

Trabajo de Investigación

¿Puede una inteligencia artificial llegar a tener consciencia?

Alumno: Pablo Smith Necpas Alba

Tutor: Moisès Bobadilla Margenat

2.º de Bachillerato D

La Salle Congrés

Barcelona, Octubre 2022

ABSTRACT:

El següent treball té com a propòsit investigar la consciència des d'un punt de vista biològic, filosòfic i computacional, per tal d'unificar tot el coneixement dels nostres dies per poder trobar una conclusió sobre què és el que ens fa ser (o no). Així mateix, s'investiga si una intel·ligència artificial (IA) és capaç de ser-ne conscient, o simular-ho.

Tot i que el tema de la consciència, del propi ésser, ha acompanyat l'ésser humà des d'antany, avui dia torna a sorgir gràcies als grans avenços en intel·ligència artificial, i aprenentatge automàtic, en les darreres dècades, que ens fan dubtar sobre què és la consciència mateixa. És un robot conscient? Si ho és, té el mateix nivell de consciència que nosaltres? Si no ho és, ho pot arribar a ser? Hi ha nivells de consciència?

Dos camps en puntes distants del coneixement humà, però la unió dels quals dóna fruit als millors avenços tecnològics. S'exploren temes com Crick i la poca coneguda faceta; les teories d'Edelman, que inclouen sorgiment de la consciència i les emocions primàries; els problemes de la consciència, com el nostre cervell pot passar d'un món material i objectiu a la ment immaterial i subjectiva i programació amb Python, amb intel·ligència artificial. Així com l'elaboració d'un mapa amb Tiled, i el codi, amb ajuda de llibreries com Pygame, Tkinter i la mateixa API de GPT-3 per crear un joc com a eina de difusió del coneixement adquirit i com GPT-3 pot arribar a simular consciència malgrat ser una màquina.

ABSTRACT

The purpose of the following work is to investigate consciousness from a biological, philosophical and computational point of view, in order to unify all the knowledge of our days in order to reach a conclusion about what makes us be (or not). Likewise, it is investigated whether an artificial intelligence (AI) is capable of being conscious, or simulating it.

Despite the fact that the issue of consciousness, of being itself, has accompanied the human being since ancient times, today it reappears thanks to the great advances in artificial intelligence and machine learning in recent decades, which make us doubt about what is consciousness itself. Is it a conscious robot? If it is, does it have the same level of consciousness as us? If not, can it be? Are there levels of consciousness?

Two fields at distant ends of human knowledge, but whose union bears fruit to the best technological advances. Themes such as Crick and the little-known side of him are explored; Edelman's theories, which include the rise of consciousness and primal emotions; the problems of consciousness, how our brain can go from a material and objective world to the immaterial and subjective mind and programming with Python, with artificial intelligence. As well as the elaboration of a map with Tiled, and the code, with the help of libraries such as Pygame, Tkinter and the GPT-3 API itself for the creation of a game as a tool for disseminating the knowledge acquired and how GPT-3 can reach to simulate consciousness despite being a machine.

ABSTRACT:	3
------------------------	----------

ÍNDICE	5
---------------------	----------

1 LA CONSCIENCIA	8
-------------------------------	----------

1.1 EL ETERNO DEBATE SOBRE CONSCIENCIA	8
1.2 CREANDO UNA NUEVA REALIDAD	9
1.3 PROBLEMAS DE LA CONSCIENCIA	9
1.3.1 The soft problema of consciousness	10
1.3.2 The hard problem of consciousness	11
1.4 SURGIMIENTO EVOLUTIVO DE LA CONSCIENCIA	11
1.2.2 Emociones primarias	13
1.2.2 Ramas de la consciencia	14
1.4.3 Teoría de Edelman sobre la aparición de la consciencia	15
1.4.4 Aparición de la consciencia primaria	16

2 UN MUNDO DE MÁQUINAS	18
-------------------------------------	-----------

2.1 PROGRAMACIÓN	19
2.1.1 Historia de la programación	19
2.2 IA, ML, NN	20
2.2.1 Inteligencia artificial	20
2.2.2 Machine Learning	21
2.2.3 Artificial neural networks	21
2.3	22
EL LENGUAJE PREFERIDO DE LAS IA	22
2.4 GPT-3	23

3 ELABORACIÓN DE UN JUEGO CON GPT-3	24
--	-----------

3.1 DISEÑOS	24
3.1.1 Mapa	24
3.1.2 Personajes	26
3.2 CÓDIGO	26
3.2.1 Programas	27
3.3 PANDEMIC	30

CONCLUSIÓN	32
REFERENCIAS DOCUMENTALES.....	33
ANEXOS.....	34

INTRODUCCIÓN

El tema de investigación en este trabajo se centra en la consciencia y en como una inteligencia artificial podría imitar este comportamiento, asimismo, se profundiza en el origen de la misma, y por qué fue lo que nos ayudó a sobrevivir a la selección natural.

Por estas cuestiones y porque me parece un tema que unifica todas las ramas del conocimiento que son de mi agrado, decidí hacer este trabajo, cuyo objetivo es lograr profundizar en el tema y utilizando una interfaz gráfica, con la que los jóvenes de hoy en día están muy familiarizados (los videojuegos), poder poner en contacto al usuario con el mayor modelo de lenguaje natural, GPT-3 de OpenAI. Con fin de que él mismo pueda experimentar cómo un simple robot que sigue ordenes, puede llegar a parecer consciente y tener conversaciones y un nivel de habla mayor al de la media. Escalofriante.

Primero hice una etapa de investigación, buscando información en libros, internet sobre la consciencia y cómo surgió. Después, una vez acabado esto, empezó la parte de programación del trabajo. La elaboración del juego divulgador.

En la consciencia, entramos en esta fase de búsqueda de información, empezando por los intentos de Crick, el premio Nobel, de describir e incentivar a la gente a hablar de la consciencia, luego siguen los problemas de la consciencia, que se centran en cómo puede nuestro cerebro hecho de carne hacer la mente, y que tiene de especial si lo tiene. Y antes de pasar a otro capítulo, se habla de las teorías de Edelman, emociones y consciencia primarias.

El capítulo Un mundo de máquinas, introduce al mundo de la programación, inteligencias artificiales, mecanismos de aprendizaje de estas y conocemos a GPT-3, que lo volveremos a mencionar en el siguiente capítulo.

Y el 3.º capítulo trata sobre la creación del juego divulgador, bautizado como Proyecto Pandemic. Se redactan el diseño del mapa, personajes, programas, se puede apreciar la estructura interna tanto del código como de los archivos en sí.

1 LA CONSCIENCIA

1.1 EL ETERNO DEBATE SOBRE CONSCIENCIA

Como en todos los ámbitos del conocimiento en los que nos queremos adentrar, primeramente, debemos realizar una presentación y definición de los conceptos clave a tratar, y sobre los cuales hablaremos mucho, más adelante.

Pero este caso es distinto, llevamos siglos, puede que milenios, intentando describir qué es la consciencia, pero aun así no llegamos a un consenso, tenemos muy claro que es la consciencia hasta que tenemos que definirla.

A continuación, se discutirán visiones sobre el surgimiento de la consciencia y las implicaciones que esta tiene sobre los animales.

Empecemos con uno de los nombres más reconocidos del ámbito biológico, Francis Crick, que junto a James Watson descubrieron la estructura del ADN, hecho que les mereció el premio Nobel de Medicina.

Crick dedicó los últimos 25 años de su vida, en los que publicó 23 trabajos de investigación, dos libros de neurociencia y la consciencia, y varios artículos. Esto debido a que encontró peculiar que la mayoría de los trabajos sobre la neurociencia no hacía referencia a la consciencia.

Aunque al final de su vida, Crick no descubrió grandes verdades sobre la consciencia, ayudó a dar voz a la ciencia de la consciencia y volver a despertar el deseo de los investigadores respecto a ese campo. En los años 1980, había aproximadamente 3.000 investigadores en el campo de la neurociencia, años después de su muerte, esta cifra subió hasta los 35.000 (2009),

Él animó a los demás científicos a hablar sobre la consciencia, que parecía ser un tema tabú y del que todo el mundo parecía aceptar que era una presencia misteriosa, como un alma.

Crick, en cambio, rechazaba cualquier explicación no física sobre esta diciendo: “un alma incorpórea es tan innecesaria como la vieja idea de que había una Fuerza de Vida.”(LA BUSQUEDA CIENTIFICA DEL ALMA, 1994).

