Driverless Car Simulation

AI Model Development

Time: 60 mins

Introduction

In this class, the student/s will learn to load the configuration file and use the neural network to automate the car movement to the right or left.

New Commands Introduced

● neat.config.Config	Loads the NEAT configuration file
● neat.Population(config)	Creates a population for specified configuration
● for gid, genome in generation:	Loops through each genome using genome id for each generation
● p.run(eval_fitness,10)	Runs the eval_fitness() function for 10 generations
● screen.get_at((x,y))	Returns the pixel data at (x, y) coordinate

Vocabulary

- **Neuroevolution** of Augmenting topologies(NEAT) is a method in machine learning which generates neural networks with minimal input parameters to give effective outputs.
- A genome in ML, contains all the information needed to develop and mutate.
- **T**

Learning Objectives

Student/s should be able to:

- ***Demonstrate*** how to configure a configuration file for a neural network.
- ***Recall*** how to train the model using data and explain to capture data by making the car move.
- ***Explain*** the logic of the coordinate of the car not present on track, to indicate that the car has gone off track.

.

Activities

1. **Class Narrative:** *(2 mins)*

- Brief the student/s that to simulate a driverless car, it needs a neural network which works similar to the brain of a driver.

2. **Concept Introduction Activity:** *(5 mins)*

    - Let the student/s play the explore-activity to observe the car simulation.
    - Explain the need of an optimized neural network and introduce Neuroevolution and its method NEAT.
    - Using the slides, explain that the student/s will learn:
        - to load the configuration file
        - to stop the car
        - to control the car movement

3. **Activity 1: Load the Configuration File** *(14 mins)*

    Teacher Activity: *(7 mins)*

    - Demonstrate loading a NEAT configuration file and updating its values.
    - Explain how the NEAT method works by creating a population of genomes with a neural network and forming a generation. Also explain that these genomes mutate, reproduce, etc and further are used to create more generations.
    - Explain how to run the genetic algorithm created using 10 generations.

    Student Activity: *(7 mins)*

    - Guide the student/s to configure the neural network configuration file for the Neuroevolution..

4. **Activity 2: Stop and Reposition the Car** *(12 mins)*

    Teacher Activity: *(3 mins)* .

    - Explain how to check if a car is off track by first finding the endpoints of the car and then checking if any of the points is not on track using coordinate color of the track.
    - Explain how to run the car straight and stop it as soon as it goes off track and reposition the car for the next genome. Also how to increase the genome count for each such simulation.

    Student Activity: *(9 mins)*

    - Guide the student/s to stop and reposition the car when it goes off track.

5. **Activity 3: Control the Car Movement** *(12 mins)*

Teacher Activity: *(6 mins)*

- Explain how to feedforward neural network for each genome using current genome and configuration and then use it to find net output of neural network.
- Explain how two given inputs fed to the neural network give 2 outputs which can be used to define the turning angle values.

Student Activity: *(6 mins)*

- Guide the student/s to automate the movement of the car to the right or left.

6. **Introduce the Post class project:** *(2 min)*
    - Load the NEAT configuration file and stop the bird on collision with the pipe.

7. **Test and Summarize the class learnings:** *(5 mins)*
    - Check for understanding through quizzes and summarize learning after respective missions.
    - Summarize the overall class learning towards the end of the class.
8. **Additional activities:**
    - Encourage the student/s to debug the code to make sure that the number of input matches with the config-feedforward.txt file.
    - Encourage the student/s to run the code for 3 generations where the first generation has 10 genomes and no hidden layers.

9. **State the Next Class Objective:** *(1 min)*
    - In the next class, student/s will learn to you will learn to use Blockchain technology.

U.S. Standards:

CSTA: 2-AP-11, 2-AP-12, 2-AP-13, 2-AP-14, 2-AP-19

Links Table		
Activity	**Activity Name**	**Link**
Class Presentation	Driverless Car Simulation	https://s3-whjr-curriculum-uploads.whjr.online/a2a643af-b7ad-480d-9158-63c240a5bc1d.html
Explore Activity	Driverless Car Simulation	https://github.com/Tynker-Computer-V

		ision/TNK-M10-PRO-C78-SAS-BP
Teacher Activity 1	Load the Configuration File	https://github.com/Tynker-Computer-Vision/TNK-M10-PRO-C78-TAS-BP
Teacher Reference: Teacher Activity 1 Solution	Load the Configuration File	https://github.com/Tynker-Computer-Vision/TNK-M10-PRO-C78-TAS
Student Activity 1	Load the Configuration File	https://github.com/Tynker-Computer-Vision/TNK-M10-PRO-C78-SAS-BP
Teacher Reference: Student Activity 1 Solution	Load the Configuration File	https://github.com/Tynker-Computer-Vision/TNK-M10-PRO-C78-SAS
Teacher Activity 2	Stop and Reposition the Car	https://github.com/Tynker-Computer-Vision/TNK-M10-PRO-C78-TAS-BP
Teacher Reference: Teacher Activity 2 Solution	Stop and Reposition the Car	https://github.com/Tynker-Computer-Vision/TNK-M10-PRO-C78-TAS
Student Activity 2	Stop and Reposition the Car	https://github.com/Tynker-Computer-Vision/TNK-M10-PRO-C78-SAS-BP
Teacher Reference: Student Activity 2 Solution	Stop and Reposition the Car	https://github.com/Tynker-Computer-Vision/TNK-M10-PRO-C78-SAS
Teacher Activity 3	Control the Car Movement	https://github.com/Tynker-Computer-Vision/TNK-M10-PRO-C78-TAS-BP
Teacher Reference: Teacher Activity 3 Solution	Control the Car Movement	https://github.com/Tynker-Computer-Vision/TNK-M10-PRO-C78-TAS
Student Activity 3	Control the Car Movement	https://github.com/Tynker-Computer-Vision/TNK-M10-PRO-C78-SAS-BP
Teacher Reference: Student Activity 3 Solution	Control the Car Movement	https://github.com/Tynker-Computer-Vision/TNK-M10-PRO-C78-SAS
Student's Additional Activity 1	Debug the Code	https://github.com/Tynker-Computer-Vision/TNK-M10-PRO-C78-SAS-BP
Teacher Reference: Student's Additional Activity 1 Solution	Debug the Code	https://github.com/Tynker-Computer-Vision/TNK-M10-PRO-C78-SAS
Student's Additional Activity 2	Update the Configuration	https://github.com/Tynker-Computer-Vision/TNK-M10-PRO-C78-SAS-BP
Teacher Reference: Student's Additional Activity 2 Solution	Update the Configuration	https://github.com/Tynker-Computer-Vision/TNK-M10-PRO-C78-SAS
Post Class Project	Flappy Bird	https://github.com/Tynker-Computer-Vision/TNK-M10-PRO-C78-PCP-BP
Teacher Reference: Post Class Project Solution	Flappy Bird	https://github.com/Tynker-Computer-Vision/TNK-M10-PRO-C78-PCP