

19-20

GRADO EN INGENIERÍA EN
TECNOLOGÍAS DE LA INFORMACIÓN
SEGUNDO CURSO

GUÍA DE ESTUDIO COMPLETA



FUNDAMENTOS DE INTELIGENCIA ARTIFICIAL

CÓDIGO 71902060

UNED

19-20

FUNDAMENTOS DE INTELIGENCIA
ARTIFICIAL

CÓDIGO 71902060

ÍNDICE

PRESENTACIÓN Y CONTEXTUALIZACIÓN
REQUISITOS Y/O RECOMENDACIONES PARA CURSAR LA
ASIGNATURA
EQUIPO DOCENTE
HORARIO DE ATENCIÓN AL ESTUDIANTE
TUTORIZACIÓN EN CENTROS ASOCIADOS
COMPETENCIAS QUE ADQUIERE EL ESTUDIANTE
RESULTADOS DE APRENDIZAJE
CONTENIDOS
METODOLOGÍA
PLAN DE TRABAJO
SISTEMA DE EVALUACIÓN
BIBLIOGRAFÍA BÁSICA
BIBLIOGRAFÍA COMPLEMENTARIA
RECURSOS DE APOYO Y WEBGRAFÍA
GLOSARIO
TUTORIZACIÓN EN CENTROS ASOCIADOS

Nombre de la asignatura	FUNDAMENTOS DE INTELIGENCIA ARTIFICIAL
Código	71902060
Curso académico	2019/2020
Departamento	INTELIGENCIA ARTIFICIAL
Título en que se imparte	GRADO EN INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN - TIPO: OBLIGATORIAS - CURSO: SEGUNDO CURSO / MÁSTER UNIVERSITARIO EN INGENIERÍA Y CIENCIA DE DATOS (complemento) / MÁSTER UNIVERSITARIO EN INGENIERÍA INFORMÁTICA (complemento)
Nº ETCS	6
Horas	150.0
Periodo	SEMESTRE 2
Idiomas en que se imparte	CASTELLANO

PRESENTACIÓN Y CONTEXTUALIZACIÓN

.Esta asignatura tiene como objetivo que el alumno conozca la historia, las bases y la metodología de la inteligencia artificial. Para ello se centra preferentemente en el estudio de dos técnicas ampliamente utilizadas en inteligencia artificial: por un lado, la búsqueda en un espacio de estados y, por otro lado, la representación de conocimiento y el razonamiento con el mismo. La asignatura es de carácter obligatorio, consta de 6 créditos y pertenece al segundo semestre del segundo curso tanto del Grado en Ingeniería Informática como del Grado en Ingeniería en Tecnologías de la Información.

La presente asignatura está relacionada con un conjunto de asignaturas de cursos posteriores, las cuales profundizan en el estudio de diferentes áreas de la inteligencia artificial. La materia aquí explicada sienta las bases para el entendimiento de conceptos más avanzados que se incluyen en el temario de dichas asignaturas. Se citan a continuación las asignaturas con un mayor grado de relación con la actual, todas ellas de cuarto curso.

Grado en Ingeniería Informática:

Ampliación de Sistemas Inteligentes
Aprendizaje Automático
Robótica Autónoma
Visión Artificial

Grado en Ingeniería en Tecnologías de la Información:

Ingeniería y Gestión del Conocimiento
Minería de Datos
Modelos Probabilistas y Análisis de Decisiones

REQUISITOS Y/O RECOMENDACIONES PARA CURSAR LA ASIGNATURA

.Al tratarse de una asignatura de carácter introductorio, no se precisa ningún requisito específico para abordar el estudio de la materia, aparte de los conocimientos que el alumno debería ya poseer en este nivel de enseñanza. Es recomendable, no obstante, haber cursado ya las asignaturas de "Lógica y Estructuras Discretas" y de "Autómatas, Gramáticas y Lenguajes".