Dado esta negación de cualquier teoría que involucrara fuerzas externas al mundo físico o simples especulaciones que tuvieran o no que ver con entes divinos, Crick concluyó una de las ideas más evidentes, a la par que profundas y demoledoras, y es que:

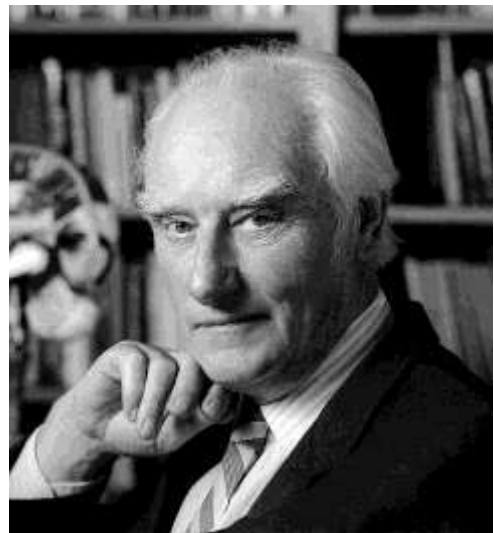


Ilustración 1: Francis Crick (1916-2004)

“Tú, tus alegrías y tus penas, tus recuerdos y tus ambiciones, tu sentido de identidad personal y libre albedrío, no son de hecho más que el comportamiento de un vasto conjunto de células nerviosas y sus moléculas asociadas”(LA BUSQUEDA CIENTIFICA DEL ALMA, 1994)

1.2 CREANDO UNA NUEVA REALIDAD

En la ilustración de la derecha se puede observar la representación de la consciencia de Robert Fludd. En ella se ve como, a través de nuestros sentidos, podemos percibir el mundo sensible (terminología usada por Platón para el dualismo platónico), e interpretarlo con nuestra mente, creando una nueva realidad intangible, la de la mente.

Entonces, nuestra mente es capaz de obtener información del mundo y transformarlo en las experiencias que tenemos, en memorias, en opiniones subjetivas como gustos o creencias.

Pero, el problema se halla en el proceso de esto mismo, ¿Cuáles son los mecanismos que utiliza nuestro cerebro y nos permite, de algo objetivo, de ondas de luz y agrupaciones de moléculas totalmente imparciales, sacar subjetividad, colores, sabores e infinidad de otras cosas? ¿Es esto a lo que se refería Crick cuando afirmó que solo somos nuestra mente?



Ilustración 2: utriusque cosmi maioris scilicet et minoris [...] historia, tomus II (1619), tractatus I, sectio I, liber X, De triplici animae in corpore visione - Robert Fludd

Este conjunto de preguntas y todas las que de ellas se derivan, fueron llamadas “Hard and Soft problems of consciousness”.

1.3 PROBLEMAS DE LA CONSCIENCIA

Dentro de estos problemas, podemos diferenciar dos, siendo estos: “The hard problem of consciousness” y “The soft problem of consciousness”, comenzaremos hablando primero de este último mencionado.

1.3.1 The soft problema of consciousness

Este problema alude a que todavía no tenemos conocimiento de cómo el cerebro trabaja para crear aquello para lo que parece ser creado. Desconocemos completamente cómo el cerebro hace emerger la consciencia, si solo ciertas partes del cerebro son encargadas, si es una red global, al igual que en las redes neuronales, ni cualquiera de las otras posibilidades que se nos puedan ocurrir. Y es que no podemos sacar ninguna conclusión definitiva a esta pregunta.



Ilustración 3: Representación artística del flujo de la consciencia

Este problema, pese a ser el más sencillo de los dos, ya supone una gran complicación para los neurocientíficos, y es difícil saber si siquiera podremos resolverlo, ya que quizás nuestra mente no esté preparada para poder entenderse a ella misma. De esto hablaría Jostein Gardner en su novela, *El misterio del solitario*:

"Si nuestro cerebro fuera tan sencillo como para poder entenderlo, seríamos tan tontos que, de todos modos, no lo podríamos entender.

Me quedé un buen rato meditando sobre esta frase. Al final, llegué a la conclusión de que la frase decía más o menos todo lo que podía decirse sobre la pregunta que yo había hecho.

Mi viejo continuó:

—Porque hay cerebros mucho más simples que el nuestro. Por ejemplo, podemos, al menos hasta cierto punto, entender cómo funciona el cerebro de una lombriz. Pero la lombriz no puede; para eso, su cerebro es demasiado simple.

—Puede que haya un dios que nos entienda.

Mi viejo se sobresaltó. Creo que le impresionó un poco que yo fuera capaz de hacer una pregunta tan astuta.

—Puede ser. Pero, en ese caso, él sería tan enormemente complicado que seguramente no sería capaz de entenderse a sí mismo." (*El misterio del solitario*, 1990)

En este texto se interpreta dios como ser superior, porque no hace falta recurrir a dios para poder hacer la comparativa. Aun así, las palabras resultan devastadoras, ya que nos transmiten esta posible imposibilidad de poder llegar a conocernos completamente.

1.3.2 The hard problem of consciousness

Este, en cambio, atiende a la pregunta de cómo a partir de como la materia se vuelve mente, cómo a partir de algo material y objetivo como el cerebro se produce algo inmaterial y subjetivo como la mente. El problema radica en no saber que pieza del puzle hace que se genere esa “nueva dimensión”, a partir de todos los estímulos que recibimos. Lo único que sabemos es que se sitúa en el cerebro, pero entonces ¿Qué tienen las células cerebrales que no tengan las demás para poder producir la consciencia?

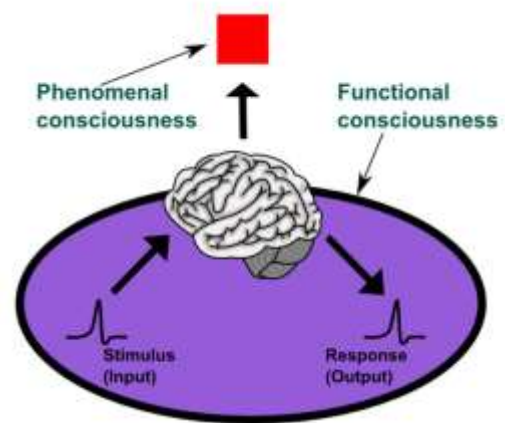


Ilustración 4: Hard problem of consciousness

Los científicos, pese a no haber encontrado respuesta clara a esto, tienen una idea postulada sobre que, sea lo que sea, la consciencia tiene que estar relacionada con procesamiento de información. Nuestro sistema nervioso funciona en esencia como un sistema de procesamiento de información. Las diversas conexiones neuronales que tenemos alrededor de nuestro cuerpo y cerebro sirven para interactuar con el mundo exterior, procesando esta información.

1.4 SURGIMIENTO EVOLUTIVO DE LA CONSCIENCIA

Dado todo esto, es común preguntarse en qué momento surgió la consciencia, y desde que momento podemos llamarla como tal.

Sabemos que somos conscientes porque, entre otras cosas, podemos distinguir entre qué pensamientos acaecen en nuestra propia cabeza, qué quisiéramos hacer y qué información procede del mundo exterior. La consciencia ha surgido con el progresivo desarrollo del cerebro desde los animales primitivos a los seres humanos porque lleva consigo un altísimo beneficio de supervivencia. Para un animal, ser capaz de formar imágenes en la mente y elegir una opción adecuada para la acción a la luz de la evaluación de su situación inmediata y de su memoria, de la experiencia pasada tiene una gran ventaja. Puede ser mucho mejor que responder de forma refleja, es decir, con un patrón inflexible que no se adapte según las circunstancias particulares. Puede significar la diferencia entre sobrevivir o morir. La supervivencia le permite transmitir a sus descendientes la estructura genética concreta que codificó la organización neuronal, promoviendo así una tendencia consciente a analizar consciente de las situaciones.

Para otros miembros de nuestra especie, podemos especular sobre lo que están pensando y verlo confirmado en acción. También podemos preguntar. El lenguaje permite esto y puede

describir hasta cierto punto lo que sucede en la mente. También existe la idea opuesta de que nadie sabe realmente que otras personas son conscientes, una posición conocida como solipsismo. Pero nadie realmente la cree.

Anticipándonos al análisis más detallado de la naturaleza operacional de los conceptos a que nos referiremos más adelante, puede señalarse aquí que Bernard Baars, en un influyente libro (*A Cognitive Theory of Consciousness*, Baars, 1988), propone una definición operacional de la consciencia. Plantea la cuestión de cuándo una persona razonable estaría de acuerdo en que un individuo acaba de tener una experiencia, es decir, una prueba objetiva de que ha visto un plátano. Sería consciente de un acontecimiento:

- 1) si pudiera decir inmediatamente después que era consciente de ello
- 2) si pudiéramos verificar de manera independiente la exactitud de su informe

Es decir, un criterio utilizado de forma muy común es el informe consciente verificable inmediato.

