

Examen Final de GRAU-IA

(26 de junio de 2018)

Duración: 2 horas 40 minutos

1. (5 puntos) Una empresa del sector de los videojuegos quiere ir mucho más allá en la introducción de la Inteligencia Artificial en sus productos. Para su próximo videojuego quieren que la IA del juego adapte la historia, el ritmo y el tipo de juego al tipo de jugador que lo usa, de forma que tenga más o menos acción, más o menos aventura, más o menos misterio, centrar la historia en el jugador individual o modificarla para que sea un juego de estrategia por equipos, todo ello dentro del mismo juego. Para hacer esta adaptación el sistema dispone del historial de jugador, tanto dentro del juego como en otros juegos de la misma compañía, extrayendo de ahí el perfil de jugador.

De cada juego del catálogo de esta empresa tiene asociado un título, un año de publicación, la edad recomendada (+0 años, +3 años, +5 años, +8 años, +12 años, +18 años), si es un juego on-line o no, si es un juego multi-jugador y la saga de juegos a la que pertenece (de la que solo se guarda el título y los juegos que la componen). La empresa solo ofrece en estos momentos ocho tipos de juego: arcade, juegos de estrategia en tiempo real (más conocidos por las siglas RTS en inglés), juegos de lucha, de deportes, aventuras gráficas, puzzles, simuladores de mundo y simuladores de vehículos, y de todos ellos se guarda la misma información excepto en el caso de los simuladores de vehículos, en los que se guarda además el conjunto de vehículos que los jugadores pueden pilotar. Cada juego esta compuesto por uno o más niveles, y de cada nivel se guarda su nombre, la dificultad (de 1 a 10), el conjunto de niveles desde los que se puede acceder a ese nivel, y el tiempo máximo para poder superar el nivel (en segundos, si vale 0 significa que no hay límite de tiempo). Cada nivel está compuesto por uno o más espacios, y de cada uno se guarda su nombre, si es un espacio interior o exterior, el conjunto de espacios desde los que se puede acceder a ese espacio y el conjunto de objetos que estan situados dentro de ese espacio. De cada objeto se guarda su nombre y su precio en créditos (en moneda ficticia, el usuario puede gastar esos créditos en el mismo juego o en otros juegos de la misma compañía). Hay cinco tipos de objetos: las llaves, de las que se guarda además el espacio al que pueden dar acceso; las armas, de las que se guarda nivel de daño que pueden generar por ataque y la energía disponible (en %); los vehículos, que pueden ser aereos, espaciales, marinos, anfibios o terrestres, y de los que se guarda el nombre, el nivel de energía y el número de plazas; las herramientas, de las que se guarda solo el nombre y el precio, y los contenedores, de los que se guarda el conjunto de objetos que tienen dentro.

La participación de los jugadores en los juegos se guarda por partidas, y de cada partida se guarda la fecha y hora de inicio, la duración (en minutos), los niveles del juego completados por el usuario, los personajes que han participado y los equipos que han participado. En un juego hay uno o varios personajes, que pueden ser personajes jugador (personajes que corresponden a usuarios reales) y personajes no jugador (más conocidos por el acrónimo NPC, personajes que corresponden a jugadores virtuales que crea el juego). De los personajes jugador se guarda el nombre del personaje (texto), el usuario real asociado a este personaje su rol (texto), su salud (en %), los puntos obtenidos (entero), los creditos obtenidos (en la moneda ficticia) y el conjunto de objetos del juego que tiene (ya sea porque los encontró o porque los compró). De los personajes no jugador se guarda el nombre del personaje (texto), su rol (texto), su salud (en %), si es enemigo, y el nivel de inteligencia artificial (de 1 a 10). Tanto los personajes jugador como los no jugador pueden formar un equipo, del que se guarda su nombre y todos sus integrantes. De cada usuario se guarda el nombre de usuario, todas las partidas que ha jugado y el crédito total acumulado (en la moneda ficticia).

