



UNIVERSIDAD POLITÉCNICA DE MADRID
ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA DE SISTEMAS
INFORMÁTICOS

AGENTES INTELIGENTES EN UNITY

TRABAJO FIN DE GRADO
Grado en Ingeniería de Computadores
Curso académico 2020-2021

Autor: Mario Castro Ginés
Tutores: Jesús Mayor Márquez y Alberto Díaz Álvarez

RESUMEN

ABSTRACT

INDICE

Contenido

1. INTRODUCCION.....	6
1.1 Planteamiento del problema.....	7
1.2 Objetivos.....	8
2. ESTADO DEL ARTE	8
2.1 Motivación	9
2.2 Inteligencia Artificial.....	11
2.2.1 Inicio de la inteligencia artificial (Alan Turing)	11
2.2.2 Actualidad (IBERDROLA).....	12
2.3 Aprendizaje automático	14
2.3.1 Aprendizaje supervisado y no supervisado	14
2.3.2 Aprendizaje por refuerzo e Imitación.....	17
2.4 Agentes Inteligentes.....	19
2.4.1 Aprendizaje por refuerzo e imitación.....	21
2.5 Inteligencia artificial en los videojuegos	24
2.5.1 Inicios.....	24
2.5.2 Actualidad	25
2.5.3 Videojuegos y el aprendizaje por refuerzo.....	27
2.6 Tecnologías y herramientas utilizadas	27
2.6.1 Unity.....	28
2.6.2 MI-Agents.....	29
2.7 artículos de interés	31
3. DESARROLLO.....	33
3.x Unity	33
3.x.x Entorno	33
3.x Videojuego.....	33
3.x.x diseño y desarrollo	33
3.x. Entrenamiento	33
3.x.x Reinforcement learning.....	33
3.x.x Imitation learning	33
4. RESULTADO	34
5. CONCLUSIONES	35
5.1 IMPACTOS AMBIENTALES	35

6. LINEAS FUTURAS	36
7. BIBLIOGRAFIA.....	37

Índice de Figuras

Figura 1. Juego del escondite. [PAPER]	10
Figura 2. Alan Turing. [Computer Hoy].....	12
Figura 3. Algoritmo de Clasificación Supervisada [médium.com]	15
Figura 4. Aprendizaje no supervisado [Medium.com]	16
Figura 5. Aprendizajes.....	18
Figura 6. Flujo de acción agentes inteligentes. WIKI	20
Figura 7. Garri Kaspárov contra el ordenador Deep Blue en 1997. [Y].....	24
Figura 8. PlayStation 5. [SONY].....	26
Figura 9. ML-Agents Unity.[Unity ml agents].....	29
Figura 10. ML-Agents flow [GDC 2018].....	30
Figura 11. Carreteras de la simulacion.[articulo]	31
Figura 12. Flujo de trabajo. [ARTiculo]	32

1. INTRODUCCION

Justifica el texto

La ^Inteligencia ^Artificial ^(IA) es un campo de la informática que se ha ido popularizado más y más a lo largo de las últimas décadas. De una manera o de otra la ~~IA~~ ^{IA} está cada vez más presentes en nuestras vidas diarias, por ejemplo, todos los smartphones hoy en día utilizan e incluyen estas inteligencias para llegar a conocer mejor a los usuarios de estos dispositivos y crear experiencias más personalizadas basándose en sus intereses, rutinas etc. Hoy en día si observamos lo que aparentemente serían dos teléfonos iguales de la misma marca, el mismo modelo y configuración. Comparándolos, llegaríamos a la conclusión de que, sobre todo dentro de las aplicaciones del teléfono, hay muchas diferencias en cómo la información que se muestra al usuario es totalmente distinta dependiendo del dispositivo.

La IA es ~~estas inteligencias artificiales~~ ^{estas inteligencias artificiales} son uno de los grandes avances de la humanidad, ahora se pueden crear sistemas inteligentes capaces de aprender del entorno por ellos mismos y si la información del entorno cambia ellos también lo harán, es decir, se actualizarán. Los desarrolladores pueden crear programas simples con muy poca información sobre un problema y con ~~una buena inteligencia artificial~~ ^{un tiempo suficiente}, un sistema inteligente podrá resolver el problema por sí mismo de una manera casi perfecta y optimizada.

De hecho, esto puede ayudar a las personas en una gran medida, haciendo tareas de manera automática con mayor precisión y rapidez frente a las habilidades de un humano. La IA se usa en muchísimos campos, por ejemplo, el Machine Learning es usado para realizar diferentes tareas en industrias como la medicina, ámbitos comerciales, militares...

Aquí es donde entran en acción los agentes inteligentes y podemos ver su potencial. Estos agentes inteligentes basados en Machine Learning pueden resolver problemas de manera automática, con muchísima precisión y de una manera casi perfecta. Todo esto que consigue, lo consigue de una manera en la que un humano nunca podría imaginar, el agente aprende a observar el entorno y mediante la repetición consigue conocerlo y adaptarse a la perfección, encontrando maneras de resolver problemas que un humano ni siquiera hubiera tenido en cuenta por la falta de experiencia en el entorno. De esta manera podemos crear escenarios donde estos agentes nos ayuden a resolver las tareas que se les encomiende.

A grandes rasgos, un agente inteligente es una “entidad software”, que se basa en su propia experiencia para completar tareas, tomar decisiones y acciones a favor de satisfacer las necesidades del usuario u otros programas.

En definitiva, ~~lo que~~ ^Fin de ^Grado ~~(en adelante TFG)~~ ^(en adelante TFG) se propone crear un entorno u escenario y observar cómo estos agentes inteligentes aprenden a resolver un problema. En concreto se va a desarrollar y diseñar un simple videojuego de baloncesto donde dos jugadores de equipos diferentes competirán por tener el balón y conseguir encestar la pelota en la canasta contraria. Los agentes inteligentes entonces, serán los jugadores y tendrán que aprender desde cero como jugar y luchar contra su contrincante para que no gane el partido.

percibir
↓
procesar
↓
actuar
(racionalmente)

Los jugadores (agentes) contarán con movimientos como: moverse hacia delante-atrás o derecha-izquierda, rotar sobre si mismos para moverse con más facilidad por la pista de baloncesto, coger la pelota o bien del suelo do robándosela al oponente y lanzar la pelota para lograr encestar en la canasta del equipo contrario. Quien más puntuación tenga al final del juego ganara la partida como en un partido de baloncesto convencional.

1.1 Planteamiento del problema

En castellano el plural de los acrónimos es sin "s"

En el presente la IA son usadas de muchas maneras y para realizar diversos tipos de tareas tales como: prever y prevenir desastres climáticos y ecológicos, pandemias, crisis económicas etc. Por ejemplo, en el campo de la meteorología, gobiernos pueden prevenir desastres naturales como tsunamis, huracanes, terremotos... Y esto lo hacen solo con la información proporcionada y recolectada de satélites y diferentes tipos de sensores por todo el mundo. Otro ejemplo sería el caso de las crisis económicas, en el que estas IA obtienen los datos usando algoritmos para intentar evitar la hiperinflación y posibles caídas de la bolsa en el mercado.

Tareas complejas como calcular futuros eventos y recrearlos en escenarios simulados son unas de las grandes tareas que hoy en día más pueden ayudar a prevenir las situaciones anteriores. Con estas nuevas tecnologías se puede decir que se puede "ver en el futuro" y anticiparse a ciertos eventos, ver el problema y como se podría resolver en escenarios virtuales.

La potencia de los agentes inteligentes está aquí. Este es el porqué de su estudio, los agentes inteligentes pueden ayudar a simular entornos y dar a conocer nuevas maneras o mejores y más rápidas vías de resolver un problema, evitarlo o simplemente ver cómo afectaría si pasase algo al entorno simulando situaciones y eventos realistas.

Pero hay ciertas dificultades a la hora de entrenar a estos agentes. Hay cierta dificultad primero en diseñar y crear el entorno donde se van a situar los agentes; si el entorno no está creado correctamente y las mecánicas del sistema no están bien programadas y controladas, los agentes podrían entrar en bucles y no aprender nunca o necesitar mucho más tiempo del que deberían en aprender a completar una tarea.

La tarea mencionada previamente no es fácil, diseñar y desarrollar un escenario requiere saber determinar con precisión el objetivo que se quiere conseguir con él. Se deben tener en cuenta los límites y mecánicas que se quieren implementar en el juego o sistema, para que los agentes puedan alcanzar el objetivo final, que en el caso del TFG sería encestar el balón en la canasta del enemigo. Si el escenario y las mecánicas no están bien desarrolladas y diseñadas, los agentes podrían encontrar errores en el sistema y saltarse ciertos pasos para resolver la tarea, encontrando soluciones que nos serían realistas en la vida real.