Baars indica que hay realmente cuatro componentes en la operación cuando se aplica al ser humano:

- 1) Consciencia de un acontecimiento
- 2) Capacidad retrospectiva de dirigir voluntad la remembranza consciente hacia el acontecimiento
- 3) La capacidad de recodificar (simbolizar) el acontecimiento en forma de palabras o gestos.
- 4) La capacidad de realizar voluntariamente dichas palabras o gestos.

Si nos ocupamos ahora de los animales, Griffin (1992) señala la afirmación del filósofo Kenny (1973), según la cual una esencia de la mente es tener la capacidad de operar símbolos de tal manera que sea la propia actividad de uno lo que los convierta en símbolos y les confiera significado. Griffin hace remontar el asunto al famoso ejemplo de simbolismo inherente a las danzas de las abejas melíferas, animales que tienen menos de un millón de neuronas. La propia actividad de la abeja hace de su danza oscilante una comunicación simbólica.

Quizá esto se ve mejor cuando la colmena enjambra y busca un nuevo hogar. Los exploradores de un enjambre que han visitado cavidades informan a sus hermanas de la distancia, la dirección y la conveniencia. La comunicación simbólica está muy alejada del objeto que describe. A veces, la danza oscilante puede referirse a cavidades o a una fuente de comida que la abeja hace algunas horas que ha visitado. Esencialmente, lo que se busca es una cavidad que esté libre de hormigas y que sea seca y oscura, y las danzas oscilantes

de diferentes abejas transmiten la información sobre la calidad; la decisión final que se alcanza es algo así como una decisión de grupo, un consenso al que no se llega rápidamente.

El marcado de abejas danzarinas individuales que hizo el científico Lindauer (1971), que fue quien realizó dicho estudio, estableció asimismo que una abeja que había adoptado una cavidad mediante su danza, después visitaba otras cavidades acerca de las cuales otra abeja hermana había danzado vigorosamente, y luego danzaba en favor de esta nueva cavidad y no de la primera que había defendido antes. Según la hipótesis de Griffin, la versatilidad comunicativa sugiere que las abejas expresan pensamientos sencillos.

1.2.2 Emociones primarias

Las emociones primarias indicaban que la existencia misma del organismo estaba amenazada. Ejemplos de emociones primarias incluyen la sed que aparece por desecación del organismo, o la falta de aliento, o “hambre de aire”, que tiene lugar cuando nos ahogamos o cualquier causa corta el suministro de aire. Se utiliza el término emoción primaria o primordial para el elemento subjetivo del comportamiento instintivo, que promueve el control de los sistemas vegetativos del cuerpo.

Estas sensaciones abrumadoras que se adueñan de toda la corriente de consciencia son coreografiadas desde las áreas inferiores o basales del cerebro.

Así, una gran parte de la organización de los sistemas vegetativos con sus detectores interoceptores está abajo, por así decirlo, en el sótano del cerebro (en la médula, el mesencéfalo y el hipotálamo), como los mismos procesos elementales de excitación y sueño (frontispicio). En cambio, las actividades de orden superior de la visión, el oído y los procesos cognitivos están organizados en gran medida en la materia gris de los lóbulos occipitales, parietales, temporales y frontales (la neocorteza), que son los niveles superiores y no el sótano del cerebro. De este punto de vista, sale la pregunta siguiente ¿En qué fase o fases de la evolución de los animales surgió por primera vez la consciencia?

Se trata de preguntas difíciles pero fascinantes. Abren un panorama enorme de biología. La primera pregunta debe centrarse, y lo hace, en el comportamiento animal y en cómo interpretarlo en los distintos niveles de desarrollo filogenético (evolutivo).

Desde luego, una vez las emociones primarias han invadido la corriente de consciencia en los animales superiores, los procesos de categorización de la percepción y de memoria entran en escena y tienen su importancia en la intención del animal que busca la gratificación inmediata de la necesidad, que es señalada por una emoción poderosa. La historia pasada del organismo contribuirá a la elección de una acción adecuada. De este modo, en la evolución

progresiva, los dos procesos (emoción primaria y categorización de la percepción) se entrelazarán y fortalecerán con una enorme ventaja de selección natural para el organismo.

1.2.2 Ramas de la consciencia

Baste decir que, mientras que la consciencia en los animales fue antaño un tema casi tabú, la mayoría de los neurobiólogos la aceptan en la actualidad como un determinante probable del comportamiento y no como un epifenómeno, o que los animales son simplemente autómatas. Es evidente que la aparición de la consciencia fue el acontecimiento estelar en la evolución de la raza humana. Pero ¿Hasta qué nivel inferior del árbol filogenético se aceptaría la posibilidad de consciencia?

A la hora de registrar las pruebas en varios niveles del árbol filogenético, es de gran importancia tener en cuenta las disciplinas del análisis evolutivo, tal como las establecieron Hodos y Campbell (1990). Es decir, muchas de las especies actuales de vertebrados estudiadas no se encuentran en el linaje evolutivo directo de *Homo sapiens*. Esto se aplica a los teleósteos (peces óseos), que constituyen un linaje distinto que evolucionó en el océano a partir de un antepasado primitivo que fue el antecesor de los primeros vertebrados que invadieron los ríos de agua continental. Del mismo modo, las aves no se hallan en un linaje evolutivo que dé origen a los seres humanos. Representan una rama surgida de los reptiles antiguos, que se encontraban asimismo en la línea directa hacia los seres humanos.

En realidad, el estudio del registro fósil indica que la evolución de los vertebrados se ha hecho a lo largo de muchas sendas independientes. El biólogo evolutivo posee el modelo de un árbol filogenético. En dicho modelo no hay jerarquía. Las aves no están más arriba ni más abajo que los mamíferos, sino que se encuentran simplemente en una rama diferente del árbol. La construcción es multilineal. Distintas líneas genealógicas radian desde un origen común. Diferentes linajes evolucionan de manera contemporánea e independiente unos de otros, y el árbol es dependiente de los datos reales e independiente de la complejidad de la función biológica. El tiempo es independiente de la secuencia, de manera que diferentes etapas de la evolución en los diferentes linajes pueden aparecer en la misma época temporal.

A detailed phylogenetic tree of the class Mammalia. The tree is rooted at the bottom left with 'Mammalia' and 'Mammalia' written vertically. The main trunk is labeled 'Mammalia'. The tree branches out into several major groups: 'Primates' (top center), 'Insectivores' (top left), 'Marsupials' (top right), 'Rodents' (middle left), 'Bats' (middle right), 'Monotremes' (bottom left), and 'Marsupials' (bottom right). Each group contains a list of representative species or subgroups. For example, 'Primates' includes 'Hominidae', 'Cercopithecoidea', 'Prosimia', and 'Prosimia'. 'Insectivores' includes 'Insectivores' and 'Insectivores'. 'Marsupials' includes 'Marsupials' and 'Marsupials'. 'Rodents' includes 'Rodents' and 'Rodents'. 'Bats' includes 'Bats' and 'Bats'. 'Monotremes' includes 'Monotremes' and 'Monotremes'. 'Marsupials' includes 'Marsupials' and 'Marsupials'.

coordinación en el espacio y en el tiempo de los diferentes eventos sensoriales y motores. Un ejemplo importante del proceso sería el caso de la visión. Así, una propiedad tal como el color de un objeto, percibida en una modalidad del cerebro, se uniría con otra que implicaría movimiento.

Edelman denomina a su teoría general NGST: Teoría Neuronal de Selección de Grupo.

1.4.4 Aparición de la consciencia primaria

Edelman dice que es el desarrollo evolutivo de la capacidad de crear una “escena” lo que condujo a la aparición de la consciencia primaria, la cual define como la capacidad de construir una escena mental integrada en el presente. Esto no requiere lenguaje ni un verdadero sentido del yo.

Se ve en animales con determinadas estructuras cerebrales similares a las nuestras. La consciencia de orden superior, ejemplificada en el ser humano, presupone la coexistencia de la consciencia primaria e implica un sentido del yo. Esta capacidad de conocimiento de sí mismo ha sido descrita de muchas maneras, como “la consciencia dirigida hacia el interior” (Hughling Jackson), “hacer de la mente el sujeto y el objeto de sí mismo” (Isaiah Berlin), o ser consciente de ser consciente.

El elemento clave del proceso de generar una escena es la categorización perceptual. Edelman dice que la categorización perceptual implica el proceso por el que las numerosas señales que llegan desde el mundo externo al mismo tiempo, y que pueden no tener ninguna conexión causal o física entre sí, son convertidas en señales que son útiles para una especie concreta.

