

Índice

| | |
|---|----------|
| 1. CUESTIONES | 1 |
| 1.1. El concepto de juego. Elementos y clasificación de los juegos. | 1 |
| 1.2. El algoritmo minmax. Componentes y funcionamiento. | 1 |
| 1.3. Modelos de representación del conocimiento. | 1 |
| 1.4. Modelos de conocimiento heredable. Herencia. | 1 |
| 1.5. Paradigmas de aprendizaje. | 1 |
| 1.6. Aprendizaje inductivo. Empleo de árboles de decisión en aprendizaje inductivo. | 1 |
| 2. RESPUESTAS | 2 |
| 2.1. El concepto de juego. Elementos y clasificación de los juegos. | 2 |
| 2.2. El algoritmo minmax. Componentes y funcionamiento. | 3 |
| 2.3. Modelos de representación del conocimiento. | 4 |
| 2.4. Modelos de conocimiento heredable. Herencia. | 5 |
| 2.5. Paradigmas de aprendizaje. | 6 |
| 2.6. Aprendizaje inductivo. Empleo de árboles de decisión en aprendizaje inductivo. | 7 |

1 CUESTIONES

- 1.1. El concepto de juego. Elementos y clasificación de los juegos.**
- 1.2. El algoritmo minmax. Componentes y funcionamiento.**
- 1.3. Modelos de representación del conocimiento.**
- 1.4. Modelos de conocimiento heredable. Herencia.**
- 1.5. Paradigmas de aprendizaje.**
- 1.6. Aprendizaje inductivo. Empleo de árboles de decisión en aprendizaje inductivo.**