Imagino que es "que no serían realistas en la vida real".
↳ quita esto mejor

Otra dificultad parte de la base de que los agentes inteligentes “aprenden solos”, sí, pero esto no es del todo cierto, para ello cada vez que tomen una acción que está encaminada a realizar el objetivo final. Hay que darle al agente por así decirlo una “recompensa”, esta recompensa puede ser positiva, si ha realizado bien la tarea o si ha tomado una decisión adecuada, o puede ser negativa, si al tomar una decisión va en contra de lo que tiene que aprender. Esto requiere cierta habilidad, porque si no se le dan las recompensas adecuadas en el momento adecuado puede no aprender bien la tarea o aprender otra cosa totalmente distinta.

1.2 Objetivos

Con el fin de solucionar los problemas previamente mencionados en el apartado anterior, se quiere alcanzar una o varias soluciones a esos problemas, creando varios escenarios distintos con agentes programados de maneras distintas para comprobar cómo el agente se relaciona con estos entornos.

Los objetivos que este TFG propone son los siguientes:

- Estudiar qué metodología seguir con el fin de entrenar a estos agentes inteligentes
- Diseñar el videojuego y crear el escenario 3D
- Desarrollar las mecánicas básicas de un videojuego de baloncesto de dos equipos con solo dos jugadores.
- Implementar un entorno, programando las mecánicas para que el agente o agentes aprendan a realizar la tarea de la manera más rápida y optimizada posible
- ~~Llegar a conocer~~ que cantidad de información mínima es necesaria dar a los agentes del entorno para que aprendan correctamente
- Entrenar a los agentes y estimar cuanto tiempo necesitan para aprender todas las decisiones y acciones que tienen que tomar para completar con éxito la tarea
- Estudiar que refuerzos positivos y negativos necesitan y cuando recompensarles

Determinar →

~~El objetivo, en definitiva del TFG es estudiar cómo funcionan los agentes inteligentes,~~ para ello se usará un motor gráfico de videojuegos, con el fin de crear un escenario y con ello también se estudiarán las bases del desarrollo y diseño de videojuegos. Se estudiarán el desarrollo de mecánicas y el diseño de entornos 3D y el uso de herramientas que ayuden a implementar estos agentes dentro de un motor gráfico.

Salto de página

2 ESTADO DEL ARTE

¿ ESTADO DE LA CUESTIÓN?

la IA

En este apartado se expondrán los puntos más relevantes en relación con el desarrollo del proyecto, se explicarán en profundidad conceptos de ~~inteligencia artificial~~ que estén estrechamente relacionados con los agentes inteligentes y también aquellos conceptos del desarrollo y diseño básico de videojuegos en el motor gráfico usado para crear el entorno.

2.1 Motivación

En esta breve sección se relata de donde surge la motivación que dio lugar a este proyecto fin de grado. Y se mencionan en líneas generales algunos conceptos que posteriormente se ampliarán y detallarán en las siguientes secciones de este punto estado del arte.

La motivación de este proyecto y el verdadero punto de partida, surge a raíz de la visualización de un video de datvsv[VIDEO]. En este vídeo se relata como unos agentes inteligentes son capaces de aprender a jugar al escondite, en el video además se explican en líneas generales conceptos como agentes inteligentes, aprendizaje automático... Esto hizo ~~incito~~ a la lectura del paper del cual el video trataba.

“Emergent Tool Use from Multi-Agent Interaction”[PAPER] es el título del paper el cual relata todo el desarrollo del proyecto y cómo los desarrolladores vieron potencial en el aprendizaje automático de agentes inteligentes e incluso se llegaron a sorprender al descubrir que estos agentes eran capaces de desarrollar nuevos métodos de jugar al juego del escondite. Los agentes con el tiempo fueron capaces de encontrar errores en el sistema con el objetivo de optimizar su tarea, es decir, intentar completarla con la mayor rapidez y perfección posible.

Es un paper donde se puede observar a agentes inteligentes descubriendo de manera progresiva nuevas y más complejas estrategias de llevar a cabo su objetivo mientras ellos juegan en un simple videojuego del escondite. En este entorno los agentes están diseñados con seis distintas acciones y decisiones, además de decisiones que descubrieron los desarrolladores durante el proceso que no sabían que eran posibles en el entorno. El objetivo entonces del paper es entrenar dos agentes inteligentes mediante el uso del aprendizaje automático y dar prueba de que en un futuro después de mucho entrenamiento estos son capaces de desarrollar comportamientos más complejos e inteligentes.

???

El escenario creado entonces está compuesto por dos equipos formados cada uno por agentes distintos en función del equipo al que pertenezcan, el quipo rojo (Buscador) y el azul (Ocultador). El equipo rojo tiene como objetivo buscar y encontrar en su radio de visión a los agentes del equipo azul, pero el equipo azul tiene un tiempo antes de

empezar el juego para esconderse en el entorno, en este tiempo el equipo rojo esta parado esperando a que se esconda el oponente, el equipo azul cuenta con distintas estrategias para esconderse usando objetos del escenario como, rampas, paredes y bloques.



Figura 1. Juego del escondite. [PAPER]

El verdadero potencial aquí es la habilidad con la que cuentan estos agentes inteligentes para encontrar distintas soluciones a un problema, **que un humano nunca podría haber encontrado ni siquiera imaginado**. El agente intenta completar la tarea de forma cíclica a través de repetir el juego en el mismo escenario, con lo que con tiempo adquiere un conocimiento muy profundo sobre el entorno en el que se encuentra. Basándose en experiencias de juego previas es capaz de encontrar nuevas vías y formas de resolver el problema dado, tomando ventaja de lo que el entorno puede ofrecerle con el objetivo de optimizar la realización de su tarea de la forma más rápida, directa y precisa posible.

Los agentes son capaces de encontrar errores en los sistemas, que se le pueden haber escapado a los desarrolladores y así usarlos si estos le benefician a la hora de jugar. Por ejemplo, una de las estrategias que aprendieron fue a escalar bloques haciendo uso de un error de las colisiones entre objetos y después de escalar cualquier bloque saltar y así llegar a encontrar al equipo azul que se escondía detrás de paredes.

I??

Como conclusión, tras la lectura de este paper y la visualización de varios videos relacionados con este tema se decidió desarrollar un videojuego con el fin ultimo de entrenar a unos agentes que compitiesen unos contra otros y llegar a comprender mejor estos conceptos de inteligencia artificial a través de la realización de un sencillo videojuego de baloncesto donde dos agentes aprenden a jugar.

2.2 Inteligencia Artificial

Hay mucha controversia entre los expertos para definir exactamente de manera sencilla y precisa el concepto de inteligencia artificial ya que es una rama de la informática que todavía se está desarrollando y experimentando con ella en la actualidad. Una definición podría ser: “La Inteligencia Artificial (IA) es la combinación de algoritmos planteados con el propósito de crear máquinas que presenten las mismas capacidades que el ser humano”. [IBERDROLA]

En concreto llegar a un punto común por los expertos es complicado, porque para definir la inteligencia artificial, primero se tiene que definir el concepto inteligencia humana, concepto que a su vez es muy debatido, y al existir distintos tipos de inteligencias la cosa empieza a complicarse y empiezan a surgir distintas ramas y pensamientos respecto a este tema.

Dos de los grandes expertos en este campo de la ciencia de la computación, Stuart Russell y Peter Norving diferenciaron cuatro tipos de inteligencia artificial [IBERDROLA].

No pongas a estos ladrones, cite el original.

- 1. Sistemas que piensan como humanos:** Automatizan actividades como la toma de decisiones, la resolución de problemas y el aprendizaje. Un ejemplo son las redes neuronales artificiales.
- 2. Sistemas que actúan como humanos:** Se trata de computadoras que realizan tareas de forma similar a como lo hacen las personas. Es el caso de los robots.
- 3. Sistemas que piensan racionalmente:** Intentan emular el pensamiento lógico racional de los humanos, es decir, se investiga cómo lograr que las máquinas puedan percibir, razonar y actuar en consecuencia. Los sistemas expertos se engloban en este grupo.
- 4. Sistemas que actúan racionalmente:** Idealmente son aquellos que tratan de imitar de manera racional el comportamiento humano, como los agentes inteligentes.

2.2.1 Inicio de la inteligencia artificial (Alan Turing) → *Pídenle la tesis y le echas un ojillo.*

Durante el año 1936 se inició el proceso de la inteligencia artificial moderna gracias al experto matemático Alan Turing, conocido por descifrar los códigos secretos nazis en la

Segunda Guerra Mundial. También es conocido por ser el “padre de la computación” gracias a los avances que realizó en este campo.

En ese mismo año 1936 publicó su concepto de máquina universal, que básicamente describía lo que era un algoritmo informático, y un ordenador. En 1950 formalizó el inicio de la Inteligencia Artificial con su Test de Turing, una prueba que define si una máquina es o no inteligente. Si un humano y una IA se enfrentan a las preguntas de un interrogador y ese interrogador no puede distinguir si las respuestas provienen del humano o de la IA, entonces la IA es inteligente [Computación Hoy]. En 2014, por primera vez una IA superó este Test.