EQUIPO DOCENTE

Nombre y Apellidos
Correo Electrónico
Teléfono
Facultad
Departamento

ANGELES MANJARRES RIESCO
amanja@dia.uned.es
91398-8125
ESCUELA TÉCN.SUP INGENIERÍA INFORMÁTICA
INTELIGENCIA ARTIFICIAL

Nombre y Apellidos
Correo Electrónico
Teléfono
Facultad
Departamento

SEVERINO FERNANDEZ GALAN (Coordinador de asignatura)
seve@dia.uned.es
91398-7300
ESCUELA TÉCN.SUP INGENIERÍA INFORMÁTICA
INTELIGENCIA ARTIFICIAL

HORARIO DE ATENCIÓN AL ESTUDIANTE

El alumno dispone de los siguientes medios para facilitar el proceso de aprendizaje en esta asignatura:

- (1) **Curso virtual de la asignatura** desde el que se puede plantear cualquier duda sobre la asignatura y acceder a los recursos de apoyo al estudio.
- (2) **Tutorías** virtuales o presenciales en el centro asociado correspondiente
- (3) **Horario de guardia del equipo docente** en el que atenderá dudas a través del correo electrónico, del teléfono o presencialmente. Los datos de contacto con el equipo docente y los horarios de guardia son los siguientes:

Ángeles Manjarrés Riesco

Dpto. de Inteligencia Artificial (Despacho: 3.08)

E.T.S.I. Informática (UNED)

C/ Juan del Rosal, nº 16

28040 Madrid (España)

Tfno.: +34 91 3988125

Correo-e: amanja@dia.uned.es

Horario de guardia: jueves lectivos, de 10:15hrs a 14:15 hrs.

Horario de asistencia al estudiante: lunes y miércoles lectivos, de 10:15 hrs. a 14:15 hrs.

Severino Fernández Galán

Dpto. de Inteligencia Artificial (Despacho: 3.23)

E.T.S.I. Informática (UNED)

C/ Juan del Rosal, nº 16

28040 Madrid (España)

Tfno.: +34 91 3987300

Correo-e: seve@dia.uned.es

Horario de guardia: lunes lectivos, de 16 hrs. a 20 hrs.

Horario de asistencia al estudiante: martes y miércoles lectivos, de 16 hrs. a 20 hrs.

TUTORIZACIÓN EN CENTROS ASOCIADOS

En el enlace que aparece a continuación se muestran los centros asociados y extensiones en las que se imparten tutorías de la asignatura. Estas pueden ser:

- **Tutorías de centro o presenciales:** se puede asistir físicamente en un aula o despacho del centro asociado.
- **Tutorías campus/intercampus:** se puede acceder vía internet.

La información ofrecida respecto a las tutorías de una asignatura es orientativa. Las asignaturas con tutorías y los horarios del curso actual estarán disponibles en las fechas de inicio del curso académico. Para más información contacte con su centro asociado.

Consultar horarios de tutorización de la asignatura 71902060

COMPETENCIAS QUE ADQUIERE EL ESTUDIANTE

Esta asignatura facilita la adquisición por parte del alumno, en mayor o menor medida, de un conjunto de competencias que se pueden dividir en dos tipos principales:

(1) **Competencias generales:**

(G.2) Competencias cognitivas superiores: selección y manejo adecuado de conocimientos, recursos y estrategias cognitivas de nivel superior apropiados para el afrontamiento y resolución de diversos tipos de tareas/problemas con distinto nivel de complejidad y novedad: Análisis y Síntesis. Aplicación de los conocimientos a la práctica Resolución de problemas en entornos nuevos o poco conocidos. Pensamiento creativo. Razonamiento crítico. Toma de decisiones.

(G.4) Competencias de expresión y comunicación (a través de distintos medios y con distinto tipo de interlocutores): Comunicación y expresión escrita. Comunicación y expresión oral. Comunicación y expresión en otras lenguas (con especial énfasis en el inglés). Comunicación y expresión matemática, científica y tecnológica (cuando sea requerido y estableciendo los niveles oportunos).

(G.5) Competencias en el uso de las herramientas y recursos de la Sociedad del Conocimiento: Manejo de las TIC. Competencia en la búsqueda de información relevante. Competencia en la gestión y organización de la información. Competencia en la recolección de datos, el manejo de bases de datos y su presentación.

