



Inteligencia Artificial para Videojuegos

Grado en Desarrollo de Videojuegos

Proyecto final

Departamento de Ingeniería del Software e Inteligencia Artificial

Facultad de Informática

Universidad Complutense de Madrid



Proyecto: Aprendizaje automático con ml agents de Unity

Fecha del enunciado: 25 de mayo de 2021

Fecha de defensa: 15 de junio de 2021

Fecha de entrega: 16 de junio de 2021

Importante: Haz la entrega en tiempo y forma, subiendo al campus virtual un fichero *IAVFinal-Apellidos.txt* sobre el proyecto final. Dentro de él incluye los datos y contribuciones de los alumnos implicados, una breve explicación del proyecto, los problemas existentes y el enlace al repositorio donde están: el fichero *README.md* con el índice de la documentación técnica, la carpeta *IAVFinal-Apellidos* con el proyecto (plugins, recursos y el código fuente), la versión ejecutable para Windows de 64bits *IAVFinal-Apellidos.exe* (con sus carpetas y ficheros acompañantes) y el video comentado con las pruebas *IAVFinal-Apellidos.mp4*.

1. Introducción

A veces puede resultar difícil crear personajes inteligentes, especialmente si el videojuego es complejo. Con el fin de crear comportamientos inteligentes, los desarrolladores tienen que escribir mucho código o usar herramientas muy especializadas.

Pero con los agentes de aprendizaje automático de Unity (los agentes ML), no hay que "programar" los comportamientos deseados. En cambio, enseñamos a los agentes inteligentes a "aprender" mediante diversas técnicas y posteriormente a entrenar para mejorar el comportamiento que estén realizando. El uso de agentes ML permite a los desarrolladores crear un juego más atractivo y una mejor experiencia de juego.

El proyecto va a tratar de realizar pequeños comportamientos usando los ml agents de unity. En general serán comportamientos sencillos, ya que parte del proyecto habrá que dedicarlo para aprender lo básico de esta herramienta de Unity. Esto servirá para tener una base sólida sobre el aprendizaje automático en Unity.

2. Planteamiento del proyecto

- El proyecto va a consistir en realizar **3 comportamientos** mediante el aprendizaje automático de Unity. Cada comportamiento se realizará en una escena diferente del proyecto, para que cada uno sea independiente del otro.
- Para hacer que los agentes “aprendan” se utilizarán dos técnicas que son el **aprendizaje por refuerzo** y el **aprendizaje por imitación**. El primero consiste en recompensar o penalizar al agente si aprende a realizar bien el comportamiento deseado y el segundo consiste en enseñar al agente como realizar un comportamiento y la IA aprende de ese comportamiento “imitándolo”.
- Todos estos comportamientos estarán debidamente documentados. En la documentación se explicará cómo se han hecho/programado los comportamientos de los agentes y también se tendrán que **hacer y explicar las pruebas** realizadas en los entrenamientos a lo largo del desarrollo. Para ayudar en la realización de las pruebas se usará la herramienta **Tensorboard** que ofrece graficas sobre diferentes aspectos en el entrenamiento de los agentes.
- Para hacer que los agentes aprendan más rápido se deberán **duplicar** varias veces **el escenario** realizado en cada escena, para así tener más agentes **entrenando** a la vez.

La entrega será realizada en tiempo y forma **[1 pto.]**, el proyecto estará bien diseñado, organizado y comentado **[1 pto.]**, y la documentación explicará con claridad cuáles fueron las técnicas implementadas **[3.5 pto.]**, las pruebas realizadas y los resultados obtenidos **[1 pto.]**.

El prototipo ejecutable será usable y funcional, permitiendo:

- Tener un comportamiento realizado mediante el aprendizaje por refuerzo. Para ello primera se preparará un pequeño escenario, no muy complejo que tenga un jugador y un objetivo al que ir, para recolectarlo. El comportamiento consistirá en hacer que el jugador vaya a tal objetivo, “enseñándole” como hacerlo con el aprendizaje por refuerzo. **[1 pto.]**
- Tener un comportamiento hecho mediante aprendizaje por imitación. Al igual que antes se preparará un pequeño escenario. El comportamiento, que será un poco más complejo que el anterior, consistirá en hacer que el agente pulse un interruptor que hará aparecer un objeto que tendrá que recoger. **[1 pto.]**
- Tener un comportamiento más realizado mediante aprendizaje por imitación. Este consistirá en realizar la IA de una de las palas de Pong mediante esta técnica. La otra pala estará controlada por el jugador. **[1.5 pto.]**

3. Restricciones y consejos

A la hora de desarrollar este proyecto es obligatorio:

- Utilizar únicamente las herramientas de Unity y opcionalmente los plugins de terceros *Bolt* o *Behavior Designer*, sin reutilizar código ajeno al que proporciona el profesor.
- Documentar claramente los algoritmos, heurísticas o cualquier “truco” utilizado.
- Diseñar y programar de la manera más limpia y elegante posible, separando la parte visual e interactiva del juego, del modelo y las técnicas de IA implementados.
- Evitar, en la medida de lo posible, el uso de recursos audiovisuales pesados o ajenos.

Pensando tanto en las pruebas como en la revisión del profesor, y también con ánimo de reutilizar el esfuerzo de desarrollo, conviene crear herramientas visuales cómodas para mostrar escenarios de ejemplo interesantes y con instrucciones de uso, etc. El manejo debe ser ágil e intuitivo para poder repetir rápidamente todas las pruebas que sean necesarias con las variaciones que hagan falta.

4. Referencias y ampliaciones

Como punto de partida para la investigación, además de la bibliografía de la asignatura, puedes utilizar las siguientes referencias. En ningún caso debes replicar el código que encuentres por ahí; asegúrate de entenderlo y verifica que funciona *exactamente* como pide este enunciado.

- Bolt, Visual Scripting
- Opsive, Behavior Designer
<https://opsive.com/assets/behavior-designer/>
- Unity, Navegación y Búsqueda de caminos
<https://docs.unity3d.com/es/2019.3/Manual/Navigation.html>
- Unity 2018 Artificial Intelligence Cookbook, Second Edition (Repositorio)
<https://github.com/PacktPublishing/Unity-2018-Artificial-Intelligence-Cookbook-Second-Edition>
- Unity Artificial Intelligence Programming, Fourth Edition (Repositorio)
<https://github.com/PacktPublishing/Unity-Artificial-Intelligence-Programming-Fourth-Edition>