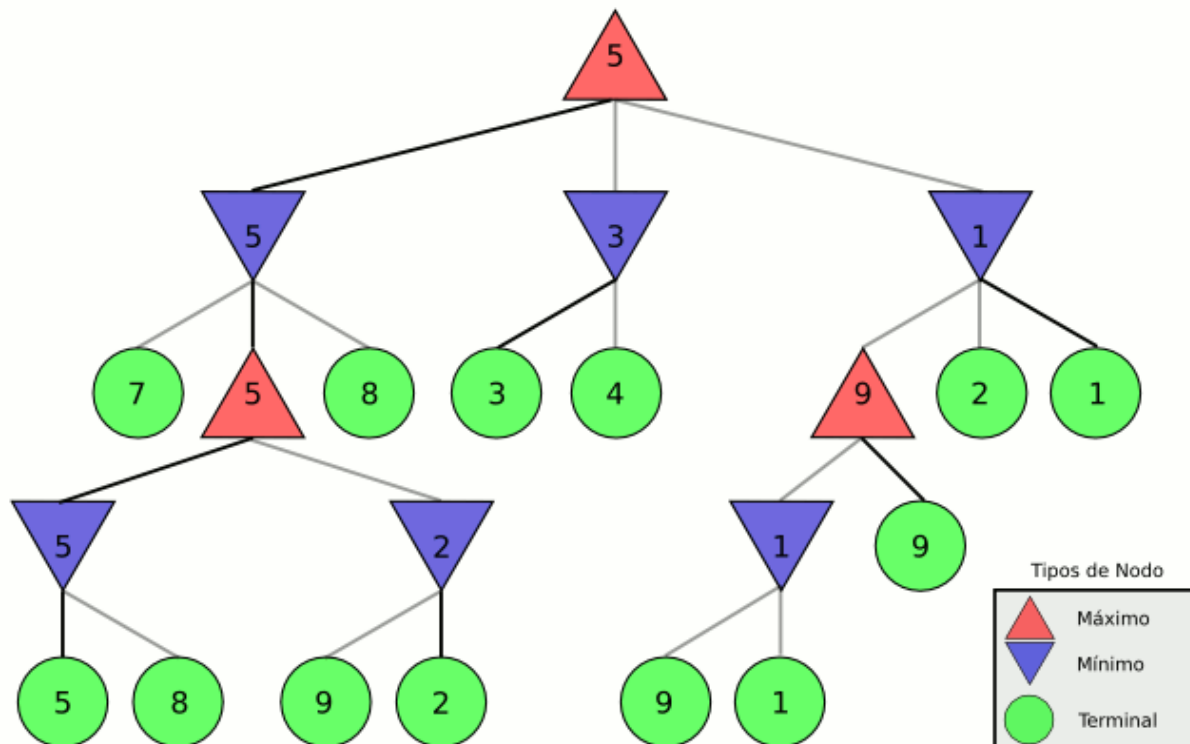
2 RESPUESTAS

2.1. El concepto de juego. Elementos y clasificación de los juegos.



Un juego es cualquier situación de decisión con **varios agentes** (llamados jugadores) gobernada por un **conjunto de reglas** y con un resultado bien definido, caracterizada porque **ninguno de los jugadores con su sola actuación puede determinar el resultado** (independencia estratégica). Lo que caracteriza a un juego es el **número de jugadores**, la **información con que cuentan los jugadores** (perfecta o imperfecta), la **existencia o no de movimientos de azar**, el **orden de actuación de los jugadores**, los **repartos de beneficios** (que determinan si se trata de juegos de suma nula o no nula) y la **existencia o no de pagos colaterales** (equilibrio de Nash). Respecto a los elementos de un juego: intervienen, además de los jugadores, las distintas estrategias que estos pueden seguir (que están condicionada por las reglas que rigen el desarrollo del juego) y los pagos o beneficios que obtendrá cada jugador en función de la estrategia que utilice cada uno. (a cada tupla de estrategias, una por jugador, le corresponde una tupla de beneficios obtenidos, uno por jugador).

2.2. El algoritmo minmax. Componentes y funcionamiento.



El algoritmo minmax es un algoritmo usado para tomar decisiones óptimas en **juegos bipersonales competitivos**. Los componentes del algoritmo son:

Los jugadores: a los que se denominará Max y Min, siendo Max el jugador que comienza la partida.

Estado inicial: que identifica la situación en que comienza el juego y determina las posibles estrategias de los jugadores

Función sucesor: determina, para cada movimiento legal, el estado al que conduce.

Función de utilidad: determina los resultados de la partida en los nodos terminales. Los resultados favorables a Max serán positivos y los favorables a Min, negativos. En caso de empate el resultado será 0.

Función terminal: determina si un nodo es o no terminal (si supone el fin de la partida).

El algoritmo calcula la acción que conduce al mejor resultado para el jugador. En el caso de Max conduce al mayor valor posible y en el caso de Min, al menor (que decir al peor resultado para Max y el mejor para Min) Para ello, genera un **árbol de juego** con todos los estados posibles que resultan de las distintas acciones de los jugadores hasta obtener los nodos terminales, calcula los valores de dichos nodos terminales utilizando la función de utilidad y los de los nodos no terminales utilizando los valores de sus hijos. El algoritmo asigna a cada nivel del árbol un rol Max o Min en función de a qué jugador le corresponde actuar y asigna el mayor (o menor) valor de los nodos de los hijos dependiendo de si se trata de una decisión que toma Max o Min (Max pretende maximizar el valor de los nodos y Min minimizarlo). Se trata de una exploración primero en profundidad en la que se debe explorar todo el árbol por lo que el tiempo necesario para calcular el árbol de juego de un caso complejo puede ser inabarcable. Por esta razón, el algoritmo de por sí no nos resulta útil. No obstante, es la base de algoritmos más complejos como la poda alpha-beta.