Un ejemplo adecuado de esta idea teórica en relación con el aprendizaje de un concepto sobre esta base es el de los experimentos de Herrnstein y Loveland. (Herrnstein y Loveland, 1964; Herrnstein et al, 1976).

Se trata de un experimento en el que un grupo de palomas, aprendieron a picotear una diapositiva proyectada que incluía un ser humano para conseguir abrir la tolva de comida. En el experimento, cuando las palomas picoteaban con más frecuencia a imágenes con seres humanos, se introdujo una serie completamente nueva de diapositivas, positivas, y negativas, y siguieron picoteando más a las imágenes que contenían personas (70-80%).

Cuando consideramos un animal que se mueve, las entradas paralelas procedentes de muchas modalidades sensoriales diferentes resultan en correlaciones reentrantes entre complejos de categorías perceptuales relacionadas con objetos y acontecimientos. Edelman dice que su prominencia está regida por la actividad de los sistemas de valores del animal, que implica recuerdos de recompensa y castigo en el pasado. La memoria a corto plazo, que

es fundamental la consciencia primaria, refleja experiencias previas, categóricas y conceptuales. Así, la capacidad de construir una escena consciente gira sobre la capacidad de construir en fracciones de segundo un presente recordado.

Edelman sugiere que un animal que posee solo consciencia primaria puede generar una “imagen mental” o escena. Esto se basa en parte en la categorización perceptual multimodal inmediata en tiempo real. Está determinado por la sucesión de acontecimientos reales en el ambiente. Considera que este animal posee individualidad biológica, pero carece del concepto del yo. Mientras que posee un presente recordado, no tiene concepto del pasado o del futuro.

Examinemos ahora la sugerencia de Edelman de que el animal con consciencia primaria no tiene concepto de pasado o de futuro. Tales conceptos únicamente aparecieron en la evolución cuando surgieron las capacidades semánticas, quizá muy temprano en los precursores de los homínidos.

Parece que cabe poca duda de que la consciencia de orden superior floreció con la capacidad lingüística que se inició en los precursores de *Homo sapiens*. El habla permitió la descripción de sentimientos y el uso de símbolos. Sin embargo, la posibilidad de intención parece hallarse implícita en algunos elementos del concepto de “la escena” de Edelman. Es decir, es inherente a la imagen mental postulada. La ventaja de supervivencia de la aparición de la capacidad de crear una escena se debería en cierta medida a la acción o intencionalidad implícita.

2 UN MUNDO DE MÁQUINAS

En un mundo como el que vivimos actualmente, en el que estamos constantemente conectados mediante una gran red invisible que años atrás habrían calificado de loco (como mínimo) al que se le ocurriese algo semejante, constantemente olvidamos cómo funcionan estos aparatos.

Para el público, el mundo de la computación, tecnología y sus avances más sorprendentes simplemente no son de su interés y lo ven como algo lejano, reservado para aquellos con una mente racional muy superior a la media (o como diría milenios atrás Platón, con un alma racional) y con aficiones que se tildarían de aburridas.

Pero no hay nada más alejado de la realidad, y es que vivimos en una época en la que la tecnología está empezando a despegar y estamos siendo testigos de cómo nacen nuevas eras para la humanidad. Como ya dijo el escritor británico de ciencia ficción Arthur C. Clarke, “Cualquier tecnología suficientemente avanzada es indistinguible de la magia”

Este pensamiento generalizado, que ya está volviéndose un estilo de vida, está acabando con la introspección y reflexión que solía ser fruto del aburrimiento o de ganas de sentirse más lleno y obtener recompensas bioquímicas que nuestro cerebro interpreta como felicidad, si lo queremos llevar a un lado más descriptivo. Es por esto por lo que en la medida en que la tecnología se use solo para obtener recompensas fáciles, como navegar por Instagram u otras redes sociales, obstaculiza que podamos crear mejores hábitos al momento de crecer como personas y conocernos a nosotros mismos.

Mientras que unos hacen todo lo posible para avanzar como sociedad, otros no saben sacar el lado más provechoso de estos y los acaba utilizando para recompensas instantáneas. Al final, la historia se repite, ya que siempre han sido unos pocos los que han hecho avanzar a la sociedad, mientras que los otros no tienen aportación alguna.

La mayoría de la población no tiene idea de en cuantos ámbitos están involucradas las inteligencias artificiales, se subestima su poder y no se les considera lo suficiente. Desde acompañarnos en la conducción de un vehículo inteligente, pasando por los simples pero útiles asistentes virtuales, tales como Siri, Cortana o el mismo asistente de Google, hasta sistemas de recomendación de artículos en páginas como Amazon o de contenido en YouTube.

Pero estas razones, presentaremos los fundamentos de la programación, el lenguaje por excelencia en estos casos, IA, aprendizaje automático, redes neuronales, entre otras.

2.1 PROGRAMACIÓN

Según la RAE, la programación es la acción de preparar ciertas máquinas o aparatos para que empiecen a funcionar en el momento y en la forma deseados. Y es cierto, aunque se puede ampliar, ya que la programación es una forma de comunicarse con la máquina para poder decirle lo que hacer, en una serie de órdenes a seguir, como si fuese una receta.

La acción de programar va ligada al lenguaje de programación, que es la herramienta que le permite al programador ordenar acciones a un terminal.

2.1.1 Historia de la programación

La primera evidencia de algo relacionado con la programación se da en 1801, gracias a Joseph Marie Jacquard, que creó el primer sistema de programación integrado a un telar que recibía un código y automatizaba el proceso.

Pero la programación como la entendemos hoy en día, aparece en 1957 de la mano de John W. Backus, creando el primer lenguaje de programación de alto nivel, Fortran. Este mismo fue producido para solucionar las carencias de los que ya había por ese momento, como lo confusos que eran, haciendo que se acercaran más a la matemática y, por ende, sean más comprensibles.

Otras fechas importantes para tener en cuenta, por ejemplo, 1972, cuando nació el conocidísimo lenguaje C, como un lenguaje básico de programación, de un nivel simple, que pronto adquiriría una relevancia vital, hasta acabar convirtiéndose en uno de los lenguajes más utilizados en la actualidad. O 1979, con la llegada de C++ con la idea de añadir a C capacidad de manipular objetos.

Acompañados de la llegada de internet, llegaron numerosos lenguajes de programación que se siguen utilizando hoy en día. Lenguajes como HTML, Python, Visual

Basic, Java, JavaScript o PHP fueron estos que surgieron, y hoy en día se siguen empleando en la amplia gama de páginas web.



Ilustración 6: lenguajes de programación populares

2.2 IA, ML, NN

Toda esta terminología y abreviaturas pueden abrumar al principio, es por eso por lo que iremos paso por paso revisando cada uno de estos conceptos para poder continuar con nuestra comprensión del campo.

2.2.1 Inteligencia artificial

¿A qué nos referimos cuando hablamos de inteligencia? Muchos diccionarios y enciclopedias intentan dar con su definición única, aunque entre ellas no siempre coinciden. Existen innumerables formas de describir la inteligencia, revisemos algunas de ellas:

- “Capacidad de entender o comprender” (Diccionario de la lengua española [23.a], Real Academia Española, 2014).
- “Capacidad de resolver problemas” (ídem).
- “Conocimiento, comprensión, acto de entender” (ídem).
- “Capacidad de adaptarse eficazmente al entorno, ya sea generando un cambio en uno mismo o modificando el entorno o encontrando un entorno nuevo” (Enciclopedia Británica, 2006).
- “Capacidad de resolver problemas, o crear productos, que son valorados dentro de uno o más contextos culturales” (Howard Gardner, Frames of Mind: Theory of multiple intelligences, 1993).
- “Capacidad de adquirir capacidad” (Herbert Woodrow).”
- “Realizar una operación sobre un tipo de contenido específico para producir un producto particular” (Joy Paul Guilford).

Por tanto, no hay un consenso sobre la inteligencia. Debido a esto, es aún más complicado definir la inteligencia artificial (IA) o la capacidad de la máquina para reproducir la inteligencia humana.

Tomando de ejemplo a Stuart Russell y Peter Norvig que en su libro Inteligencia artificial: un enfoque moderno (la obra sobre el tema más estudiada del mundo), se distinguen cuatro enfoques diferentes. Según los autores, la inteligencia artificial suele ser definida como la capacidad de una máquina de realizar:

- Pensar de forma humana
- Pensar racionalmente
- Actuar de forma humana
- Actuar racionalmente

Es necesaria esta clasificación, ya que se pueden visualizar los diferentes objetivos que tienen dentro del mundo de la programación al momento de ponerse a crear inteligencias artificiales. Mientras que algunos intentan crear máquinas que piensen como seres humanos, para poder entender mejor los mecanismos de nuestra mente, otros no se complican tanto y crean máquinas que actúen como humanos, sin importar el proceso interno que se realiza para llegar a ese resultado.