(2) **Competencias específicas:**

(BC.6) Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.

(BC.7) Conocimiento, diseño y utilización de forma eficiente de los tipos y estructuras de datos más adecuados a la resolución de un problema.

(BC.15) Conocimiento y aplicación de los principios fundamentales y técnicas básicas de los sistemas inteligentes y su aplicación práctica.

RESULTADOS DE APRENDIZAJE

Esta asignatura permite alcanzar los siguientes logros a la finalización del curso académico:

(RA1) Adquirir una perspectiva global de qué es la inteligencia artificial.

(RA2) Conocer la historia de la inteligencia artificial.

(RA3) Aplicar búsqueda en un espacio de estados a la resolución de problemas.

(RA4) Utilizar las ampliaciones de la lógica clásica que resultan necesarias en inteligencia artificial.

(RA5) Entender el funcionamiento de un sistema basado en reglas.

(RA6) Caracterizar los distintos tipos de redes semánticas utilizadas para representar conocimiento en inteligencia artificial.

(RA7) Usar marcos como método de representación de conocimiento estructurado y como método de inferencia.

CONTENIDOS

Bloque 1: Aspectos Conceptuales de la Inteligencia Artificial y la Ingeniería del Conocimiento

(Coincide con el capítulo 1 del texto base de la asignatura.)

En el capítulo 1 del texto base se introduce el concepto de inteligencia artificial como ciencia e ingeniería, se relata brevemente su historia, destacando el papel que juega en el contexto de las ciencias cognitivas, las ciencias de la computación y la ingeniería informática; y se presenta el panorama general de los diferentes paradigmas, técnicas, métodos y áreas de aplicación.

Estos contenidos introductorios permiten situar el resto de temas de la asignatura en su contexto histórico y comprender su relevancia en el ámbito global de la inteligencia artificial. En coherencia con la visión histórica presentada, en esta guía se sugiere una secuenciación de estudio de los capítulos del texto base diferente a la que el libro ofrece. Por otra parte, se introducen también áreas de estudio de la inteligencia artificial que serán abordadas en profundidad en asignaturas del cuarto curso, en particular: Ampliación de Sistemas Inteligentes, Aprendizaje Automático, Robótica Autónoma y Visión Artificial (del Grado en Ingeniería Informática); e Ingeniería y Gestión del Conocimiento, Minería de Datos, Modelos Probabilistas y Análisis de Decisiones (del Grado en Ingeniería en Tecnologías de la Información).

Los objetivos de aprendizaje cubiertos por este capítulo son pues objetivos de particular naturaleza, que no implican el desarrollo de competencias procedimentales técnicas, ni el estudio memorístico de conocimientos. Su propósito es presentar una amplia visión del campo desde múltiples perspectivas, sentar principios fundamentales, y señalar cuestiones candentes y ampliamente debatidas entre los profesionales del campo.

A continuación destacamos las cuestiones más relevantes abordadas en las diferentes secciones del capítulo con el fin de ayudar al alumno a centrar su estudio.

Concepto de inteligencia artificial (Secciones 1.1, 1.2, 1.5 y 1.6).

1. Objetivos como ciencia y como ingeniería del conocimiento:

- Papel y singularidades en el contexto de la ingeniería informática y las ciencias de la computación.
- Papel en el contexto de las ciencias cognitivas: conexiones con la neurofisiología, la psicología, la sociología y la filosofía.
- El aprendizaje como adaptación a un medio.
- Razonamiento, percepción y acción.

1. Hipótesis fuerte y débil.

2. Aproximaciones dominantes: simbólica o representacional, conexionista, situada e híbrida.

3. Principales áreas y campos de aplicación.

4. Los éxitos.

5. Las limitaciones:

- La imprecisión del concepto de inteligencia.
- Ausencia de una teoría completa del conocimiento y de la neurofisiología.
- Las cuestiones clave: carácter social y distribuido de la inteligencia, emociones, voluntad, autoconciencia, y significados.
- Las objeciones a las analogías mente-programa, cerebro-hardware, pensar-computar.
- Diferencias constitutivas entre los seres vivos y las máquinas; la “fisicalidad” de la inteligencia.
- Limitaciones de carácter lógico-matemático, disponibilidad de herramientas formales y modelos de computación, y de arquitecturas de ordenadores.