2.3. Modelos de representación del conocimiento.

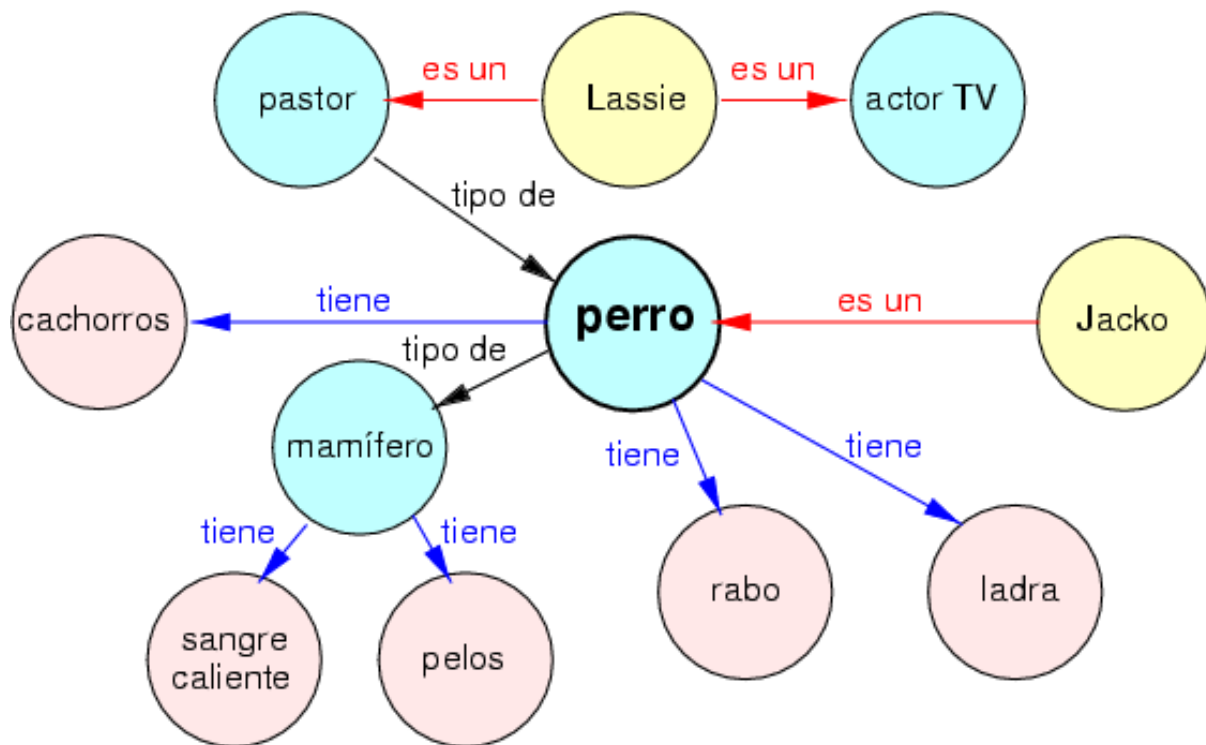


Un modelo de representación del conocimiento es un formalismo simbólico empleado para expresar el conocimiento, un modelo mental que se caracteriza por la semántica con que representa ciertos aspectos del dominio o universo con el que un sistema software pueda trabajar. Entre sus características está aquello que puede representar (representación icónica o representación descriptiva), la granularidad de la representación (nivel de detalle), el tipo de descripciones (intensionales o extensionales), la representación del metaconocimiento (conocimiento sobre el propio conocimiento, independiente del mismo) y la jerarquización de objetos y herencia. Así, encontramos modelos inspirados por la psicología como:

- Los modelos de conducta o reglas de producción, que son reglas de inferencia del tipo si ocurre A, entonces ocurre B.
- Los modelos de razonamiento o lógicas, que son lenguajes formales con símbolos, semántica y sintaxis que permiten representar el conocimiento y formular inferencias que permitan obtener nuevo conocimiento a partir de conocimiento previo gracias a la teoría de la demostración, siendo la lógica de proposiciones y la lógica de predicados las dos más simples.
- Los modelos de memoria, que representan el conocimiento como conjuntos de elementos y relaciones entre ellos. Bien con modelos de conocimiento relacional simple (ficheros y bases de datos, con relaciones expresables mediante tablas) que requerirán de un motor de inferencia para generar nuevo conocimiento, o bien con modelos de conocimiento heredable (que jerarquizan el conocimiento), utilizando redes asociativas (semánticas, de clasificación o causales), marcos, frames, representación orientada a objetos...

Además, existen otros modelos y problemas en la representación del conocimiento como la necesidad de representación del sentido común, la organización jerárquica del conocimiento, el razonamiento temporal o la lógica difusa.

2.4. Modelos de conocimiento heredable. Herencia.



Los modelos de conocimiento heredable se basan en la jerarquización del conocimiento de forma que se puedan inferir características o atributos de ciertos elementos a partir de otras ya conocidas. Por ejemplo, un marco o frame es una estructura de datos formada por un conjunto de campos que se asocian por lo general a atributos que sirven para identificar los marcos. Se utilizan para tareas de reconocimiento ya que la información contenida en un marco puede hacer que se activen otros marcos conectados con los primeros, dando lugar así a una red de activación que permite predecir y explicar la información que deriva de los marcos activados inicialmente. Al conocimiento que surge de este mecanismo se le suele denominar herencia o reconocimiento descendiente porque se determinan propiedades desconocidas de ciertos elementos a partir de otras ya conocidas que las encapsulan. Por ejemplo, si un frame 'pájaro' contiene el campo 'movimiento: volar', y otro frame, 'periquito', contiene el campo 'tipo: pájaro' entonces se puede inferir que el periquito puede volar a no ser que el frame 'periquito' contenga un campo 'movimiento' con un valor distinto a 'volar'.