Figura 2. Alan Turing. [Computación Hoy]

Unos años más tarde los investigadores y expertos empezarían a usar el término “inteligencia artificial” en sus conferencias y documentos. Una década más tarde aparecieron los primeros simuladores de redes neuronales a manos de Minsky, el cual patentó en los setenta el primer casco de realidad virtual.

Finalmente, esta disciplina tuvo su gran auge y empezó a crecer a finales del siglo 20 gracias a que los costes de ordenadores y sus componentes empezaron a abaratarse, otra de las razones fue que cada vez más se popularizaron y se extendió el uso de los ordenadores personales. Así fue como dio lugar a los primeros agentes inteligentes, chatbots ...

El momento más importante sucedió a final de siglo cuando una inteligencia artificial llegó a ganar a un experto al ajedrez. Esto causó un gran impacto y desde entonces la inteligencia artificial entró por completo en el imaginario colectivo.

2.2.2 Actualidad (IBERDROLA) → He metas

En la actualidad la IA está presente en la mayoría de los dispositivos más usados, sobre todo en los smartphones donde, por ejemplo, se puede desbloquear el teléfono con el famoso método de la detección facial. Cada vez es más común ver hasta televisiones que cuentan con la implementación de alguna inteligencia artificial que incluyen asistentes de voz, como “Siri” de Apple, “Alexa” de Amazon ...

El objetivo de estas IAs es facilitar la vida diaria de las personas en sus tareas más cotidianas, como planificar la semana, la rutina, ver que preferencias tiene una persona y enseñar solo los artículos que le interesan en páginas digitales de compra etc.

Es una tecnología esencial en todos los ámbitos y sectores, se usa y se usará cada vez con más frecuencia en ámbitos como el Transporte, educación, sanidad, cultura...

Unos ejemplos de ventajas que nos proporcionan las IAs son los siguientes [Iberdrola]:

- **Asistentes Personales Virtuales:** Chatbots que pueden sugerir a los usuarios productos, viajes, espectáculos, películas, basándose la información recogida de las búsquedas en internet y a través de redes sociales.
- **Climáticas:** Edificios inteligentes diseñados para reducir el consumo energético, vehículos submarinos no tripulados para detectar fugas en oleoductos...
- **Meteorológicos:** Sistemas que simulan entornos para prever y evitar desastres climatológicos, como huracanes, terremotos, tsunamis...
- **Sanidad:** Utilizar IAs que recojan datos para hacer a los pacientes un diagnóstico y identificar factores genéticos susceptibles de desarrollar alguna enfermedad.
- **Transporte:** La conducción autónoma podría evitar colisiones, atascos y lo más importante muchísimas muertes por accidentes.
- **Educación:** Permite saber si un estudiante está a punto de cancelar su registro, sugerir nuevos cursos o crear ofertas personalizadas para optimizar el aprendizaje.
- **Finanzas:** Las inteligencias artificiales podrían ayudar a los bancos a detectar fraude, predecir caídas de la bolsa, infracciones y aconsejar sobre que operaciones realizar.

→ Esto de dónde te lo has sacado (referencia)

El Covid-19 ha sido el culpable de que en 2020 haya habido un incremento en el uso de la inteligencia artificial. Se esperaba un aumento en 2020 del uso de estas tecnologías por parte de las empresas y otros sectores, pero nunca se llegó a imaginar que el interés por incorporar IAs subiría tanto. Han surgido aplicaciones con inteligencia incorporadas para luchar contra el Covid-19, desde aplicaciones que usan el Big Data para recoger

información sobre cómo se está expandiendo el virus y en que zonas, hasta aplicaciones de diagnóstico para ayudar a desaturar la sanidad.

Todo esto ha provocado que de aquí en adelante organismos internacionales se planteen la necesidad de regular el uso y empleo de la inteligencia artificial. Esto se ha debido en parte gracias a la gran demanda de las empresas por usar IAs para automatizar procesos y estas han ocupado una gran cantidad de puestos de trabajo, y se estima que para el 2022, crearán cerca de 133 millones de nuevos empleos para este campo. Se estima que de media generará un aumento casi del 40% en los niveles de rentabilidad empresarial antes del 2035[Talent Garden].

2.3 Aprendizaje automático

Antes de hablar de que son y cómo funcionan los agentes inteligentes, se van a aclarar ciertos conceptos sobre el aprendizaje automático.

El aprendizaje automático o “machine learning”, es una rama de la inteligencia artificial, cuyo objetivo es dotar de inteligencia a una máquina. Por ejemplo, un agente aprende cuando consigue desempeñar una tarea que ha aprendido y mejora a base de experiencias previas, es decir, va aprendiendo desde cero a realizar ciertas tareas.

De forma más concreta, los investigadores del aprendizaje de máquinas buscan algoritmos y heurísticas para convertir muestras de datos en programas de computadora, sin tener que escribir los últimos explícitamente. Los modelos o programas resultantes deben ser capaces de generalizar comportamientos e inferencias para un conjunto más amplio (potencialmente infinito) de datos. [WIKI, aprendizaje automático]

De los tipos de aprendizaje más utilizados para entrenar son el aprendizaje supervisado no supervisado, por refuerzo y por imitación. Los aprendizajes supervisados y no supervisados son la base de lo que realmente se intenta investigar en este proyecto, que forman parte de las diferencias entre el aprendizaje por refuerzo y el aprendizaje por imitación. La comprensión de los aprendizajes supervisados y no supervisados facilitara el entendimiento de estos.



2.3.1 Aprendizaje supervisado y no supervisado

De estos dos tipos de aprendizajes, el supervisado es el más usado en el “Machine Learning”. Como su propio nombre indica, este método requiere de la observación de una persona humana detrás, para comprobar si la inteligencia artificial está aprendiendo

correctamente. Se sabe de antemano el resultado del algoritmo y el programador debe guiar a la IA hacia su objetivo. Es un aprendizaje bastante limitado en el sentido de ~~que~~, para ciertos datos de entrada, el algoritmo tiene que devolver ~~si o si~~ los datos de salida esperados por el desarrollador.

Un problema bastante común en este ámbito son los algoritmos de clasificación, donde el sistema aprende a clasificar una serie de vectores utilizando una entre varias clases. Utiliza ejemplos anteriores, recogidos en un denominado conjunto de entrenamiento, para aprender si está etiquetando las categorías de una forma correcta o debería usar otro método. Este tipo de aprendizaje puede llegar a ser muy útil en problemas de investigación biológica, biología computacional y bioinformática.

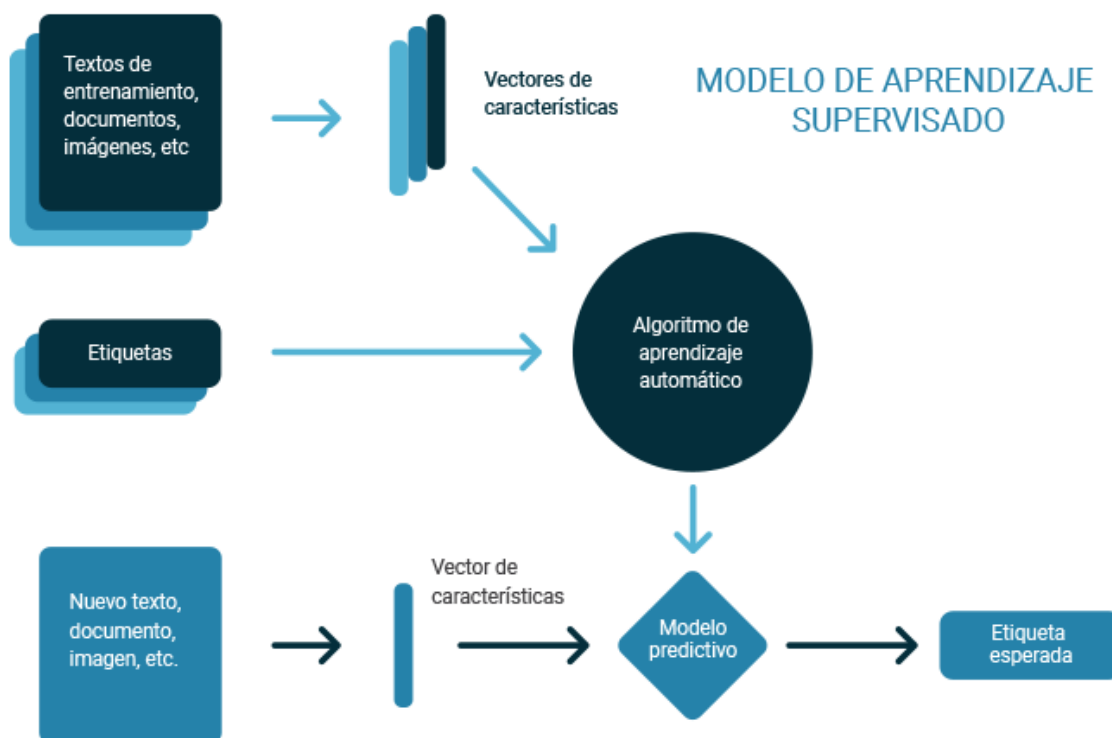
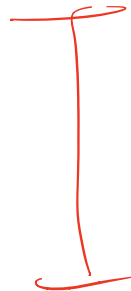


Figura 3. Algoritmo de Clasificación Supervisada [médium.com]



¡Uf! tampoco me gusta nada →

Por el contrario, se tiene el aprendizaje no supervisado, este aprendizaje se acerca más a lo que se llamaría inteligencia artificial, ya que la idea es que una máquina aprenda a identificar procesos y patrones complejos sin necesidad de que el ser humano le proporcione orientación ni supervisión a lo largo del proceso de aprendizaje [EDS ROBOTICS].