Historia de la inteligencia artificial (Sección 1.3)

1. Precedentes históricos en las ciencias cognitivas.

2. Hitos históricos desde que se acuñó el concepto.

3. Deficiencias del proceso evolutivo.

4. Panorama actual y futuro.

Metodología de la inteligencia artificial (Sección 1.3)

1. Pasos metodológicos en el desarrollo de una aplicación de inteligencia artificial.

Paradigmas de la inteligencia artificial (Sección 1.4)

1. Concepto de paradigma en inteligencia artificial.

2. Paradigma simbólico o representacional:

- Caracterización del paradigma.
- Inspiración biológica.
- Metodología de desarrollo de aplicaciones.

- Problemática básica.

- Aplicaciones.

1. Paradigma situado o reactivo:

- Caracterización del paradigma.

- Inspiración biológica.

- Metodología de desarrollo de aplicaciones.

- Problemática básica.

- Aplicaciones.

1. Paradigma conexionista (“métodos numéricos”):

- Caracterización del paradigma.

- Inspiración biológica.

- Metodología de desarrollo de aplicaciones.

- Problemática básica.

- Aplicaciones.

1. Paradigma híbrido

- Caracterización del paradigma

- Inspiración biológica

- Metodología de desarrollo de aplicaciones

- Problemática básica

- Aplicaciones

A continuación proporcionamos unas breves aclaraciones para facilitar la comprensión del capítulo, ya que consideramos que el alumno puede encontrar esta lectura particularmente compleja dado que trata cuestiones filosóficas y epistemológicas de gran calado. Estas aclaraciones también facilitarán la lectura de los textos complementarios que se facilitarán a través de la plataforma virtual de aprendizaje una vez iniciado el curso.

El autor del texto, el doctor José Mira Mira, fue catedrático de Ciencias de la Computación e Inteligencia Artificial y director del Departamento de Inteligencia Artificial de la ETS de Ingeniería Informática de la UNED, desde 1989 hasta su fallecimiento en el año 2008. Sus intereses de investigación estaban relacionados con los aspectos metodológicos de la Ingeniería del Conocimiento y con las relaciones entre la Neurociencia y la Computación. Desde una perspectiva aplicada, su interés se centraba en el desarrollo de Sistemas Basados en el Conocimiento en los dominios de la Medicina y la Industria.

Destacan, en esta síntesis de su visión del campo, la perspectiva multidisciplinar, amplia e integradora, el cuestionamiento profundo, el énfasis en lo biológico, su concepción de la inteligencia como fenómeno emergente de mecanismos genéticos, moleculares, neuronales, metabólicos, evolutivos, personales, históricos, sociales y culturales.

Nos detendremos a continuación en algunas cuestiones concretas que consideramos pueden plantear al alumno mayores dificultades:

El concepto de computabilidad. A lo largo del capítulo se insiste en que los problemas adecuados para ser abordados desde la inteligencia artificial son aquellos para los cuales disponemos tan sólo, en principio, de “descripciones poco claras, incompletas, imprecisas y con alto grado de dudas y errores potenciales debido a su complejidad”. Los objetivos de la inteligencia artificial consistirían, básicamente, en dar formalidad a estas descripciones, traduciéndolas a descripciones claras, completas, precisas e inequívocas. En referencia a una cita del filósofo y matemático Leibniz (a quien se debe el cálculo diferencial, cuya notación se utiliza aún hoy en día), el autor identifica el concepto anterior con el concepto de computabilidad: “todo lo que sepamos describir de forma clara, completa, precisa e inequívoca es computable”. De ahí que utilice expresiones que pueden resultar difíciles de interpretar tales como “hacer computable la semántica”, con lo que se refiere a la posibilidad de traducir la riqueza de significados del mundo real a un lenguaje interpretable por un ordenador. No debe confundirse pues esta acepción de la computabilidad con la noción rigurosa establecida en las ciencias de la computación, según la cual un algoritmo es computable cuando puede ser calculado por una máquina de Turing. En cuanto a esta acepción, un algoritmo especificado de forma clara, completa, etc. puede no ser computable. Incluso un algoritmo terminante puede considerarse no computable en la práctica, por cuestiones de complejidad espacial y temporal.