Por otro lado, las llamadas redes asociativas están formadas por nodos que representan un concepto o una proposición, enlazados entre sí de forma que se representan relaciones de inclusión, pertenencia o causalidad o bien a categorías gramaticales (verbo, sujeto, complemento, etc.) Algunos tipos de redes asociativas son:

- redes semánticas: son las destinadas a representar o a comprender el lenguaje natural.
- redes de clasificación: clasificaciones de objetos o conceptos según sus características propias (con herencia).
- redes causales: son las que llevan asociadas una relación de influencia o causa entre los nodos que las forman (el hecho representado por el nodo A provoca el hecho representado por el nodo B y por eso ambos nodos están relacionados.)

2.5. Paradigmas de aprendizaje.



En el campo del aprendizaje automático encontramos, principalmente, los siguientes paradigmas de aprendizaje:

Aprendizaje memorístico surge cuando el aprendizaje consta de asociaciones arbitrarias o cuando el sujeto lo hace de forma arbitraria. Los datos se almacenan sin tratar de comprenderlos o de inferirlos a partir de otros ya conocidos y de forma repetitiva. Por ejemplo, si un sistema recibe, ante una determinada acción en cierta circunstancia, una respuesta desventajosa, puede ‘memorizar’ que esa acción da malos resultados y, si esos malos resultados se repiten en el tiempo, concluir (aprender) que esa no es una acción adecuada para dicha situación.

Aprendizaje deductivo consiste en aplicar la deducción para obtener descripciones generales a partir de un ejemplo de concepto y su explicación. Es decir, es la obtención de nuevo conocimiento a partir de conocimientos que ya se poseen. También es el razonamiento artificial (dar una explicación a un evento observado a partir de conocimientos previos y crear así nuevo conocimiento.)

Aprendizaje analítico, Basado en explicaciones Construir una explicación para cada ejemplo en relación con un concepto dado y generalizar la explicación de modo que pueda emplearse en el futuro. Los métodos analíticos usan el conocimiento primario para deducir las hipótesis generales. Aquí, se buscan hipótesis generales que se ajusten al conocimiento primario mientras se cubren los datos observados.

Aprendizaje analógico se buscan soluciones a problemas nuevos encontrando similitudes con problemas ya conocidos y adaptando sus soluciones. Se basa en la idea de que, si dos situaciones son similares en algún aspecto, entonces también pueden serlo en otros.

Aprendizaje inductivo se trata de aprender un concepto o una clasificación a partir de ejemplos y contraejemplos. Se formulan hipótesis mediante la búsqueda de regularidades en unos ejemplos de ‘entrenamiento’ observados previamente (ejemplos) y se aceptan o rechazan dichas hipótesis con la aparición de nuevos ejemplos.

Tipos de aprendizaje según su conocimiento:

Supervisado Se dispone de un profesor/supervisor que proporciona una salida deseable para cada entrada percibida, ya sea una clase o un valor a aproximar (clasificación vs regresión).

No supervisado No se dispone de una salida deseada cada entrada, sino que se busca agrupar/clasificar los datos en función de ciertas características (medida de distancia)

Refuerzo Se aprende (sin supervisor) a partir de la información obtenida al realizar procesos de ensayo-error en los que se obtienen “señales” de beneficio/coste.

2.6. Aprendizaje inductivo. Empleo de árboles de decisión en aprendizaje inductivo.



Se trata de aprender un concepto o una clasificación a partir de ejemplos y contraejemplos. El objetivo es aprender la función f . Un ejemplo es un par $(x, f(x))$. Dada una colección de ejemplos de f , devolver una función h (hipótesis) que aproxime a f . La razón por la cual el aprendizaje es difícil es porque no es fácil determinar si una función h es una buena aproximación de f . Una buena hipótesis estará bien generalizada si puede predecir ejemplos que no se conocen.

Un árbol de decisión se puede utilizar para el aprendizaje inductivo si toma como entrada un objeto o una situación descrita a través de un conjunto de atributos y devuelve una decisión (valor previsto de la salida dada por la entrada). Los atributos de entrada y salida pueden ser discretos o continuos. Aprender una función de valores discretos se denomina clasificación. Aprender una función continua se denomina regresión. Resumiendo, un árbol de decisión, desarrolla una secuencia de tests para poder alcanzar una decisión. Cada nodo corresponde con un test sobre el valor de una de las propiedades, y las ramas que salen de cada nodo están etiquetadas con los posibles valores de dicha propiedad. Cada nodo hoja representa el valor que ha de ser devuelto si dicho nodo hoja es alcanzado.