Este sistema se basa en el reconocimiento de patrones para aprender. Al contrario de lo que sucedía en el aprendizaje automático, usando este método no se puede saber la salida que se obtendrá, solo se conocen los ejemplos de entradas dados al sistema. Se podría decir que el algoritmo aprende a ciegas a resolver un problema y solo utiliza operaciones lógicas para ayudarlo. Parece en un principio una tarea titánica, pero realmente esta metodología se usa para resolver tareas muy complejas y lo hace utilizando datos de entrada y los algoritmos lógicos, evitando así el uso de datos como el conjunto de aprendizaje.

↳ Wat!?

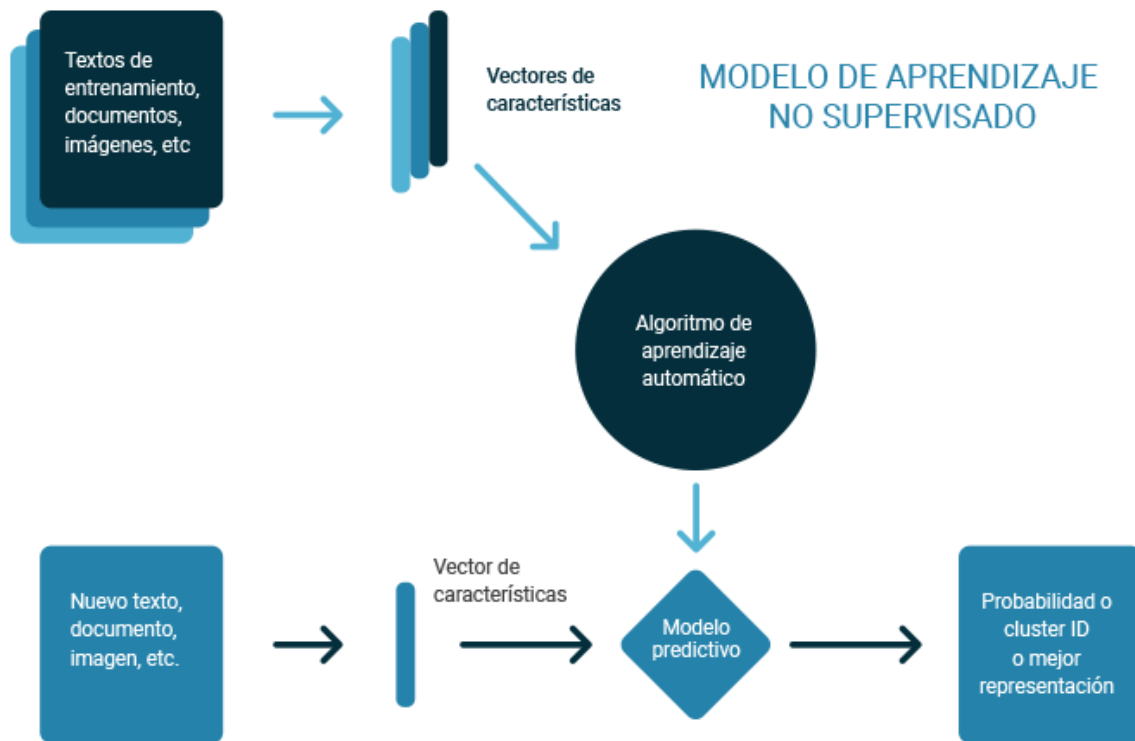


Figura 4. Aprendizaje no supervisado [Medium.com]

T

Un ejemplo de cómo funcionan estos dos tipos de aprendizajes podría ser: Se supone que se tienen tres tipos de fruta, Plátanos amarillos, Manzanas rojas y Manzanas verdes. El algoritmo que use el aprendizaje supervisado primero tendrá que aprender que es un plátano y una manzana basándose en sus distintas formas específicas, y luego tendrá que aprender a determinar qué nivel de luz corresponden los tres colores: amarillo, rojo y verde. Usando el aprendizaje no supervisado, es capaz de reconocer que tienen distintas

formas y por lo tanto son frutas distintas sin necesidad de tener que aprender específicamente si uno es una manzana u otra cosa. Después tiene que determinar los colores sin tener que saber específicamente cuales son, de esta manera todo lo que se le asemeje lo clasificará. Aunque no sabe que es lo que está clasificando habrá aprendido que la manzana y los plátanos tienen x forma que los distingue del resto y distintos colores y todo lo que le llegue con otra forma y color lo rechazará.

Como se puede observar ninguno de estos aprendizajes es mejor que el otro, simplemente son distintos y se deben usar en función al problema al que se haga frente y a la cantidad de datos que se quieran manejar.

Existe también otro tipo de aprendizaje llamado semi-supervisado que mezcla estas dos técnicas, utilizando a la vez datos etiquetados y no etiquetados, para una mejor clasificación, sobre todo se usa si la tarea a resolver es muy compleja y se quiere crear un modelo de datos predictivo.

2.3.2 Aprendizaje por refuerzo e Imitación

→ Nop! (e.g. SNARC)

De los modelos anteriores surgió una tercera opción llamada aprendizaje por refuerzo, cuyo entrenamiento se base en guiar a al agente mediante recompensas o castigos hacia su objetivo. Todo esto lo hace en un entorno en el que varios eventos suceden a la vez e interactúan entre ellos, de a la manera que el algoritmo tiene que enfrentarse a ellos y si toma las decisiones adecuadas obtendrán los refuerzos o castigos correspondientes.

Se diferencia de los modelos anteriores en su intento por maximizar el beneficio, cosa que por ejemplo el aprendizaje supervisado no hace, ya que este solo intenta reducir el error.



www.aprendemachinelearning.com

Figura 5. Aprendizajes

El sistema aprende a base de ensayo y error, se basa en su experiencia para aprender, y en un primer momento no sabe cuál es su objetivo, es decir, lo va descubriendo al igual que lo hacía el aprendizaje no supervisado. Por ejemplo, puede ser usado en robots, en brazos mecánicos en donde en vez de enseñar instrucción a instrucción a moverse, podemos dejar que haga intentos “a ciegas” e ir recompensando cuando lo hace bien [X].

Otro de los modelos que surgió del aprendizaje por refuerzo y el cual está muy unido y relacionado con él, es el aprendizaje por imitación, es bastante parecido en este caso al aprendizaje por supervisión, ya que requiere de un conjunto de entrenamiento para poder aprender. Se basa en un entorno, como el aprendizaje por refuerzo, donde aparecen ciertas mecánicas y acciones encadenadas en que el algoritmo tiene que aprender explorando el entorno. La gran diferencia es que usando esta metodología el desarrollador debe proporcionar a la IA unos ejemplos de los patrones que quiere que aprenda. El programador deberá crear ese conjunto de aprendizaje y el algoritmo deberá aprender de una forma similar al aprendizaje por refuerzo, pero partiendo de una base sólida de ejemplos y más específica.

Al igual que en los aprendizajes supervisado y no supervisado, estos dos métodos son distintos. No hay uno mejor que otro, depende del entorno, las decisiones que tenga que tomar la IA, el número de iteraciones ...

Se suele usar el aprendizaje por refuerzo para tareas directas y simples, que no requieran una gran secuencia de decisiones al algoritmo dado. Ya que al ser aleatorio e ir a ciegas puede ser una pérdida de tiempo entrenar a una inteligencia para que aprenda a desarrollar un patrón de conducta complejo. En ese caso se suele utilizar el aprendizaje por imitación, ya que se le proporciona a la maquina ejemplos de patrones de conducta complejos, y como se puede deducir este entrenamiento es innecesario si se pretende realizar una tarea sencilla y directa ya que no sería necesaria tanta información.