El aprendizaje como proceso de metanivel. En la metáfora mente-programa, los procesos mentales se equiparan a programas informáticos. Un metaprograma es un programa que en lugar de manipular datos manipula otros programas. En este sentido, el aprendizaje es un proceso mental equiparable a un metaprograma, ya que su efecto es modificar los propios procesos mentales. Es en este sentido que el autor lo describe como proceso de “metanivel”.

Niveles de descripción. El autor hace también énfasis repetidas veces en la importancia de no confundir niveles de descripción o formalización. En particular, insiste en la importancia de distinguir entre los propios procesos fisiológicos, la descripción que de ellos se hace en lenguaje natural, la traducción que de estas descripciones se hace en términos de lenguajes matemáticos formales y en términos de lenguajes informáticos interpretables por un ordenador, y finalmente la materialización de estos procesos en la máquina. Al alumno puede resultar difícil apreciar la relevancia de esta observación y sus implicaciones desde un punto de vista práctico.

En primer lugar, los procesos fisiológicos han sido programados por la evolución natural y después por la genética, la historia personal de adaptación al medio, etc. La descripción que de ellos hacemos parece presuponer una intencionalidad de la que los procesos en sí carecen: no son autoconscientes, sino que obedecen a sus leyes internas, siguen su propia “causalidad”, en palabras del autor. Por otro lado, a medida que formalizamos las descripciones evidentemente hay una pérdida de riqueza descriptiva. Finalmente, cuando un proceso se traduce a un ordenador, hay adicionalmente una pérdida de significados, de

conexión con referentes reales. De nuevo en el dominio de la máquina los procesos obedecen a sus propias leyes de causalidad, que son las de la electrónica y no las de la biología.

Estas reflexiones son interesantes y muy relevantes a nivel filosófico, y advierten ante el excesivo optimismo de la inteligencia artificial en su objetivo de reproducir la vida. Es lógico que el alumno encuentre dificultades para entenderlas en toda su profundidad, y no debe pretenderlo, ya que implican cuestiones de definición imprecisa, inaprensible, grandes preguntas eternamente debatidas en la historia del pensamiento humano.

Como complemento a este capítulo se proporcionarán textos adicionales de lectura obligatoria sobre aspectos generales de la IA a través del curso virtual. Se aconseja encarecidamente la realización de las actividades de aprendizaje propuestas para este tema, con el fin de ahondar en las cuestiones analizadas y de contrastar la visión y perspectiva del autor de este capítulo con las de otros autores, reflejadas en las mencionadas lecturas.

El resultado de aprendizaje que el alumno adquiere tras el estudio de este bloque de contenidos se resume en adquirir una perspectiva global de qué es la inteligencia artificial, y conocer la historia de la inteligencia artificial. (RA1 y RA2, respectivamente).

Bloque 2: Introducción a las Técnicas de Búsqueda

(Coincide con el capítulo 8 del texto base de la asignatura.)

La gran mayoría de sistemas inteligentes actuales aplican métodos de búsqueda para resolver algunas de las tareas que llevan a cabo. De ahí la necesidad de abordar en esta asignatura el estudio de diferentes métodos de búsqueda que son relevantes para la inteligencia artificial. A lo largo del capítulo 8 del texto base se supone que no se utiliza información del dominio para guiar la búsqueda de forma eficiente. La búsqueda que sí utiliza dicha información se denomina “búsqueda heurística” y será objeto de estudio en el capítulo 9 del texto base.