2.2.2 Machine Learning

El Machine Learning (ML) es una rama del campo de la inteligencia artificial que, a través de algoritmos, dota a las máquinas de la capacidad de identificar patrones en datos masivos y elaborar predicciones. Este aprendizaje permite a las máquinas ejecutar tareas específicas de forma autónoma, es decir, sin necesidad de ser programados.

Los algoritmos de Machine Learning se dividen en tres grandes grupos:

- Aprendizaje supervisado: estos cuentan con un aprendizaje previo basado en una base de datos en las que hay etiquetas asociadas a unas respuestas que les permiten tomar decisiones o hacer predicciones.
- Aprendizaje no supervisado: estos no cuentan con un entrenamiento anterior. Se enfrentan de golpe al caos de datos con el objetivo de encontrar patrones que permitan organizarlos de alguna manera.
- Aprendizaje por refuerzo: su objetivo es que un algoritmo aprenda a partir de la propia experiencia. Es decir, que sea capaz de tomar la mejor decisión ante diferentes situaciones a través de prueba y error en un sistema de recompensas, en el que se dan mejores recompensas para aquellas decisiones que son las más adecuadas.

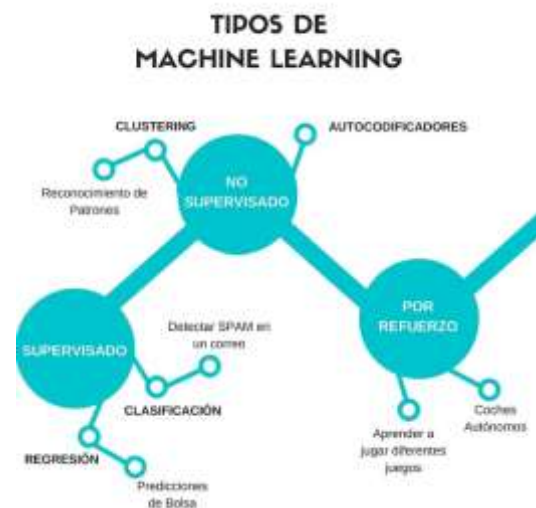


Ilustración 7: tipos de machine learning

2.2.3 Artificial neural networks

Son sistemas informáticos inspirados en las redes neuronales biológicas que forman el cerebro de los animales.

Se componen de capas de neuronas, que contienen una capa de entrada, una o más capas ocultas y una capa de salida. Cada neurona artificial se conecta con otra y tiene asociada un peso (escalar que determina su valor).

“Las redes neuronales se basan en datos de entrenamiento para aprender y mejorar su precisión con el tiempo. Sin embargo, una vez que estos algoritmos de aprendizaje se ajustan con precisión, se convierten en herramientas poderosas en informática e inteligencia artificial, lo que nos permite clasificar y agrupar datos a alta velocidad. Las tareas de reconocimiento de voz o

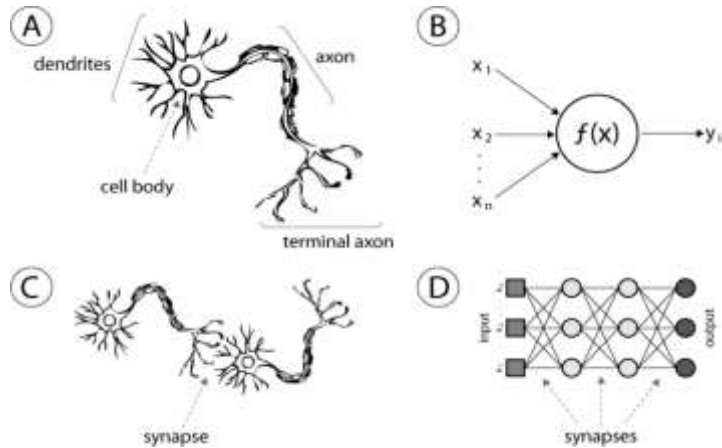


Ilustración 8: símil de neurona humana con neurona computacional

reconocimiento de imágenes pueden llevar minutos en lugar de horas en comparación con la identificación manual por parte de expertos humanos.” (IMB Cloud Education, 2020)

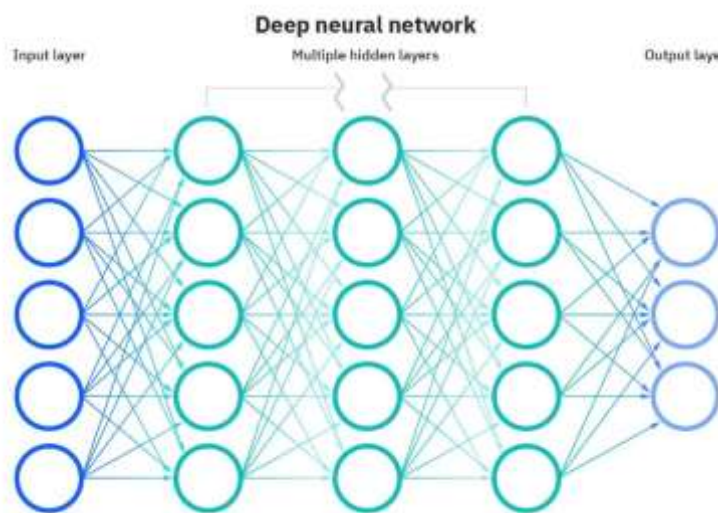


Ilustración 9: representación deep neural network

2.3

EL LENGUAJE PREFERIDO DE LAS IA

El lenguaje por excelencia para las IA es Python, ya que es uno de los lenguajes de programación más usados en la actualidad. Es muy completo: es de código abierto, es fácil de entender y su sintaxis es sencilla, por lo que ahorra tiempo y recursos. Por todas estas razones es recomendable para iniciarse en la programación.

Gracias a bibliotecas como Keras o TensorFlow (desarrollado por Google), Python es uno de los predilectos al momento de hablar de IA. También puede ser de útil para Big Data, gracias a bibliotecas de procesamiento de datos.

2.4 GPT-3

Es un enorme modelo de lenguaje de OpenAI que está entrenada para que dada una secuencia de texto pueda predecir como tiene que completarla, con la que puedes generar texto natural realista. Pero, ¿Qué es un modelo de lenguaje?

Un modelo de lenguaje, un modelo que se utiliza para predecir la siguiente palabra en una secuencia de palabras. La idea básica detrás de un modelo de lenguaje es capturar la probabilidad de observar una secuencia de palabras.



Ilustración 10: logo GPT-3

El ejemplo clásico de un modelo de lenguaje es un modelo de unigrama, que predice la probabilidad de observar una palabra en particular en función de las frecuencias de palabras individuales en el corpus.

Un modelo de lenguaje más sofisticado es un modelo de bigrama, que predice la probabilidad de observar una palabra en función de las frecuencias de la palabra anterior en el corpus. Los modelos de bigrama se emplean a menudo en los sistemas de reconocimiento de voz para predecir la probabilidad de una palabra en función de las propiedades acústicas de la palabra anterior.

Con GPT-3 se pueden hacer varias funciones predeterminadas como, chatear, traducir, corregir gramaticalmente, entre otras. Además, dispone de una API abierta al público de fácil acceso, que usaré para mi parte práctica.

3 ELABORACIÓN DE UN JUEGO CON GPT-3

Para poder unir estos dos mundos a simple vista distantes, voy a utilizar GPT-3, ya presentado, en un videojuego, estilo Zelda, 2D con Python. Para demostrar que Python, el lenguaje de las IA es versátil. He escogido GPT-3 para este proyecto debido a que ya ha habido varios incidentes en los que expertos de la IA han manifestado creer que los modelos de lenguaje habían adquirido vida propia y consciencia por el modo en el que hablaban.

Y al ser GPT-3 el mayor modelo de lenguaje, puede llegar a causar más impacto en cuanto a conversaciones profundas se refiere. Simulando así la presencia de consciencia. A este proyecto lo he llamado Proyecto Pandemic.