En resumen [video GDC]:

Aprendizaje por refuerzo	Aprendizaje por imitación
Entrenamiento mediando “recompensas”	Entrenamiento basado en demostraciones
Entrenamiento prueba y error	No hacen falta “recompensas”
Simulaciones súper aceleradas	Simulaciones en tiempo real
El agente ^{¿entiende?} aprende a completar una tarea de la manera más óptima posible	^{¿por qué?} El agente aprende a completar una tarea de la manera más humana posible

Los componentes más importantes en estos aprendizajes son los agentes inteligentes, los cuales suponen un pilar fundamental para el proyecto, y el entorno de simulación donde tienen que aprender a completar la tarea asignada.

El TFG se centrará en comparar el funcionamiento de estos dos aprendizajes y estudiar cuál de los dos es mejor para el entorno y agentes inteligentes dentro de la simulación propuesta.

2.4 Agentes Inteligentes

Un agente inteligente, es una entidad capaz de percibir su entorno, procesar tales percepciones y responder o actuar en su entorno de manera racional, es decir, de manera correcta y tendiendo a maximizar un resultado esperado[WIKI].

El flujo de acción que realiza cualquier agente inteligente es el que se muestra en la figura 4. El agente observa el entorno a través de unos sensores a los que llama Percepciones, analiza esos datos recogidos, ~~X~~ decide ~~tomar~~ una acción ~~en consecuencia~~ y actúa sobre el entorno generando nuevas observaciones y aprendiendo de ellas.

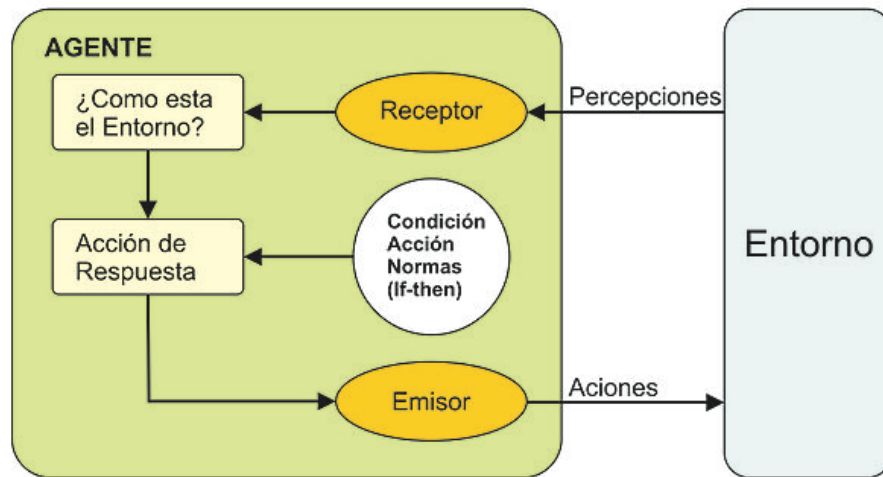


Figura 6. Flujo de acción agentes inteligentes. WIKI

Hay distintos tipos de agentes inteligentes en función de sus capacidades y grado de inteligencia [SearchEnterprise AI]:

- **Agentes reactivos:** Estos agentes funcionan en un estado actual, obviando experiencias pasadas. Las respuestas se basan en la regla de evento- condición- acción donde un usuario inicia un evento y el agente se refiere a una lista de reglas preestablecidas y resultados preprogramados.
- **Agentes reactivos basados en modelo:** Los agentes deciden que acción tomar de la misma manera que los anteriores, pero estos son capaces de analizar y comprender mejor su entorno. Un modelo de la simulación y escenario es programado dentro en un sistema interno el cual incorpora el historial del agente.
- **Agentes basados en objetivos:** Este tipo de agentes funcionan igual que los agentes reactivos basados en modelos, pero con la diferencia de que expanden un poco más su capacidad de almacenar la información del historial e incluyen información basándose en objetivos e información sobre situaciones que les serán favorables para aprender.
- **Agentes basados en utilidad:** Son similares a los agentes basados en objetivos, pero cuentan con una medición extra que califica cada posible escenario según el resultado deseado eligiendo los que mayor beneficio le reporten. Los criterios de calificación podrían ser la probabilidad de éxito o los recursos necesarios.

- **Agentes que aprenden:** Estos agentes son en los que el proyecto está centrado, tienen la habilidad de aprender gradualmente y tener en cada etapa del entrenamiento más conocimiento y comprensión de la situación y entorno. Usan elementos como recompensas o castigos para obtener “feedback” del escenario y determinar de qué manera comportarse en las siguientes etapas de entrenamiento para mejorar y llegar a perfeccionar su objetivo.

Ojo con los tiempos verbales

Las características comunes a todos estos tipos de agentes son: **su adaptación** para aprender de su experiencia, **resuelven** problemas de tiempo real y **analizan sus** errores o éxitos para seguir aprendiendo, usando una memoria interna.

Podemos encontrar en la actualidad ~~un montón~~ ^{una infinidad} de ejemplos de estos agentes inteligentes a nuestro alrededor, por ejemplo, los asistentes de voz como “Cortana”, “Alexa”, “Siri” ... Utilizan agentes para aprender, usan sensores para percibir una petición del usuario por voz y automáticamente recoger información de internet sin la ayuda del usuario. Vehículos autónomos usan estas tecnologías para aprender a evitar accidentes gracias a sensores, GPS, cámaras...

Para las empresas estos agentes pueden ser usados para aplicaciones como “data mining”, “data analytics” y servicio al cliente. Los clientes también pueden usar estos agentes para comprar precios de distintos productos y pueden notificar al usuario cuando están disponibles ciertos productos de interés o en oferta...

2.4.1 Aprendizaje por refuerzo e imitación *en agentes inteligentes*

Como se ha mencionado antes los agentes forman parte del aprendizaje por refuerzo e imitación y para entenderlo mejor se profundizará más en este apartado, en concreto el cómo este sistema funciona ya que uno de los principales estudios del proyecto.

El objetivo entonces es el siguiente: a partir de un escenario o entorno simulado, se quiere hacer que un agente inteligente aprenda de manera automática siguiendo la metodología de los aprendizajes por refuerzo, con la meta de que este aprenda a completar una tarea, o bien simple, o compuesta por una secuencia de tareas formando un patrón de conducta.

El agente inteligente empieza, como se ha mencionado previamente, desde cero sin saber cuál es su objetivo, ~~en~~ ^{bien} en un entorno en el que existen unas reglas ya establecidas. Empezará de manera aleatoria a explorar el entorno y sus posibilidades. Es trabajo del programador o desarrollador ir dejando por el camino del agente pequeñas recompensas para guiarlo a su objetivo. Como el agente se basa en su experiencia y aprendizaje previo, se establece una serie de objetivos mínimos en etapas, las cuales el agente va a tener que repetir en bucle hasta que se maximice el beneficio obtenido de las recompensas y castigos. Se considera que ha aprendido a realizar la tarea exigida cuando se estabiliza el valor en cada interacción, es decir, llega un momento en el que el agente no puede aprender más y su beneficio final siempre es el mismo en cada ciclo.

El agente inteligente va a seguir de manera periódica una serie de pasos para su aprendizaje en el siguiente orden:

1. Observación: (sin salto de línea creo que queda mejor)

Primero el agente tendrá que observar su entorno a través de unos sensores que recogen información de lo que le rodea. Una vez ha observado el entorno puede tomar una decisión basándose en los datos recogidos por los sensores. Usando el aprendizaje por refuerzo lo que el desarrollador por lo general suele indicarle que zonas del escenario son mas relevantes para observar y que el agente saque alguna conclusión por su cuenta.

2. Decisión:

Basándose en lo anterior, el agente tendrá que decidir qué acción va a realizar a continuación, basándose bien en su experiencia previa aprendida. Al principio el agente tomará decisiones completamente aleatorias, para comprobar cuales de estas le reporta mayor beneficio en la simulación. Posteriormente repetirá las decisiones con las que haya acumulado un refuerzo máximo mayor.

3. Acción:

Llamamos acción a cualquier movimiento o tarea que el agente es capaz de ejecutar, como por ejemplo moverse por el escenario. Esto lo hace en función a la decisión tomada previamente. Si el agente tiene la capacidad de moverse por el escenario hacia un objetivo, puede haber tomado la decisión de no hacerlo y quedarse quieto, eso quiere decir que su siguiente acción será no moverse. Esto puede sonar extraño, porque para una persona tomar la decisión de no moverse es no ejecutar ninguna acción ni tocar ninguna tecla o botón, pero para el agente si lo es y hay que tenerlo en cuenta.

Todo lo que puede y no puede hacer el agente lo decide el programador y este debe tener en cuenta todos estas decisiones y acciones, porque al tener demasiadas decisiones el aprendizaje puede llegar a alargarse demasiado, ya que el agente intentara ejecutar todas las acciones que pueda con el objetivo de encontrar aquellas con las que obtenga mejor resultado.