Los personajes pueden realizar acciones, y cada acción tiene asociado un nombre, la fecha y hora en la que se completó la acción, el tiempo de reacción del personaje a la hora de ejecutar la acción (en milisegundos), el conjunto de acciones de las que esta acción depende para poder ejecutarse, el espacio en el que se ha realizado la acción y el conjunto de objetos que requiere la acción para poder hacerla. Los personajes también pueden participar en un diálogo, y de los diálogos se guarda el identificador de diálogo y el conjunto de personajes que ha participado. Todos los juegos (incluso los puzzles de habilidad más simples) tienen una historia asociada, ligada a lo que el jugador va haciendo en el juego. La historia está compuesta por hitos argumentales, es decir puntos importantes dentro de la historia del juego a los que el jugador ha de llegar. Cada hito argumental ocurre en un nivel del juego (un nivel puede tener varios hitos asociados) y tiene asociada una acción objetivo que el jugador ha de conseguir realizar. La información que se guarda para cada hito incluye el tema (una misión individual, una misión grupal, una relación personal, un misterio), si el hito es central (es decir, imprescindible para el juego) o si es opcional (se puede añadir o eliminar sin que el juego pierda sentido), la acción a realizar, la fecha y hora límite para realizar la acción (todo a ceros si no hay límite temporal). Todos los hitos argumentales tienen asociadas tres escenas pre-grabadas de animación 3D: la escena de inicio (un cortometraje donde se narra parte de la historia y le dan pistas al jugador de la acción objetivo que ha de realizar) la escena de objetivo cumplido (se confirma al jugador que ha conseguido el objetivo del hito, y se cuenta un trozo de historia que es consecuencia del éxito) y

la de objetivo no cumplido (un trozo de historia consecuencia del fracaso), de estas escenas solo se guarda el identificador del video y la duración de la escena. Además un hito argumental puede incluir uno o más diálogos (que se insertan en el juego para guiar al usuario en cierta dirección, o darle pistas).

Los servidores de la empresa recopilan toda esta información y la complementan con otros estadísticos generados de forma automática, como el tiempo medio de reacción del jugador (en milisegundos) en las acciones de la partida actual / en el día actual / en la última semana / en el último mes; o el tiempo extra (en segundos) que el usuario dedica a explorar los espacios en la partida actual / en el día actual / en la última semana / en el último mes, para las escenas animadas el porcentaje de video (en media) que ve el usuario en la partida actual / en el día actual / en la última semana / en el último mes (hay usuarios que verán normalmente los videos enteros, otros que se saltarán el video para ir a la acción, pero esto puede variar por días, según si el usuario tiene tiempo para verlos), etc.

Por lo que nos dicen los expertos se suelen usar varias características para construir un perfil de jugador:

- La **salud** del personaje en la partida actual, que puede ser muy baja ($< 10\%$), baja ($< 40\%$), normal ($< 80\%$) o elevada ($\geq 80\%$).
- El **tiempo de reacción** medio de las acciones en la partida actual, que puede ser rápido (< 0.4 seg), normal (< 0.9 seg) o lento (≥ 0.9 seg).
- El **tiempo de exploración** medio en las partidas del día actual, que puede ser mínimo (< 5 seg), normal (< 50 seg) o elevado (≥ 50 seg).
- El **tiempo de narración** medio que admite el usuario en las escenas animadas del día actual, que puede ser mínimo ($< 3\%$), bajo ($< 20\%$), normal ($< 80\%$) o elevado ($\geq 80\%$).
- El **nivel de trabajo en equipo**, que está asociado al porcentaje de partidas jugadas e el último mes en las que el usuario ha participado dentro de un equipo, y que puede ser muy bajo ($< 10\%$), bajo ($< 40\%$), normal ($< 80\%$) o elevado ($\geq 80\%$).
- La **interacción con otros personajes**, que está asociada al número de diálogos en los que el usuario ha participado en el último mes, y que puede ser muy baja (< 5), baja (< 20), normal (< 30) o elevada (≥ 30).
- La **tipología de juegos** que el usuario ha jugado en el último año, que puede ser variada (un poco de todos los tipos de juego) mayormente de acción (si la mayoría de partidas jugadas son de juegos de tipo arcade o lucha), mayormente de ingenio (si la mayoría de juegos son de tipo puzzle, aventura gráfica o simulador de mundo), o mayormente competitivo (si la mayoría de juegos son de tipo deporte o simulador de vehículo).