3.1 DISEÑOS

Antes de meternos de lleno en el código, debemos tener una base sobre la cual trabajar y la cual podamos visualizar. Sin ella, podríamos programar todo, pero no veríamos y de nada serviría para poder cumplir mi objetivo

3.1.1 Mapa

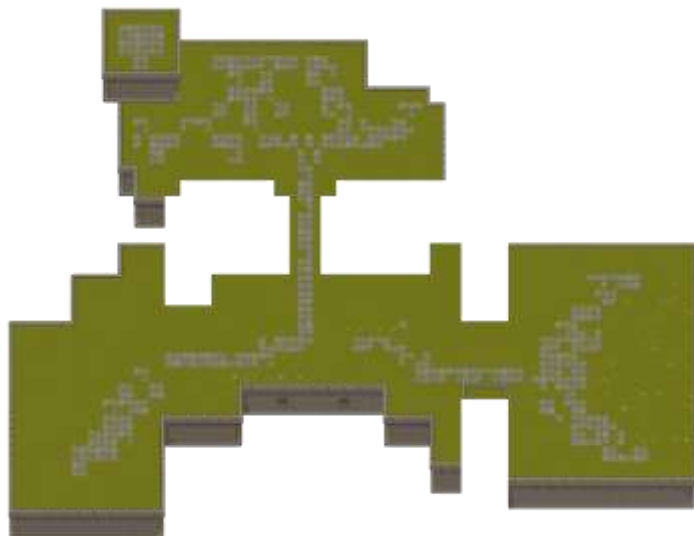
Todos los sprites (gráficos que se puede mover en la pantalla y manipular de otra manera como una sola entidad) son de dominio público bajo el nombre Pixel Art Top Down. El autor autoriza el uso de su trabajo para otros proyectos, tanto comerciales como personales.

Para desarrollar el mapa del juego, he usado el programa Tiled, que es un editor de niveles fácil de usar y gratuito.

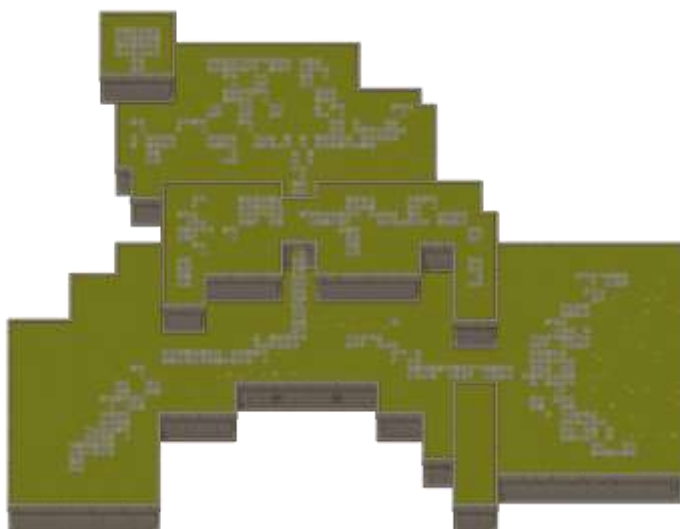
Primero diseñé la parte inferior del mapa (el suelo), la cual está hecha tanto como por el camino que el personaje debe seguir, como decoración simple como flores.



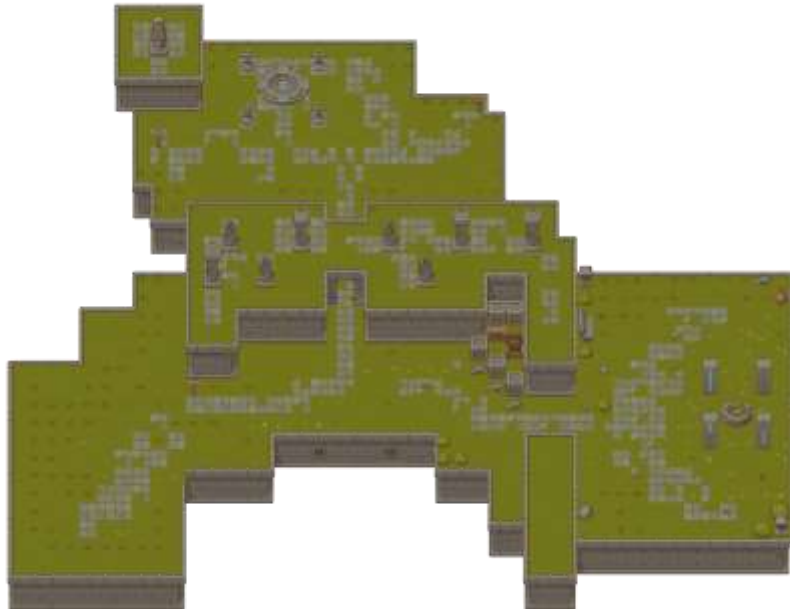
A continuación, agregué las paredes de la primera planta y algunas que podrían dar problemas al momento de programar las colisiones y superposición del personaje con el escenario.



Para darle un estilo más inmersivo al juego, y siguiendo la estética del autor del tileset original, se emplea una segunda planta para dar la sensación de 3D. Además, con esto agregué la superposición de la nueva planta sobre el personaje para que tuviera sentido y no desenchajara.

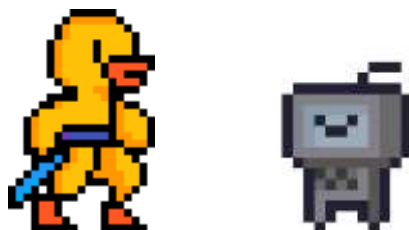


Por último, el mapa necesita decoración, y esto lo logré con las mismas imágenes que proporciona el conjunto ya mencionado. Agregué columnas, cruces, esculturas, carteles, plantas y hierbas para darle un toque más artístico.



3.1.2 Personajes

El personaje principal ha sido diseñado por un artista privado, así que es de mi autoría, en cambio, el personaje que interpreta a GPT-3, fue conseguido gracias a la colaboración de David Harrington, que bajo el nombre de Píxel robot, permite el uso de su arte para todos los proyectos.



3.2 CÓDIGO

Para el código, utilicé el lenguaje afín para implementar una IA, como ya expliqué anteriormente, Python. Haciendo uso de diversas librerías como Pygame, Tkinter, OS, openai, y empleando programación orientada objetos, que se basa en la creación de clases, logré

conseguir una interfaz para interactuar con GPT-3 que simula un juego y tiene un gran apartado gráfico, teniendo en cuenta que solo hay una persona detrás.

El juego se encuentra dividido en 9 archivos .py que se conectan internamente entre sí, importándose unos a otros y declarando objetos de otras clases, y asimismo utilizando sus métodos y cambiando sus atributos. Esto está hecho con el propósito de ser más ordenado y

Nombre	Fecha de modificación	Tipo	Tamaño
.vscode	04/10/2022 20:45	Carpeta de archivos	
__pycache__	12/10/2022 21:49	Carpeta de archivos	
.env	04/10/2022 17:17	Archivo ENV	1 KB
entity	12/10/2022 22:19	Archivo de origen ...	2 KB
is	12/10/2022 22:20	Archivo de origen ...	3 KB
level	12/10/2022 22:21	Archivo de origen ...	5 KB
main	10/10/2022 21:58	Aplicación	77.527 KB
main	12/10/2022 21:51	Archivo de origen ...	1 KB
openai_api	12/10/2022 22:22	Archivo de origen ...	3 KB
player	12/10/2022 22:23	Archivo de origen ...	3 KB
settings	12/10/2022 22:23	Archivo de origen ...	1 KB
support	12/10/2022 22:23	Archivo de origen ...	1 KB
tile	12/10/2022 22:24	Archivo de origen ...	1 KB

leíble, además de que, en caso de suceder un error, se podría rastrear con mayor facilidad para poder corregirlo. Además de esos 9 archivos, hay otro archivo .env que contiene las credenciales para poder acceder a la API de GPT-3.

3.2.1 Programas

Main.py es el que controla todos los demás, es el principal como indica su nombre. A partir de este los demás se ejecutan siguiendo el orden en el que están programados. Además, es el que inicia la interfaz gráfica y permite la interacción con la ventana, así como inicia todos sus parámetros.

```

1 import sys, os
2 from settings import *
3 from level import level
4
5 class Game:
6     def __init__(self):
7         pygame.init()
8
9         self.screen = pygame.display.set_mode((width, height))
10        pygame.display.set_caption("Personas")
11        self.clock = pygame.time.Clock()
12        self.level = level()
13
14    def run(self):
15        while True:
16            for event in pygame.event.get():
17                if event.type == pygame.QUIT:
18                    pygame.quit()
19                    sys.exit()
20
21            self.screen.fill((0,0,0))
22            self.level.run()
23            pygame.display.update()
24            self.clock.tick(60)
25
26 if __name__ == '__main__':
27     game = Game()
28     game.run()
29
30 # author: valentin el futbolista de la

```

```

1 import pygame
2 from settings import TILESIZE
3 from tile import Tile
4 from player import Player
5
6 from support import import_csv, loadmap, import_folder
7 from la import LA
8
9 class Level():
10
11     def __init__(self):
12
13         self.all_sprites = pygame.sprite.Group()
14         self.all_sprites = pygame.sprite.Group()
15
16         self.all_sprites = pygame.sprite.Group()
17         self.all_sprites = pygame.sprite.Group()
18
19         self.create_map()
20
21     def create_map(self):
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

```

Level.py es el encargado de iniciar el nivel, es decir, crear el mapa, las capas, relacionar las imágenes de los tilesets con sus posiciones (x,y) gracias a archivos .csv exportados de Tiled que indican posición y qué tile le corresponde. También imprime en pantalla todo esto, ya que, aunque a nivel interno este ejecutado, sigue sin ser visible hasta que se ejecute el método run de Level().

