Dizerție – Automatizarea decizilor agentilor intr-un mediu virtual

Cuprins

1. Introducere
   1. Scop

* Implementarea de algoritmi de ML (supervised – random forrest & decission tree and unsupervised - reinformcement learning)
  1. Obiectivele
* Automatizarea deciziilor agenților într-un mediu virtual

1. Sinteza lucrării

* Prezentarea lucrării pe scurt

1. Contribuțiile autorului

* Circuitul din Unity
* Colectarea datelor din C#
* Scriptul în Python (API)
* Modelul de ML (random forrest & decission tree)
* Diagrama se face folosind limbajul mermaid
* Automatizarea începerii jocului și repornirea în momentul în care jocul este încheiat, fie că agentul a câștigat sau a pierdut

1. Articol științific (compararea diverselor algoritmi de ML folosiți în jocuri video pentru automatizarea deciziilor agenților)
2. Concluzie

* Rezultate (comparative între user vs ML supervised vs ML unsupervised)
* Future works

Capitolul 1 – Introducere

Scopul

Scopul lucrării este de a învăța un agent șă concureze într-un mediu virtual, în cazul de față agentul fiind representant de un pilot de kart iar mediul virtual este un circuit de karting. În acest experiment se vor compara rezultatele obținute de agent în urma folosirii atât a unor algoritmi supervizați (Decision Tree și Random Forrest) și nesupervizați (Reinforcement Learning).

În vederea realizării experimentului s-au folosit mai multe intrumente digitale:

* Unity Hub pentru mediul virtual
* Anaconda – pentru realizarea API-ului care comunică cu mediul din Unity și în care se realizează partea de ML
* Visual Studio Code – editor text pentru Python si Mermeid

Limbaje de programare utilizate:

* C# - în mediul unity
* Python – pentru API și Machine Learning

Capitolul 2 – Obiectivele

Un prim obiectiv este învățarea unui agent să concureze într-un mediu virtual și să își îmbunătățească rezultatele progresiv.

Având în vedere că agentul are nevoie ca date de intrare de informații precum poziția pe harta și senzori, acesta poate fi mai apoi replicat și îmbunătățit într-un automobil autonom.

Capitolul 3 - Sinteza lucrării

Această lucrare este dezvoltată în cadrul Facultății de Electronică, Telecomunicații și Tehnologi Informaționale, parte al Universității Politehnica din Timișoara. Lucrarea reprezintă munca depusă pe parcursul a doi ani de zile 2020-2022.

Pentru partea teoretică, sau parcurs mai multe articole științifice și ale resurse de documentare care ating subiecte precum Machine Learning, Reinforcement Learning și altele.

Pentru partea practică s-a realizat un experiment prin intermediul căreia se compară mai mulți algoritmi de Machine Learning, atât supervizați cât și nesupervizați.

Capitolul 3 – Contribuțiile autorului

3.1 Circuitul din Unity

În vederea învățării agentului să piloteze kartul, se va folosii un mediu virtual deja existent în Unity care conține kartul și mai multe scenele. Contribuția autorului este să adapteze mediul și codul din Unity pentru a putea prealua datele necesare din acel mediu, dar și ca să poată sa controleze kart-ul programabil. De asemenea s-au mai adaugat și 5 senzori în fața kartului.

3.2 Colectarea datelor în C#

Din punct de vedere al colectării datelor avem mai mulți pași care au fost parcurși:

1. Adaptarea codului pentru prelucrarea datelor

Pentru învățarea agentului cum să conducă este nevoie de mai multe date de intrare și se ieșire.

Avem următoarele date de intrare (unity -> api):

* Timpul din Unity
* Poziția kart-ului pe axa X
* Poziția kart-ului pe axa Y
* Poziția kart-ului pe axa Z
* Senzorul din partea din stânga
* Senzorul din partea din stânga față
* Senzorul din partea din fată
* Senzorul din partea din dreapta față
* Senzorul din partea din dreapta
* Distanța senzorului din stânga
* Distanța senzorului din stânga față
* Distanța senzorului din față
* Distanța senzorului din dreapta față
* Distanța senzorului din dreapta
* Deplasarea în fată
* Zona

Avem următoarele date in ieșire(api -> unity):

* În față
* În stânga
* În dreapta
* În spate

1. Salvarea datelor într-un fișier csv pentru fiecare sesiune
2. Antrenarea modelului
3. Exemplificarea modelului (supervizat)