. *Tutorialspoint.* [Online]
Available at: https://www.tutorialspoint.com/artificial_intelligence/artificial_intelligence_neural_networks.htm
[Accessed 14 01 2018].

Ucsusa, 2017. *Ucsusa.* [Online]
Available at: https://www.ucsusa.org/clean-vehicles/how-self-driving-cars-work#.WnHduK6Wapo
[Accessed 30 01 2018].

Walia, A. S., 2017. *Towards Data Science.* [Online]
Available at: https://towardsdatascience.com/activation-functions-and-its-types-which-is-better-a9a5310cc8f
[Accessed 14 01 2018].

White, D. J., 1993. *palgrave.* [Online]
Available at: http://www.cs.uml.edu/ecg/uploads/AIfall14/MDPApplications3.pdf
[Accessed 13 01 2017].

artint.info, 2017. *Artificial Intelligence - foundations of computational agents -- 11.3.3 Q-learning.* [Online]
Available at: http://artint.info/html/ArtInt_265.html
[Accessed 08 12 2017].

Auto Express, 2018. *What is Audi's Virtual Cockpit?.* [Online]
Available at: http://www.autoexpress.co.uk/audi/102388/what-is-audis-virtual-cockpit
[Accessed 28 01 2018].

Bellman, R., 1954. Bellman Equation. In: *The theory of dynamic programming* . California : The RAND Corporation.

Bellman, R., 1954. *THE THOERY OF DYNAMIC PROGRAMMING,* California: SANTA MONICA.

cs.stanford.edu, 2017. *Neural Network.* [Online]
Available at: https://cs.stanford.edu/people/eroberts/courses/soco/projects/neural-networks/Applications/index.html
[Accessed 29 01 2018].

deloitte, 2018. *deloitte.* [Online]
Available at: https://www2.deloitte.com/us/en/pages/technology-media-and-telecommunications/articles/global-mobile-consumer-survey-us-edition.html
[Accessed 28 01 2018].

D, S., 2017. *Teaching a Neural Network to play a game using Q-learning.* [Online]
Available at: https://www.practicalai.io/teaching-a-neural-network-to-play-a-game-with-q-learning/
[Accessed 06 12 2017].

Eremenko, K., 2017. *PYTORCH WINDOWS INSTALLATION WALKTHROUGH.* [Online]
Available at: https://www.superdatascience.com/pytorch/
[Accessed 10 12 2018].

Feldman, M., 2018. *Redpixie.* [Online]
Available at: https://www.redpixie.com/blog/examples-of-machine-learning
[Accessed 27 01 2018].

Ghosh, P., 2017. *2017 Machine Learning Trends.* [Online]
Available at: http://www.dataversity.net/2017-machine-learning-trends/
[Accessed 28 01 2018].

Husain, A., 2018. *Self-Driving Cars: Past, Present and Future.* [Online]
Available at: https://www.witi.com/articles/493/Self-Driving-Cars:-Past,-Present-and-Future/
[Accessed 29 01 2018].

John, M., George, S. & Andrew, H., n.d. *Backpropagation.* [Online]
Available at: https://brilliant.org/wiki/backpropagation/
[Accessed 04 12 2017].

Juliani, A., 2016. *Medium.* [Online]
Available at: https://medium.com/emergent-future/simple-reinforcement-learning-with-tensorflow-part-0-q-learning-with-tables-and-neural-networks-d195264329d0
[Accessed 13 01 2018].

Juliani, A., 2016. *Medium.* [Online]
Available at: https://medium.com/@awjuliani/simple-reinforcement-learning-with-tensorflow-part-4-deep-q-networks-and-beyond-8438a3e2b8df
[Accessed 14 01 2018].

Khabar, G., 2018. *Glocal Khabar.* [Online]
Available at: https://glocalkhabar.com/featured/leaving-ibm-produce-artificial-intelligence-

engineers-nepal/
[Accessed 29 01 2018].

Kivy, 2018. *Kivy*. [Online]
Available at: https://kivy.org/#home
[Accessed 10 01 2018].

Lin, S., 2011. Artificial Neural Network. In: X. Huang, ed. *Advanced Research on Computer Education, Simulation and Modeling.* Wuhan, China: Springer , p. 346.

Magazine, C., 2018. *Chatbots Magazine.* [Online]
Available at: https://chatbotsmagazine.com/reinforcement-learning-and-its-practical-applications-8499e60cf751
[Accessed 29 01 2018].

Maruti Techlabs, 2017. *Reinforcement Learning and Its Practical Applications.* [Online]
Available at: https://chatbotsmagazine.com/reinforcement-learning-and-its-practical-applications-8499e60cf751
[Accessed 28 01 2018].

