

ANTEPROYECTO DEL TRABAJO DE FIN DE GRADO

| | | | | | |
|--|---|--------------------------|------------------------------------|--|--------------------------|
| Alumno/a | Antonio David Ponce Martínez | | | | |
| Titulación: | Grado en Ingeniería Informática | | | | |
| Tutor/es: | Ezequiel López Rubio José Jesús de Benito Picazo | | | | |
| Título | Desarrollo de jugadores automáticos mediante aprendizaje profundo por refuerzo para videojuegos | | | | |
| Subtítulo <i>(solo si en grupo)</i> | | | | | |
| Título en inglés | Development of automated players by deep reinforcement learning for videogames | | | | |
| Subtítulo en inglés <i>(solo si en grupo)</i> | | | | | |
| Trabajo en grupo: | <input checked="" type="checkbox"/> Sí | <input type="checkbox"/> | <input type="checkbox"/> No | <input checked="" type="checkbox"/> X | <input type="checkbox"/> |
| Otros integrantes del grupo: | | | | | |

INTRODUCCIÓN

Contextualización del problema a resolver. Describir claramente de dónde surge la necesidad de este TFG y el dominio de aplicación. En caso de que el TFG se base en trabajos previos, debe aclararse cuáles son las aportaciones del TFG.

Recientes estudios han mostrado como es posible crear agentes capaces de realizar tareas complejas como la de jugar a un videojuego únicamente utilizando como entrada al problema los píxeles de la pantalla de juego, llegando incluso a alcanzar un nivel sobrehumano en algunos de ellos [1][2]. Este logro es especialmente llamativo, ya que el agente tiene acceso a los mismos datos que un jugador humano. En el aprendizaje por refuerzo el agente aprende interactuando con su entorno sin indicarle cómo ha de hacerlo, aprendiendo a elegir las acciones para maximizar la recompensa recibida en cada situación mediante ensayo y error, incluso cuando esta recompensa no es obvia desde un principio. Aprende a resolver los problemas en vez de aprender cómo son las soluciones [3].

OBJETIVOS

Descripción detallada de en qué consistirá el TFG. En caso de que el objeto principal del TFG sea el desarrollo de software, además de los objetivos generales deben describirse sus funcionalidades a alto nivel.

Se pretende desarrollar un agente mediante aprendizaje por refuerzo profundo, el cual deberá realizar ciertas tareas definidas en el espacio de un videojuego. El trabajo en sí estará dividido en dos secciones:

Desarrollo del agente

Para implementar dicho comportamiento se hará uso de algoritmos de aprendizaje profundo por refuerzo. Las redes neuronales son la unidad central de estos algoritmos, las cuales intentarán modelar o bien la

calidad de las acciones realizadas o cual es la probabilidad de realizar una acción dado un estado concreto de nuestro entorno virtual. Se implementarán algunos de estos algoritmos y analizarán los distintos resultados obtenidos.

La implementación de este trabajo se realizará con Python, utilizando las librerías Keras junto a Tensorflow para modelar la red neuronal. Dichas redes serán usadas en algoritmos como Deep Q-Learning o Policy gradients.

Integración del entorno con el agente

Para este caso, el entorno virtual sería un videojuego retro, con el cual podremos interactuar gracias a librerías como Gym Retro [4] o ViZDoom [5], las cuales tienen soporte para Python que usaremos como API para obtener datos como los últimos fotogramas de juego, puntuación actual o cualquier otro dato que veamos considerable. Será necesario utilizar estas librerías para obtener los datos necesarios para entrenar a nuestro agente.

Las tareas a realizar por el agente consistirán en maximizar la recompensa obtenida en diversos escenarios, lo cual podría abarcar desde que el agente tenga que sobrevivir el máximo tiempo posible, tenga que recoger cierto tipo de coleccionables o obtener una alta puntuación en el episodio.

ENTREGABLES

Listado de resultados que generará el TFG (aplicaciones, estudios, manuales, etc.)

Código para entrenar al agente.

Memoria sobre el desarrollo del proyecto.

Material visual sobre los resultados del agente.

MÉTODOS Y FASES DE TRABAJO

METODOLOGÍA:

Descripción de la metodología empleada en el desarrollo del TFG. Especificar cómo se va a desarrollar. Concretar si se trata de alguna metodología existente y, en caso contrario, describir y justificar adecuadamente los métodos que se aplicarán.

Se definirán ciertas metas a completar antes de una fecha límite (milestones). Tras alcanzar dichas metas, se discutirán los avances del trabajo para ver qué podría mejorarse y qué problemas se han encontrado en el transcurso del desarrollo.

Estas metas tendrán una fecha límite de entre una y dos semanas para fomentar avances en el desarrollo del agente y poder combinarse bien con el horario universitario.

FASES DE TRABAJO:

Enumeración y breve descripción de las fases de trabajo en las que consistirá el TFG.

- 1ª fase:** Documentación sobre aprendizaje por refuerzo profundo, algoritmos en su aplicación a videojuegos y los algoritmos sobre los que se basan (Q-Learning).
- 2ª fase:** Investigación sobre la implementación de las redes neuronales profundas en Keras.
- 3ª fase:** Diseño e implementación de la red neuronal.
- 4ª fase:** Integración de la recolección de datos del entorno virtual mediante nuestras APIs.
- 5ª fase:** Fase de entrenamiento y afinamiento del agente.
- 6ª fase:** Fase de recopilación de resultados. Análisis de los mismos y últimos retoques al agente.
- 7ª fase:** Desarrollo de la memoria con los resultados y conclusiones finales.

TEMPORIZACIÓN:

La siguiente tabla deberá contener una fila por cada una de las fases enumeradas en la sección anterior. En caso de tratarse de un trabajo en grupo, se añadirá una columna HORAS por cada miembro del equipo. Debe especificarse claramente el número de horas dedicado por cada alumno/a y la suma de horas individual deberá ser también de 296.

| FASE | HORAS |
|---------|------------------|
| | Nombre Apellidos |
| PRIMERA | 50 |
| SEGUNDA | 35 |
| TERCERA | 35 |
| CUARTA | 40 |
| QUINTA | 40 |
| SEXTA | 60 |
| SÉPTIMA | 36 |
| | 296 |

TECNOLOGÍAS EMPLEADAS:

Enumeración de las tecnologías utilizadas (lenguajes de programación, frameworks, sistemas gestores de bases de datos, etc.) en el desarrollo del TFG.

Python

Keras

TensorFlow

ViZDoom

Gym Retro

RECURSOS SOFTWARE Y HARDWARE:

Listado de dispositivos (placas de desarrollo, microcontroladores, procesadores, sensores, robots, etc.) o software (IDE, editores, etc.) empleados en el desarrollo del TFG.

PC compuesto por Intel i5-8300H, nVidia GeForce GTX-1060 y 8GB RAM DDR4

Jupyter Notebook

Editor de LaTeX LYX

Listado de referencias (libros, páginas web, etc.)

[1] Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., & Riedmiller, M. (2013). Playing atari with deep reinforcement learning. arXiv preprint arXiv:1312.5602.

[2] Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., Hubert, T., Baker, L., Lai, M., Bolton, A. and Chen, Y., 2017. Mastering the game of Go without human knowledge. Nature, 550(7676), p.354.

[3] <https://thenewstack.io/reinforcement-learning-ready-real-world/>

[4] <https://github.com/openai/retro>

[5] <http://vizdoom.cs.put.edu.pl/>