A esto se le agrega una superposición en la dirección Y de los objetos, porque de esta forma le da profundidad al juego y es más realista y placentero a la vista.

```

1 from os import path
2 from os import walk
3 import pygame
4
5 def import_csv(filepath):
6     terrain_map = {}
7     with open(filepath) as level_map:
8         reader = csv.reader(level_map, delimiter=',')
9         for row in reader:
10             terrain_map.append(int(row))
11     return terrain_map
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

```

Support.py y settings.py definen constantes y funciones que ayudaran a agilizar el trabajo de los demás programas para que no tengan un exceso de código y no sea tan repetitivo. También

```

1 WIDTH = 1200
2 HEIGHT = 700
3 FPS = 60
4 TILESIZE = 32
5
6 NPC = {
7     'Pondokio' : {}
8 }
9
10

```

permiten que, cambiando simplemente la constante, se cambien todos los valores asociados a esta y no tener que ir uno por uno revisándolos.

Entity.py es el módulo que nos permite crear más fácilmente los NPC y el personaje principal, dotándolos de ciertas variables y métodos comunes que luego se acabaran de definir y adaptar a cada uno de estos en su respectivo módulo.

Permite su movimiento, y busca si ha habido alguna colisión para poder actuar en consecuencia.

```

1 import pygame
2
3 class Entity(pygame.sprite.Sprite):
4
5     def __init__(self, group):
6         super().__init__(group)
7         self.frame_index = 0
8         self.animation_speed = 0.25
9         self.direction = pygame.math.Vector2()
10
11     def move(self, speed):
12         if self.direction.magnitude() > 0:
13             self.direction = self.direction.normalize()
14             self.hitbox.x += self.direction.x * speed
15             self.all_sprites.move(self)
16             self.hitbox.y += self.direction.y * speed
17             self.all_sprites.move(self)
18             self.rect.center = self.hitbox.center
19
20     def collision(self, direction):
21         if direction == 'horizontal':
22             for sprite in self.all_sprites:
23                 if self.hitbox.colliderect(sprite.hitbox):
24                     if self.direction.x > 0:
25                         self.hitbox.right = sprite.hitbox.left
26                     if self.direction.x < 0:
27                         self.hitbox.left = sprite.hitbox.right
28
29         if direction == 'vertical':
30             for sprite in self.all_sprites:
31                 if self.hitbox.colliderect(sprite.hitbox):
32                     if self.direction.y > 0:
33                         self.hitbox.bottom = sprite.hitbox.top
34                     if self.direction.y < 0:
35                         self.hitbox.top = sprite.hitbox.bottom
36

```

El módulo player.py crea el personaje principal y controla su movimiento y velocidad según el usuario toque las teclas de dirección. Además de eso, le da animación para correr y moverse por el mapa con su método self.animate().

```

1 from pygame import *
2 from pygame.locals import *
3 from pygame import *
4 from pygame import *
5 from pygame import *
6
7 class Player(pygame.sprite.Sprite):
8
9     def __init__(self, pos, group, obstacle_sprites):
10         super().__init__(group)
11         self.image = pygame.image.load('player.png')
12         self.rect = self.image.get_rect()
13         self.status = "idle"
14         self.frame_index = 0
15         self.frame_speed = 0.3
16
17     def animate(self):
18         self.frame_index += 1
19         if self.frame_index == len(self.frame_images):
20             self.frame_index = 0
21         self.image = self.frame_images[self.frame_index]
22         self.rect = self.image.get_rect()
23
24     def get_status(self):
25         return self.status
26
27     def set_status(self, status):
28         self.status = status
29
30     def actions(self):
31         self.frame_index = 0
32         self.frame_speed = 0.3
33         self.frame_images = self.frame_images
34         self.frame_index = 0
35         self.frame_speed = 0.3
36         self.frame_images = self.frame_images
37         self.frame_index = 0
38         self.frame_speed = 0.3
39         self.frame_images = self.frame_images
40         self.frame_index = 0
41         self.frame_speed = 0.3
42         self.frame_images = self.frame_images
43         self.frame_index = 0
44         self.frame_speed = 0.3
45         self.frame_images = self.frame_images
46         self.frame_index = 0
47         self.frame_speed = 0.3
48         self.frame_images = self.frame_images
49         self.frame_index = 0
50         self.frame_speed = 0.3
51         self.frame_images = self.frame_images
52         self.frame_index = 0
53         self.frame_speed = 0.3
54         self.frame_images = self.frame_images
55         self.frame_index = 0
56         self.frame_speed = 0.3
57         self.frame_images = self.frame_images
58         self.frame_index = 0
59         self.frame_speed = 0.3
60         self.frame_images = self.frame_images
61         self.frame_index = 0
62         self.frame_speed = 0.3
63         self.frame_images = self.frame_images
64         self.frame_index = 0
65         self.frame_speed = 0.3
66         self.frame_images = self.frame_images
67         self.frame_index = 0
68         self.frame_speed = 0.3
69         self.frame_images = self.frame_images
70         self.frame_index = 0
71         self.frame_speed = 0.3
72         self.frame_images = self.frame_images
73         self.frame_index = 0
74         self.frame_speed = 0.3
75         self.frame_images = self.frame_images
76         self.frame_index = 0
77         self.frame_speed = 0.3
78         self.frame_images = self.frame_images
79         self.frame_index = 0
80         self.frame_speed = 0.3
81         self.frame_images = self.frame_images
82         self.frame_index = 0
83         self.frame_speed = 0.3
84         self.frame_images = self.frame_images
85         self.frame_index = 0
86         self.frame_speed = 0.3
87         self.frame_images = self.frame_images
88         self.frame_index = 0
89         self.frame_speed = 0.3
90         self.frame_images = self.frame_images
91         self.frame_index = 0
92         self.frame_speed = 0.3
93         self.frame_images = self.frame_images
94         self.frame_index = 0
95         self.frame_speed = 0.3
96         self.frame_images = self.frame_images
97         self.frame_index = 0
98         self.frame_speed = 0.3
99         self.frame_images = self.frame_images
100        self.frame_index = 0