Morgan, L., 2015. *11 Cool Ways to Use Machine Learning.* [Online]
Available at: https://www.informationweek.com/strategic-cio/executive-insights-and-innovation/11-cool-ways-to-use-machine-learning/d/d-id/1323375?
[Accessed 28 01 2018].

Nielsen, M., 2017. *How the backpropagation algorithm works.* [Online]
Available at: http://neuralnetworksanddeeplearning.com/chap2.html
[Accessed 10 12 2017].

ocw.mit.edu, n.d. *Markov Decision Processes.* [Online]
Available at: https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-825-techniques-in-artificial-intelligence-sma-5504-fall-2002/lecture-notes/Lecture20FinalPart1.pdf
[Accessed 24 12 2017].

Otterlo, M. v., 2009. *Markov Decision Processes: Concepts and Algorithms.* [Online]
Available at: http://pdfs.semanticscholar.org/968b/ab782e52faf0f7957ca0f38b9e9078454afe.pdf
[Accessed 13 01 2018].

Programiz, 2018. *Programiz.* [Online]
Available at: https://www.programiz.com/python-programming
[Accessed 14 01 2018].

Richard S. Sutton, A. G. B., 1998. *Citeseer.* [Online]
Available at: http://citeseer.ist.psu.edu/viewdoc/summary?doi=10.1.1.32.7692
[Accessed 13 01 2018].

Robot Store, 2017. *iRobot Roomba Red Intelligent vacuum 4100.* [Online]
Available at: http://www.therobotstore.com/ProductDetails.asp?ProductCode=VR-ROOMBARED
[Accessed 20 01 2018].

Rouse, M., 2017. *Whatis.com.* [Online]
Available at: http://whatis.techtarget.com/definition/machine-learning
[Accessed 14 01 2018].

Sefair, D. M., 2017. *My first experience with deep reinforcement learning.* [Online]
Available at: https://medium.com/ai-society/my-first-experience-with-deep-reinforcement-learning-1743594f0361
[Accessed 29 14 2017].

selfdrivingcars.mit.edu, n.d. *MIT 6.S094: Deep Learning for Self-Driving Cars.* [Online]
Available at: https://selfdrivingcars.mit.edu/deeptraffic/
[Accessed 02 12 2018].

StackExchange, 2016. *machine learning - A list of cost functions used in neural networks, alongside applications - Cross Validated.* [Online]
Available at: https://stats.stackexchange.com/questions/154879/a-list-of-cost-functions-used-in-neural-networks-alongside-applications
[Accessed 05 12 2017].

Sutton, R. S., 1988. *Springer.* [Online]
Available at: https://link.springer.com/article/10.1007/BF00115009
[Accessed 13 01 2018].

Tagliaferri, L., 2017. *Digitalocean.* [Online]
Available at: https://www.digitalocean.com/community/tutorials/an-introduction-to-machine-learning
[Accessed 15 01 2018].

techrepublic, 2018. *techrepublic.* [Online]
Available at: https://www.techrepublic.com/article/how-68-of-consumers-are-already-using-ai-without-knowing-it/
[Accessed 28 01 2018].

Tokic, M., 2010. *Tokic.* [Online]
Available at: http://www.tokic.com/www/tokicm/publikationen/papers/AdaptiveEpsilonGreedyExploration.pdf
[Accessed 15 01 2018].

Tom Schaul, J. Q. I. A. a. D. S., 2016. *Reinforcement learning (RL).* [Online]
Available at: https://arxiv.org/pdf/1511.05952.pdf
[Accessed 14 01 2019].

Touzet, C. F., 1999. *Neural Networks and Q-Learning for Robotics.* [Online]
[Accessed 10 12 2017].

Trymachinelearning, 2017. *trymachinelearning.* [Online]
Available at: http://trymachinelearning.com/training-method-classification/
[Accessed 14 01 2018].

Tutorialspoint, 2017. *Tutorialspoint.* [Online]
Available at: https://www.tutorialspoint.com/artificial_intelligence/artificial_intelligence_neural_networks.htm
[Accessed 14 01 2018].

Ucsusa, 2017. *Ucsusa.* [Online]
Available at: https://www.ucsusa.org/clean-vehicles/how-self-driving-cars-work#.WnHduK6Wapo
[Accessed 30 01 2018].

Walia, A. S., 2017. *Towards Data Science.* [Online]
Available at: https://towardsdatascience.com/activation-functions-and-its-types-which-is-better-a9a5310cc8f
[Accessed 14 01 2018].

White, D. J., 1993. *palgrave.* [Online]
Available at: http://www.cs.uml.edu/ecg/uploads/AIfall14/MDPApplications3.pdf
[Accessed 13 01 2017].