A partir de estas características queremos adaptar la partida actual modificando algunos de sus parámetros para ajustar la experiencia de juego a lo que el usuario parece necesitar o preferir:

- *Ajustar el número de enemigos*: si la salud del personaje es alta o el tiempo de reacción es rápido y la tipología de juegos es mayormente de acción o mayormente competitivo entonces hay que incrementar el número de NPCs enemigos; si la salud es normal y el tiempo de reacción es normal o alto y la tipología de juegos es mayormente de acción entonces hay que incrementar el número de NPCs enemigos; si la salud es muy baja o si el tiempo de reacción es lento entonces hay que reducir el número de NPCs enemigos; en los otro casos no modificar los enemigos.
- *Ajustar el nivel de la IA de los enemigos*: si la salud del personaje es alta o el tiempo de reacción es rápido y la tipología de juegos es mayormente de aventuras o mayormente competitivo entonces incrementar el nivel de la IA de los NPCs enemigos; si la salud es normal y el tiempo de reacción es normal o alto y la tipología de juegos es mayormente competitivo entonces hay que incrementar el nivel de la IA de los NPCs enemigos; si la salud es muy baja o si el tiempo de reacción es lento entonces hay que reducir el nivel de la IA de los NPCs enemigos; en los otro casos no modificar la IA de los enemigos.
- *Ajustar misiones de equipo*: si el nivel de trabajo en equipo es bajo y el tiempo de narración es normal o elevado entonces introducir hitos argumentales con tema de misión grupal; si el nivel de trabajo en equipo es elevado y el tiempo de narración es elevado entonces introducir hitos argumentales con tema misión individual; si el tiempo de narración es mínimo entonces eliminar todos los hitos argumentales con tema misión individual o misión grupal que no sean centrales; en el resto de casos no variar.
- *Ajustar misterios*: si la tipología de juegos es mayormente de ingenio y el tiempo de narración es elevado o el tiempo de exploración es elevado entonces introducir hitos argumentales con tema misterio; si el tiempo de narración es bajo o el tiempo de exploración es bajo entonces reducir hitos argumentales con tema misterio; si el tiempo de narración es mínimo entonces eliminar todos los hitos argumentales con tema misterio que no sean centrales; en el resto de casos no variar.
- *Ajustar relaciones*: si el nivel de interacción con otros personajes es bajo o muy bajo y el tiempo de narración es normal o elevado y el tiempo de exploración es normal o elevado entonces introducir hitos argumentales con tema relación personajes; si el nivel de interacción con otros personajes es normal y el tiempo de narración es bajo entonces reducir hitos argumentales con tema relación personajes; si

el tiempo de narración es mínimo entonces eliminar todos los hitos argumentales con tema relación personajes que no sean centrales; en el resto de casos no variar.

La salida de este sistema de adaptación ha de sugerir exactamente que modificaciones hay que introducir en el juego (número de enemigos a añadir/eliminar, niveles de IA que aumentar/disminuir, que hitos hay que añadir o eliminar, etc...).

- a. (2 puntos) Diseña la ontología del dominio descrito, incluyendo todos los conceptos que aparecen en la descripción e identificando los atributos más relevantes. Lista que conceptos forman parte de los datos de entrada del problema y que conceptos forman parte de la solución. (Nota: tened en cuenta que la ontología puede necesitar modificaciones para adaptarla al apartado siguiente).
 - b. (2 puntos) El problema descrito es un problema de análisis. Explica cómo lo resolverías usando clasificación heurística, usando los conceptos de la ontología desarrollada en el apartado anterior. Da al menos 4 ejemplos de reglas para cada una de las fases de esta metodología.
 - c. (1 punto) Las características que se usan durante el proceso de adaptación del videojuego no son independientes entre sí. Los tiempos de reacción suelen influir en la salud del personaje (con tiempos de reacción lentos la salud suele empeorar); el tiempo de exploración suele influir en el tiempo de narración (un usuario que dedica mayor tiempo a explorar el entorno suele tener más paciencia o curiosidad por ver los videos enteros, y por ello el tiempo de narración es mayor, mientras que un tiempo de exploración menor suele asociarse a un jugador con menos tiempo disponible o menos curiosidad, y por lo tanto tener un tiempo de narración menor); el nivel de trabajo en equipo influye en la interacción con otros personajes (a mayor trabajo en equipo, mayor interacción con otros personajes). Define el problema de asociación heurística como una red bayesiana expresando en ella al menos las relaciones indicadas no solo en este apartado sino en el resto del enunciado, de forma que todas las características abstractas del problema que hayas definido en el apartado anterior tengan algún tipo de influencia en la solución. Separa bien en el diagrama que variables describen características de problema y cuales describen soluciones. Lista de forma clara los diferentes valores que puede tomar cada variable. Da un ejemplo de tabla de probabilidad de algún nodo, inventándote las probabilidades, pero expresando como influyen los valores de los nodos padre en las probabilidades de los valores de los nodos hijo.
2. (5 puntos) La empresa *TraeMeLo* quiere convertirse en la líder dentro del sector emergente de empresas que reciben pedidos de sus clientes para ir a recoger algún producto/compra en uno de los comercios registrados en el sistema (normalmente una tienda, una farmacia, un restaurante) y traérselo a sus casas, siempre que sea dentro de la misma ciudad, no exceda los 10kg de peso y con un volumen de máximo 40x40x30cm. Los pedidos los llevan unos mensajeros, que pueden ir en bici (con un cajón trasero en el que pueden llevar un máximo de 2 pedidos a la vez) o en scooter (con un cajón trasero en el que pueden llevar un máximo de 4 pedidos a la vez).