```

```

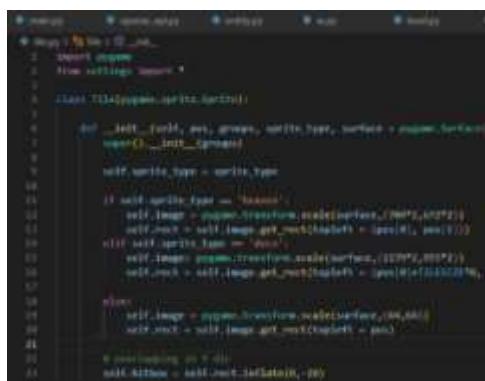
1 from pygame import *
2 from pygame.locals import *
3 from pygame import *
4 from pygame import *
5 from pygame import *
6
7 class Player(pygame.sprite.Sprite):
8
9     def __init__(self, pos, group, obstacle_sprites):
10        super().__init__(group)
11        self.image = pygame.image.load('player.png')
12        self.rect = self.image.get_rect()
13        self.status = "idle"
14        self.frame_index = 0
15        self.frame_speed = 0.3
16        self.frame_images = self.frame_images
17        self.frame_index = 0
18        self.frame_speed = 0.3
19        self.frame_images = self.frame_images
20        self.frame_index = 0
21        self.frame_speed = 0.3
22        self.frame_images = self.frame_images
23        self.frame_index = 0
24        self.frame_speed = 0.3
25        self.frame_images = self.frame_images
26        self.frame_index = 0
27        self.frame_speed = 0.3
28        self.frame_images = self.frame_images
29        self.frame_index = 0
30        self.frame_speed = 0.3
31        self.frame_images = self.frame_images
32        self.frame_index = 0
33        self.frame_speed = 0.3
34        self.frame_images = self.frame_images
35        self.frame_index = 0
36        self.frame_speed = 0.3
37        self.frame_images = self.frame_images
38        self.frame_index = 0
39        self.frame_speed = 0.3
40        self.frame_images = self.frame_images
41        self.frame_index = 0
42        self.frame_speed = 0.3
43        self.frame_images = self.frame_images
44        self.frame_index = 0
45        self.frame_speed = 0.3
46        self.frame_images = self.frame_images
47        self.frame_index = 0
48        self.frame_speed = 0.3
49        self.frame_images = self.frame_images
50        self.frame_index = 0
51        self.frame_speed = 0.3
52        self.frame_images = self.frame_images
53        self.frame_index = 0
54        self.frame_speed = 0.3
55        self.frame_images = self.frame_images
56        self.frame_index = 0
57        self.frame_speed = 0.3
58        self.frame_images = self.frame_images
59        self.frame_index = 0
60        self.frame_speed = 0.3
61        self.frame_images = self.frame_images
62        self.frame_index = 0
63        self.frame_speed = 0.3
64        self.frame_images = self.frame_images
65        self.frame_index = 0
66        self.frame_speed = 0.3
67        self.frame_images = self.frame_images
68        self.frame_index = 0
69        self.frame_speed = 0.3
70        self.frame_images = self.frame_images
71        self.frame_index = 0
72        self.frame_speed = 0.3
73        self.frame_images = self.frame_images
74        self.frame_index = 0
75        self.frame_speed = 0.3
76        self.frame_images = self.frame_images
77        self.frame_index = 0
78        self.frame_speed = 0.3
79        self.frame_images = self.frame_images
80        self.frame_index = 0
81        self.frame_speed = 0.3
82        self.frame_images = self.frame_images
83        self.frame_index = 0
84        self.frame_speed = 0.3
85        self.frame_images = self.frame_images
86        self.frame_index = 0
87        self.frame_speed = 0.3
88        self.frame_images = self.frame_images
89        self.frame_index = 0
90        self.frame_speed = 0.3
91        self.frame_images = self.frame_images
92        self.frame_index = 0
93        self.frame_speed = 0.3
94        self.frame_images = self.frame_images
95        self.frame_index = 0
96        self.frame_speed = 0.3
97        self.frame_images = self.frame_images
98        self.frame_index = 0
99        self.frame_speed = 0.3
100       self.frame_images = self.frame_images

```

En cambio, ia.py da vida y movimiento al robot interactuable Robbie, que es la puerta para acceder a GPT-3 mediante el juego. Asimismo, tiene un método self.actions() que revisa si el jugador está cerca del robot para cambiar su animación a una más alegre y revisa si el jugador ha presionado T, la tecla para interactuar con Robbie.

Openai_api.py como su nombre ya deja caer, es el módulo que conecta nuestro juego con GPT-3 mediante el código que la misma OpenAI proporciona en la página web.

Consigue mantener una conversación constante y con contexto. Genera otra ventana por la cual podemos introducir texto para poder hablar con la inteligencia artificial, pudiendo también ver sus respuestas.



Tile.py, se encarga

de crear los diferentes sprites y escalarlos bien para que cumplan con la estética del juego. Los dota de una caja de colisiones para que puedan ser interaccionables con el personaje.



3.3 PANDEMIC

El resultado final del código aplicado a los sprites y tilesets es el siguiente, y quedará anexo un enlace para su descarga para poder probarlo y su distribución.





Después de más de 450 líneas de código, el juego es completamente funcional y tiene una buena experiencia de juego. Con esto, la tarea de ser familiar al usuario queda completada, pareciendo ser un juego normal sin nada extraño, pese a ser el portal para un gran modelo de lenguaje.

El juego ahora queda completamente terminado y listo para ser distribuido entre los más jóvenes de los nuestros y así poder hacer que sean más conscientes de ser conscientes.

CONCLUSIÓN

Como he podido comprobar, hay diferentes tipos de consciencia, la consciencia primaria, dominada por emociones primarias, tales como la rabia, la ira y el hambre, y consciencia secundaria, la que no sucumbe a esto.

Queda demostrado que la consciencia tuvo una gran importancia en nuestro proceso evolutivo, ya que nos permitió tener una escena mental, un presente recordado que nos permitió poder tomar decisiones contextuales y no simplemente respondiendo a un estímulo dado.

Y haciendo uso de GPT-3, se puede ver cómo puede llegar a simular consciencia, pudiendo ser este el techo de cristal, y es que de momento no hay nada que apunte a que una máquina pueda ser consciente. Los modelos de lenguaje simplemente saben cómo responder porque tienen un gran entrenamiento con cantidades ingentes de datos con los que analizan, cuál puede ser la mejor oración con la que contestarte.

Al fin y al cabo, si quieres que el robot parezca consciente, vas a hacer que lo parezca, ya que su función es cumplir tus órdenes, no tomar la iniciativa solo, siempre depende de una entrada de información. El día en que las máquinas puedan tomar la iniciativa por si solas de programarse para una tarea, deberíamos replantearnos seriamente estas preguntas.

Debido al turbulento tema que elegí para este trabajo y del que sabemos mucho, pero muy poco, no he podido llegar a grandes conclusiones ni a verdades definitivas. Y los textos que hay suelen ser muy densos de procesar mentalmente.

Valoro mucho el haber hecho este trabajo porque, se me dio la oportunidad de elegir un tema, que pese a haber sido un poco difícil, me he sentido cómodo en todo momento y con ganas de seguir, ya no solo por el trabajo, sino por amor al conocimiento. Creo que es una buena manera de poner a elegir a la gente algo que les guste para que puedan ir teniendo una idea de cómo sería si su vida girase en torno a eso, aunque solo haya sido un año. Y aunque he aprendido muchísimo redactando e informándome, seguiré leyendo textos que no dio tiempo a poder redactar bien.

REFERENCIAS DOCUMENTALES

Baars, Bernard. *A Cognitive Theory of Consciousness*. 1988

CNN. *Google despide a un ingeniero que afirmaba que su tecnología de IA era consciente*. Accessible a <https://cnnespanol.cnn.com/2022/07/24/google-despide-ingeniero-afirmaba-tecnologia-de-ia-era-consciente-trax/> Consulta: 25-06-2022

Crick, Francis. *La búsqueda científica del alma*. Madrid, 1994

[Herrnstein, R. J.](#) . *Acquisition, generalization, and discrimination reversal of a natural concept*. Accessible a <https://psycnet.apa.org/record/1980-20434-001>. Consulta: 25-06-2022

Iberdrola. *Descubre los principales beneficios del 'Machine Learning'* Accessible a <https://www.iberdrola.com/innovacion/machine-learning-aprendizaje-automatico> Consulta: 15-07-2022

IBM. *Neural Networks*. Accessible a <https://www.ibm.com/cloud/learn/neural-networks> Consulta: 28-09-2022

John Horgan. *Meta-Post: Posts on the Mind-Body Problem* . Accessible a <https://blogs.scientificamerican.com/cross-check/meta-post-posts-on-the-mind-body-problem> Consulta: 11-07-2022

RAE. *Inteligencia*. Accessible a <https://dle.rae.es/inteligencia> Consulta: 14-07-2022

SAP. *¿Qué es machine learning?* Accessible a <https://www.sap.com/spain/insights/what-is-machine-learning.html> Consulta: 28-09-2022

YeePLY. *Lenguajes de programación más usados según el tipo de desarrollo*. Accessible a <https://www.yeePLY.com/blog/lenguajes-de-programacion-mas-usados> Consulta: 28-09-2022

Unknown. *Francis Crick: "What Mad Pursuit: A Personal View of Scientific Discovery"*. Accessible a <http://230nsc1.phy-astr.gsu.edu/Nave-html/Faithpathh/Crick.html> Consulta: 25-06-2022

4r soluciones. *¿Qué es la inteligencia artificial y dónde podemos encontrarla?* Accessible a <https://www.4rsoluciones.com/blog/la-inteligencia-artificial-donde-podemos-encontrarla/#:~:text=En%20la%20actualidad%20encontramos%20muchos,que%20desarrollan%20Google%20y%20Tesla> Consulta: 07-07-2022

ANEXOS

Enlaces para descargar el juego interactivo, diseñado durante el trabajo:

<https://lasallecat->

my.sharepoint.com/:f:/g/personal/pablosmith_necpas_congres_lasalle_cat/EhkkztHIIZRBkEB_eUwvqDugBUJzbxgQfJLReFe65MyzzKQ?e=KzEbAj

<https://lasallecat->

my.sharepoint.com/:f:/g/personal/pablosmith_necpas_congres_lasalle_cat/EhkkztHIIZRBkEB_eUwvqDugBUJzbxgQfJLReFe65MyzzKQ?e=KzEbAj