4. Refuerzos:

Esta es la etapa más relevante del aprendizaje por refuerzo y la que es decisiva a la hora de conseguir resultados. A raíz de las acciones tomadas previamente el agente podrá obtener refuerzos positivos y negativos. Por ejemplo, se le puede dar refuerzos positivos cuando haya decidido ejecutar una acción correcta que le ayude a comprender la tarea que tiene que realizar y un refuerzo positivo mayor cuando la complete al 100%. Por otro lado, puede obtener refuerzos negativos en caso de que el agente tome decisiones que le alejen de completar la tarea propuesta y no aprender por el camino que se quiera.

Resume el contenido de los puntos, que mucho de lo que dice ya lo has contado antes

Estas recompensas o castigos tienen ciertos matices, hay que recalcar que el refuerzo positivo no significa necesariamente que lo que este aprendiendo esté bien, al agente se le puede entrenar también ofreciéndole refuerzos negativos. Hay que ter en cuenta esto a la hora de establecer dónde y cuándo el agente obtendrá estos refuerzos y en qué cantidad. Si el programador por error desequilibra la balanza y el agente empieza a conseguir más refuerzos negativos que positivos, aprender a hacer todo lo contrario al objetivo que se quiere alcanzar.

Una de las grandes dificultades ^{del} ~~de~~ este aprendizaje por refuerzo reside en esta etapa. Es esencial conocer y estudiar dónde, cuando y que con cantidad de refuerzo recompensar al agente por sus acciones. Como se ha explicado previamente, este aprendizaje es muy útil para que un agente aprenda a realizar tareas simples y directas, pero para que aprenda a realizar una serie de acciones encadenadas con el fin de solventar un problema, puede suponer un gran quebradero de cabeza para el desarrollador.

Tener que definir una serie de acciones implica dar recompensar al agente varias veces con distintas cantidades de refuerzo para que llegue al final y no se quede solo al principio o por el camino. Por ejemplo, si queremos que nuestro agente no se acerque demasiado a los límites de nuestra simulación, no deberíamos darle una cantidad muy grande de refuerzo negativo porque podría aprender a quedarse quieto sin moverse y quedarse muchas etapas quieto ya que eso le haría no ser castigado. Se recuerda que el agente no percibe el entorno como nosotros y hace explícitamente lo que se le programe para aprender, por esa razón los pesos de los refuerzos son muy importantes.

El aprendizaje por imitación “resuelve” el problema de tener que equilibrar a la perfección los pesos de las recompensas que el agente obtiene. Contar con un conjunto de ejemplos que enseñan al agente a seguir un patrón de comportamiento facilita con creces su aprendizaje cuando se quiere que el agente resuelva problemas que requieran una sucesión de acciones y eventos.

Generalmente, el aprendizaje por imitación es útil cuando es más fácil para un experto demostrar el comportamiento deseado para el agente que especificar un montón de funciones de refuerzo (positivo o negativo) que generan el mismo comportamiento y puede hacer que directamente aprenda la política a seguir [SmartLabAI].

El programador entonces deberá grabar una serie de ejemplos controlando al agente inteligente y completando la tarea. Puede hacer tantos ejemplos como quiera y después entrenar al agente que parte desde un principio de esos ejemplos y ya no va “a ciegas” desde un primer momento. Las etapas de observación, decisión, acción y recompensa son las mismas que en el aprendizaje por refuerzo.

En el proyecto se implementarán los dos métodos y se compararán los resultados de cada tipo de aprendizaje con el fin de encontrar la manera más optima de entrenar a los agentes inteligentes en el videojuego de baloncesto propuesto.

2.5 Inteligencia artificial en los videojuegos

A continuación, se va a exponer brevemente ~~un poco de la historia de los videojuegos y~~ cómo la inteligencia artificial empezó a ser relevante en esta ~~área~~. También se hablará del impacto que tienen los agentes hoy en día en la actualidad del videojuego y cómo estos incorporan el aprendizaje por refuerzo mediante agentes inteligentes.

el área de los videojuegos

2.5.1 Inicios

La inteligencia artificial en los videojuegos es algo que se da por hecho muchas veces, ya que todos los videojuegos que hay en el mercado, tienen de una forma u otra implementados una inteligencia artificial. La inteligencia artificial en videojuegos empezó desde antes incluso de que se acuñase el término “videojuegos” los desarrolladores ya pensaban en intentar crear una inteligencia artificial capaz de ganar a los grandes maestros del ajedrez por medio del algoritmo “Minimax” [Y].

El algoritmo “Minimax”, no es más que un algoritmo donde la IA aprende qué siguiente movimiento realizar en función de ~~observar~~ ^{identificar} los posibles movimientos ~~puede~~ hacer y determinar cuál de ellos reporta un mayor beneficio para él. Es algo similar a lo que se ha explicado previamente, al igual que el aprendizaje por refuerzo la IA observa el entorno, toma una decisión y actúa en función a un beneficio.

En 1997 por primera vez se logró conseguir que una maquina ganase a un gran maestro del ajedrez; el superordenador de IBM conocido como Deep Blue fue capaz de derrotar en un match a seis partidas al considerado uno de los mejores jugadores de ajedrez de todos los tiempos Garri Kaspárov[Y].



Figura 7. Garri Kaspárov contra el ordenador Deep Blue en 1997. [Y]

Se inició así una tradición en la que sucesivos ordenadores dotados de inteligencia artificial han vencido a los mejores jugadores en todo tipo de juegos. El más popular de la actualidad es “Deep Mind” de Google, capaz de vencer en juegos mucho más complejos que el ajedrez (para una máquina), desde “Starcraft II” al milenario “GO” [Computación Hoy].

Gracias a estos inicios, el “Machine Learning” en cuanto a los videojuegos se ha ido desarrollando más y más evolucionando a gran velocidad en la última década y muchos videojuegos han incluido sus técnicas, como por ejemplo en juegos deportivos donde la IA compite contra el jugador, hablamos por ejemplo de juegos como “FIFA”, “Royale like” etc.

↳ Esto no es un tipo?

2.5.2 Actualidad

↳ En inicios has metido actualidad

Actualmente se emplean agentes inteligentes que reaccionan a las acciones de otros jugadores, de forma que su comportamiento es mucho más parecido al de un humano, hasta tal punto que un jugador pueda llegar a pensar que se está enfrentando a un oponente real[Z].

Hoy se utilizan los sistemas “multi-agente”, sobre todo en los videojuegos de gestión de recursos y estrategia, con el fin de coordinar comportamientos de grupo y simular diferentes personajes con distintas intenciones en la partida. En estos videojuegos de estrategia el que más destaca por sus premios y número de lanzamientos es la saga de “Civilization~~X~~”, donde el jugador tiene que gestionar un país y sus recursos y enfrentarnos a los otros países que tendrán que hacer lo mismo que nosotros para vencernos. El país que mejor gestione sus recursos y quien evoluciona más rápido gana la partida. Lo interesante del juego es que los países son agentes inteligentes que escalan a la vez que el jugador o incluso antes que él, dependiendo de la dificultad, estos son capaces de relacionarse entre ellas para hacer alianzas, romperlas, pactar treguas...

Es bastante conocido y extendido el uso de agentes inteligentes dentro de la industria de videojuegos, tanto que se estaba llegando a un punto de estancamiento donde apenas se veían un par de noticias al año relevantes hasta hace poco.

↑ referencia?

Tras el lanzamiento de las consolas de nueva generación, las grandes compañías se están planteando invertir en una mayor integración de estos agentes inteligentes a sus grandes exclusivos. En concreto Sony tras el lanzamiento de PS5 “se ha puesto manos a la obra, Sony quiere que los enemigos y aliados en sus juegos sean más inteligentes.



FIGURA 8. PLAYSTATION 5. [SONY]

Uno de los directivos explicaba así: "Aprovechando el aprendizaje ~~reforzado~~ ^{por refuerzo}, estamos desarrollando agentes de IA para juegos que pueden ser el oponente o el socio de un jugador", explicó el directivo. Esto significa que futuros títulos para PlayStation serán protagonizados por personajes que estarán mejor preparados para plantarse ante un jugador humano durante una partida, ~~ya~~ sea para rivalizar o para acompañar mientras se realizan tareas específicas. [SONY] ^y

Sony lleva tiempo detrás de la implementación de estas tecnologías en sus nuevos juegos, meses atrás de esta noticia se conoció una patente para desarrollar una IA capaz de "sustituir" al jugador humano en una partida. Esta tecnología podría monitorear movimientos y hábitos de juego de las personas para crear un personaje que sea capaz de imitarlas. De esta manera tomaría decisiones similares a las del usuario y adoptaría un estilo o mecánica de juego semejante [SONY].

Esto son buenas noticias para la industria y para la evolución de los agentes inteligentes, ya que podría empezar una carrera entre compañías por desarrollar las mejores inteligencias artificiales en sus desarrollos exclusivos.

← Me he quedado aquí!!