Los clientes pueden solicitar a través de una app la recogida de un pedido en un comercio registrado (la app envía un identificador de comercio, y el sistema tiene un método auxiliar para convertir ese string en coordenadas de GPS) y la entrega en la dirección registrada del cliente (hay que tener en cuenta que un cliente puede realizar más de un pedido, y en una misma dirección puede haber más de un cliente). El sistema entonces queda a la espera a que el comercio envíe un aviso de que el pedido está listo para ser recogido, y en ese momento asigna alguno de los mensajeros que tenga espacio dentro del cajón trasero para recoger el pedido en el comercio y llevarlo a su destino, pudiendo parar por el camino en otros puntos para recoger o entregar pedidos. La empresa nos pide que desarrollemos un planificador simple que genere un plan para recoger y entregar los pedidos recibidos en un periodo de tiempo con los mensajeros disponibles en cada momento (a veces una avería o una baja por enfermedad hace que ese número varíe). En el modelo deben quedar explícitos los pedidos a recoger, el aviso del comercio de que el pedido está listo, la recogida del pedido por parte del mensajero y su entrega al cliente.

- a. (3.5 puntos) Describe el dominio (incluyendo predicados, acciones, etc...) usando PDDL. Da una explicación razonada de los elementos que has escogido. Ten en cuenta que el modelo del dominio ha de poderse extender a más o menos mensajeros, más o menos comercios, más o menos clientes y a futuros vehículos capaces de llevar más pedidos a la vez.
- b. (1.5 puntos) El jefe del proyecto nos ha proporcionado una tabla como ejemplo de las peticiones que el sistema recibirá en un periodo de 20 minutos, con una serie de pedidos que se han de transportar desde un comercio registrado hasta la dirección del cliente. También nos dice que al inicio de ese intervalo de 20 minutos tenemos disponibles dos motos (capacidad máxima de 4 pedidos) y una bici (capacidad máxima de 2 pedidos) estacionadas en el parking central de *TraeMeLo*, y que tras entregar todos los pedidos del periodo las motos y la bici han de volver a ese parking.

pedido	dirección recogida	dirección entrega	cliente
p1	PizzaHotBalmes88	PauClarís55	Mario
p2	TodoPastaAragon167	Meridiana131	Aitana
p3	SupermercadoBruc31	Villaroel21	Pau
p4	frankfurtCasp13	ManuelGirona86	Amalia
p5	MercahoyValencia216	RamblaPrim12	Jaume
p6	Farmacia24hDiagonal340	Meridiana131	Aitana
p7	TodoPastaAragon167	Lepant95	Nuria
p8	FarmaciaGarciaSardenya501	Lepant95	David
p9	PizzaHotBalmes88	RocBoronat01	Esther
p10	BurgerFastCiencies62	AvCarrilet73	Miquel
p11	Farmacia24hDiagonal340	AvCarrilet73	Miquel
p12	MercahoyValencia216	RamblaPrim12	Laura

Describe este problema usando PDDL. Da una breve explicación de cómo modelas el problema.

Las notas se publicarán el día **4 de julio**.