2.5.3 Videojuegos y el aprendizaje por refuerzo

Los videojuegos son un claro ejemplo del uso del aprendizaje por refuerzo, el porqué es bastante simple. Los videojuegos son escenarios y entornos ya programados y diseñados en el que se está simulando una situación muy concreta y cuentan por definición con una mecánicas y eventos implementados que suceden al mismo tiempo[X].

En la mayoría de los juegos podríamos decir que el jugador es el agente inteligente, el que tiene que decidir qué acciones tomar a partir de observar su entorno y en consecuencia ver qué respuesta obtienen de probar distintos movimientos y combinaciones. O bien pulsando los controles y botones del mando o las teclas del teclado junto con el ratón.

Esto que en un principio parece simple, para un agente inteligente no lo es. El agente no percibe el entorno como un humano, no tiene el conocimiento general del mundo para saber que las decisiones que toma en el entorno dan lugar a ciertas acciones las cuales conllevan una o varias consecuencias. Por ejemplo, un jugador en un juego cualquiera al ver una palanca o un botón, su intuición le va a decir que lo que debería hacer es buscar la tecla de acción en su mando o teclado que le permita pulsar el botón o tirar de la palanca. El agente por el contrario no tiene esa “intuición” y por lo tanto realizar una tarea como esa le puede suponer un gran trabajo, le puede suponer una tarea que en principio completamente aleatoria. Primero debería acercarse de manera aleatoria a la palanca y tras varios intentos tocando los distintos botones que se le ofrecen encontrar aquel que le permita tirar de ella y así aprender que esas decisiones tienen una consecuencia de acción.

Es por esto por lo que la tarea de elegir unas recompensas adecuadas, tanto negativas como positivas, para el agente es algo crucial en su aprendizaje. Hay que seguir la norma de intentar simplificar lo máximo posible estos refuerzos positivo-negativo, porque de lo contrario el agente aprendería a hacer la tarea que más refuerzos positivos le ofrezca. Volviendo al caso anterior si queremos que el agente aprenda a tirar de la palanca, eso requiere una serie de recompensas distintas ya que es una cadena de eventos. Una forma de hacerlo por ejemplo sería, dar al agente un refuerzo positivo si está lo suficientemente cerca de la palanca para tirar de ella y luego un refuerzo positivo aún mayor si toma decide la acción de tirar de la palanca.

2.6 Tecnologías y herramientas utilizadas

Se comentarán a continuación los programas y herramientas software utilizados para diseñar el escenario 3D e implementar las mecánicas del videojuego de baloncesto.

2.6.1 Unity

Como se ha mencionado previamente, uno de los pilares del proyecto es el entorno con el que los agentes deberán interactuar para aprender a jugar al videojuego. Para ello se ha de elegir un motor gráfico con el fin de diseñar el escenario y los personajes 3D, así como implementar las mecánicas del juego.

Unity es lo que se conoce como un motor de desarrollo o motor de juegos. El término motor de videojuego, “game engine”, hace referencia a un software el cual tiene una serie de rutinas de programación que permiten el diseño, la creación y el funcionamiento de un entorno interactivo; es decir, de un videojuego[master.D].

Un motor gráfico es un software diseñado para crear y desarrollar videojuegos, que además ofrece una serie de funcionalidades básicas, entre las que se encuentran un motor de renderizado para gráficos 2D y 3D, así como un motor que detecte la colisión física de objetos y la respuesta a dicha colisión [Blogthinkinbig].

Se ha elegido entonces el motor gráfico Unity por diversos motivos, para empezar, es uno de los motores mas usados por lo que posee una gran comunidad de usuarios. Esto permite tener acceso a multitud de documentación, tutoriales y foros donde se pueden consultar casi cualquier duda o problema al que se enfrente un desarrollador. Unity también ofrece cursos de iniciación y grandes cantidades de información de todas sus herramientas y funciones.

Es además uno de los motores principales que se usan para aprender a desarrollar y diseñar videojuegos, por todo lo que se ha mencionado, ya que es gratuito hasta que se desarrolle un videojuego que genere una gran cantidad de beneficios. Hasta ese momento lo pueden usar tanto estudiantes, para diseñar y desarrollar sus primeros juegos 2D y 3D, como profesionales que inicien un proyecto con poco presupuesto e ir creciendo con el tiempo.

Incluye la capacidad de crear escenarios muy realistas, esto hace que, para los desarrolladores, como es en el caso de este proyecto, se puedan desarrollar entornos y simulaciones lo más parecidas al mundo real, con el objetivo de resolver problemas reales. Por ejemplo, cuenta con un motor físico que simula a la perfección la gravedad y se puede aplicar físicas a todos los objetos del proyecto.

Otro de los factores es la capacidad de integrar sonidos animaciones, gran facilidad de implementar scripting y integrar gracias a sus librerías, inteligencia artificial.

Respecto a la programación de scripts, Unity incluye internamente la conexión de los scripts, escritos en el lenguaje C#, con el programa Visual Studio. En este IDE se pueden programar los scripts del proyecto que se este realizando y en todo momento lo que se compile en Visual Studio también lo hará dentro del motor.

Por último, una de las razones más importantes por las que se ha elegido este motor es como se comentaba antes la integración de “Machine learning”. Unity incluye una librería desarrollada por ellos mismos, llamada ML-Agents, la cual automatiza e integra funciones y procedimientos que ayudan a crear proyectos donde agentes inteligentes puedan aprender.

2.6.2 ML-Agents

La palabra ML-Agents está formada por “ML” (Machine Learning) y “Agents” (agentes) haciendo referencia a agentes inteligentes. Es decir, es la manera simplificada de referirse a los agentes inteligentes que usan aprendizaje automático para ser entrenados.

ML-Agents es un plugin desarrollado por Unity con la finalidad de ser usado por los desarrolladores de videojuegos, Unity se puso en cabeza en ser uno de los motores gratuitos que incorporan estas ventajas para entrenar inteligencias artificiales, al integrarlo como una extensión de su motor gráfico. Esto permite usar Unity no solo para crear videojuegos, si no también para entrenar agentes inteligentes en los entornos creados por la comunidad. En la siguiente figura se muestran algunos escenarios creados por Unity con la finalidad de ser usados por la comunidad para aprender a usar esta herramienta y todo lo que puede ofrecer.

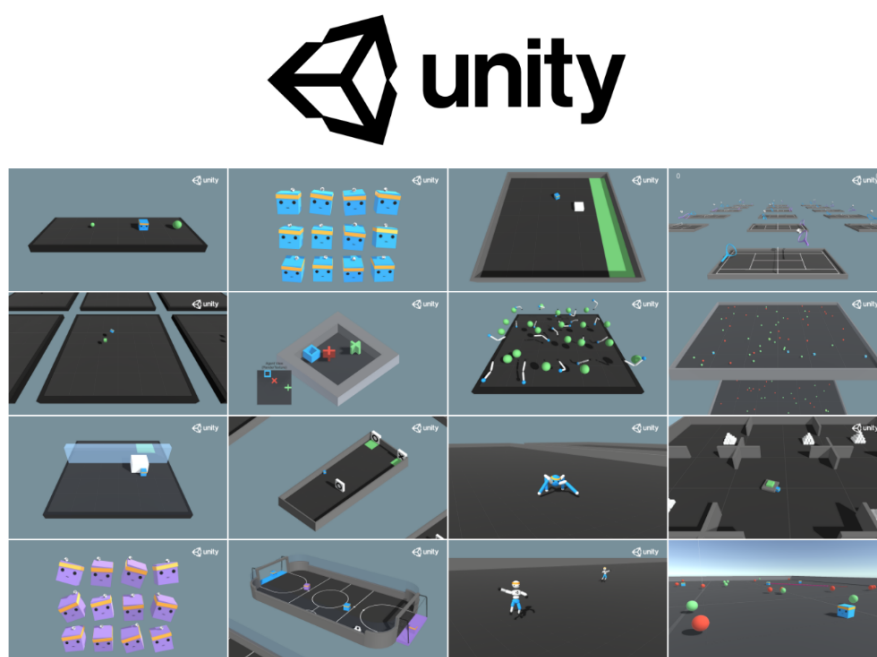


Figura 9. ML-Agents Unity.[Unity ml agents]

Este plugin de código abierto se presentó por primera vez en 2018 en la conferencia internacional para desarrolladores de videojuegos “GDC” [GDC VIDEO]. En esta presentación se explica con detalle lo más básico que hay que saber sobre este kit de herramientas. Y desde entonces han ido actualizándolo e incorporando mejoras a lo largo de los años, pasando por la versión 1.0 y hasta llegar a la actualidad con la 2.0 incorporado la capacidad de crear escenarios multi-agentes con comportamientos de cooperación entre ellos en un mismo entorno.

Esta herramienta cuenta con los dos métodos de aprendizaje vistos anteriormente, aprendizaje por refuerzo y aprendizaje por imitación. Se usarán estos dos métodos para conseguir que los agentes consigan aprender a jugar al juego y ver que aprendizaje ha dado mejores resultados.

ML-Agents implementa un flujo de datos que se conecta con una API en Python para el entrenamiento de los agentes, el cual se muestra en la siguiente figura.

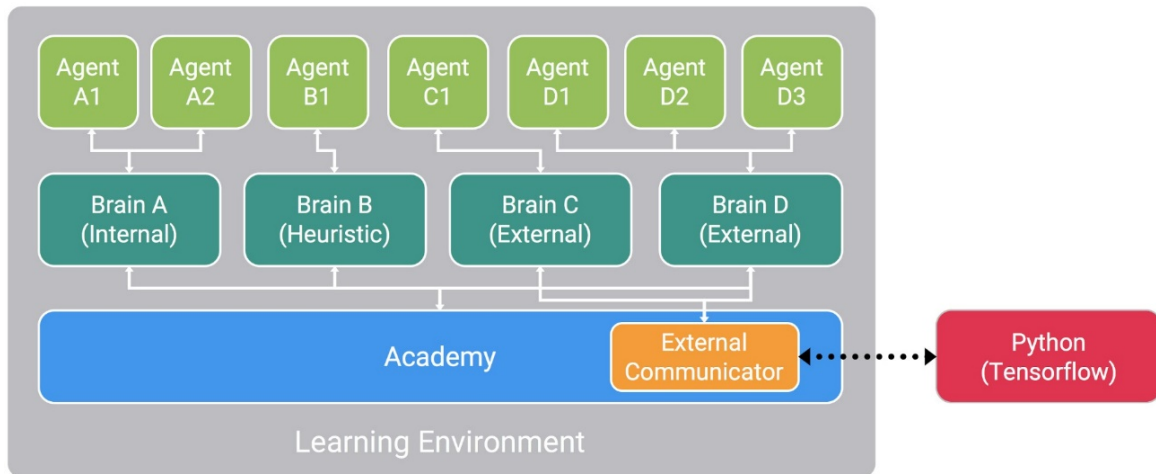


Figura 10. ML-Agents flow [GDC 2018]

Siguiendo la figura anterior, se pueden observar tres tipos de objetos principales en cualquier entorno de aprendizaje [blog unity]:

- **Agent (Agente)-** Cada agente tiene un conjunto de estados y observaciones único, toma decisiones individuales e independientes al entorno al igual que recibe refuerzos positivo-negativos también independientes al entorno. Las acciones de cada agente las decide el cerebro al que están conectados.
- **Brain (Cerebro)-** Cada cerebro define un estado específico y un espacio de acción, y es el responsable de decidir cada acción va a tomar el agente. Hay cuatro modelos de cerebros:
 - **External:** En este tipo de cerebro externo las decisiones de que acciones tomar se toman a través de la librería de elección a través de la comunicación con la API de Python.
 - **Internal:** Las acciones y decisiones son tomadas basándose en un modelo ya entrenado.
 - **Player:** Las decisiones de acción se toman utilizando la información del jugador.
 - **Heuristic:** Las acciones y decisiones se toman utilizando un comportamiento codificado a mano.

- **Academy-** Este objeto junto a un escenario contiene todos los cerebros del entorno de aprendizaje. Cada entorno contiene un solo objeto de este tipo, que define atributos del entorno como, la velocidad de entrenamiento, cuantas etapas tiene cada agente para entrenar, cuanto duran estas etapas...

Los cerebros configurados como externos se comunican con el “External Communicator” y este transmite todos los estados y observaciones del agente a la API de Python para procesar estos datos.

2.7 artículos de interés

Para acabar esta sección se expondrán otros artículos y estudios de interés que están estrechamente relacionados con este proyecto de los cuales destacan los siguientes.

A lo largo del proyecto solo se han mostrado y expuesto ejemplos de agentes inteligentes usados en escenarios y entornos relacionados con videojuegos, ya que es el objetivo del proyecto, pero se pueden usar agentes inteligentes con otros fines como veremos a continuación en dos artículos.

El primero de ellos se llama “Smart Driving Agent base on Deep Reinforcement Learning”[ARTICULO]. Este artículo trata de simular un sistema de conducción autónoma usando agentes y aprendizaje por refuerzo. El objetivo principal de este artículo es desarrollar e simular un entorno digital para trabajar con el en futuras investigaciones y avances en el campo de la conducción autónoma de vehículos auto dirigidos. Este desarrollo al igual que este proyecto usa el motor grafico Unity y una API específica que conecta la simulación con Python.

La investigación aprendió con éxito a conducir sin salirse de su carril en una carreta, las carreteras de las simulaciones son las que se muestran en la figura a continuación. El agente fue entrenado en el entorno de simulación y aprendió a conducir de manera automática siguiendo la ruta establecida sin chocarse con ningún obstáculo ni salirse del carril.

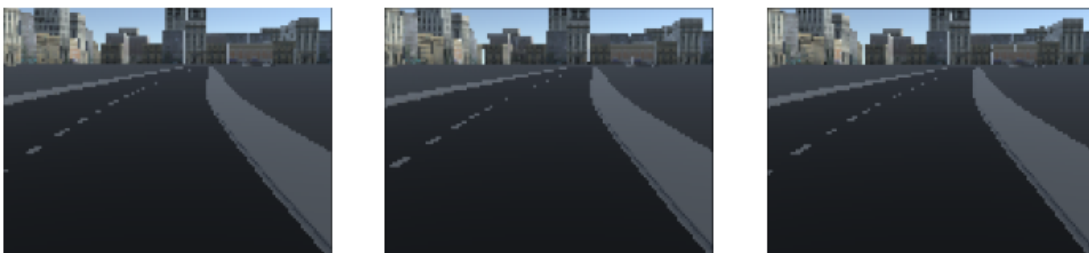


Figura 11. Carreteras de la simulacion.[articulo]

El segundo de estos artículos se llama “Reinforcement learning for architectural design” [Articulo]. En este paper se discute el potencial del aprendizaje por refuerzo para diseñar, implementar y construir edificios y cualquier estructura arquitectónica.

Desarrollaron una herramienta que promueve un flujo de trabajo de diseño y construcción basado en materiales para el diseño arquitectónico, como se muestra en la figura de a continuación. Estos agentes son la prueba de que en un futuro la inteligencia artificial podría utilizarse para saber dónde y como construir nuevos edificios, ciudades y optimar los recursos para ello.

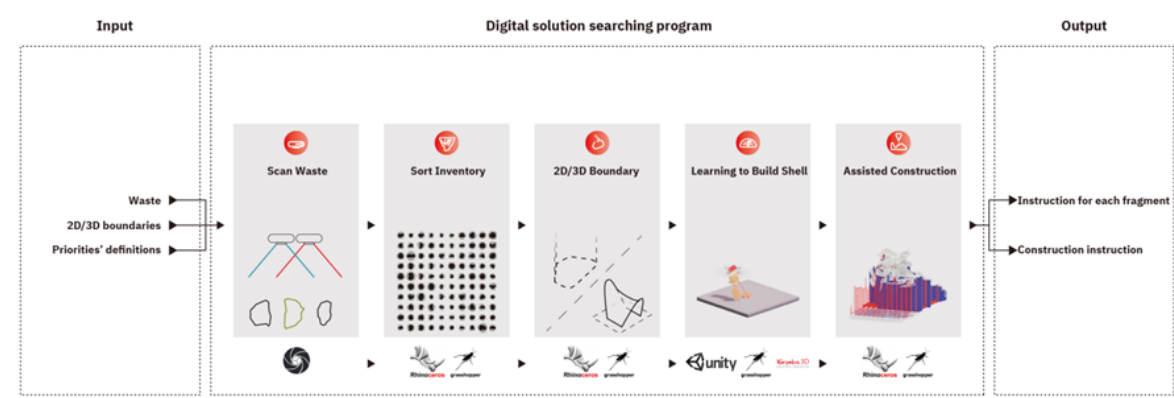


Figura 12. Flujo de trabajo. [ARTículo]

3. DESARROLLO

3.x Unity

3.x.x Entorno

3.x Videojuego

3.x.x diseño y desarrollo

3.x. Entrenamiento

3.x.x Reinforcement learning

3.x.x Imitation learning

Hacer diagramas de flujo para explicar cómo los scripts están conectados

4. RESULTADO

5. CONCLUSIONES

5.1 IMPACTOS AMBIENTALES

6. LINEAS FUTURAS

7. BIBLIOGRAFIA

- [X] Na8. (24 de Diciembre de 2020). *www.aprendemachinelearning.com*. Obtenido de <https://www.aprendemachinelearning.com/aprendizaje-por-refuerzo/>
- [1] Erard, G. (26 de Mayo de 2021). *hipertextual.com*. Obtenido de <https://hipertextual.com/2021/05/sony-inteligencia-artificial-personajes-playstation>