Las tres primeras secciones del capítulo 8 del texto base introducen e ilustran el concepto de “búsqueda” en inteligencia artificial. Después de haber leído estas tres secciones, se deben tener claros conceptos como: estado inicial, estado objetivo, operador, coste de un operador, espacio de búsqueda, estrategia de control, completitud o admisibilidad. El resto del capítulo diferencia entre espacios de búsqueda con forma de árbol y espacios de búsqueda con forma de grafo. En el caso de que un estado pueda ser visitado más de una vez a lo largo del proceso de búsqueda, es más eficiente suponer que el espacio de búsqueda es un grafo. Dos estructuras de datos resultan de especial interés independientemente de si el espacio de búsqueda es tratado como un árbol o como un grafo: por un lado, la lista ABIERTA contiene los nodos generados y no expandidos y, por otro lado, TABLA_A permite obtener el mejor camino parcial encontrado desde cada estado al estado inicial. Conviene tener claro

qué efectos produce la expansión de un estado tanto sobre ABIERTA como sobre TABLA_A.

En este capítulo se estudian los siguientes tipos de recorrido en árboles:

- (1) Primero en anchura
- (2) Primero en profundidad
- (3) Coste uniforme
- (4) Búsqueda en profundidad iterativa
- (5) Búsqueda en anchura iterativa
- (6) Búsqueda bidireccional (Aunque el texto base incluye este tipo de búsqueda en la sección correspondiente a recorrido en grafos, la mayoría de textos sobre inteligencia artificial la explican en el contexto de recorrido en árboles.)

Para cada método hay que tener claro su funcionamiento, las ventajas e inconvenientes derivadas de su complejidad temporal y de su complejidad espacial, así como sus características en cuanto a completitud y admisibilidad.

La *búsqueda primero en anchura* usa ABIERTA como una cola, posee complejidad espacial y complejidad temporal exponenciales con la profundidad de la solución y, además, es completa y admisible (esto último siempre que todos los operadores del mismo nivel de profundidad tengan igual coste).

La *búsqueda primero en profundidad* usa ABIERTA como una pila, tiene complejidad espacial lineal con la profundidad límite y no es ni siquiera un método completo.

La *búsqueda de coste uniforme* ordena los estados de ABIERTA teniendo en cuenta para cada estado el coste del camino que lo une con el estado inicial, posee complejidad análoga a la búsqueda primero en anchura y siempre es admisible.

La *búsqueda en profundidad iterativa* consiste en realizar sucesivas búsquedas primero en profundidad, incrementando la profundidad límite en una unidad para cada nueva búsqueda primero en profundidad. De esta manera se combinan las ventajas de una menor complejidad espacial (como en la búsqueda primero en profundidad) y de la admisibilidad (como en la búsqueda primero en anchura).

La *búsqueda en anchura iterativa* es un método de naturaleza iterativa al igual que la búsqueda en profundidad iterativa, aunque la cantidad que se incrementa en cada iteración es el número de sucesores de cada nodo que son generados y el tipo de búsqueda que se desarrolla en cada iteración es similar a la búsqueda primero en anchura. Este método no es admisible.

La *búsqueda bidireccional* simultanea la búsqueda usual del estado objetivo desde el estado inicial con otra búsqueda en dirección contraria, que parte del estado objetivo en busca del estado inicial. Esto reduce el exponente de la complejidad exponencial de otros métodos desde d a $d/2$, siendo d la profundidad de la solución, lo cual significa un ahorro considerable. La condición de terminación de este método se da cuando algún nodo coincide en las dos listas ABIERTA de las búsquedas directa e inversa. Para que este método se

pueda llevar a cabo los estados objetivo deben ser conocidos a priori, los operadores deben ser reversibles y al menos una de las dos búsquedas se debe realizar en anchura.

Por último, la búsqueda en grafos puede provocar que se descubra más de un camino desde el estado inicial hasta cualquier otro nodo. Esto implica que en TABLA_A deba anotarse para cada nodo n cuál de sus padres conocidos marca el mejor camino encontrado hasta el momento desde n al nodo inicial. Estos enlaces desde cada nodo a uno de sus padres forman un árbol en el espacio de búsqueda, que cambia a medida que avanza el proceso de búsqueda. Pueden surgir tres casos a la hora de expandir el siguiente nodo de ABIERTA, que siempre deberá ser sacado de ABIERTA después de su expansión y al que llamaremos “nodo padre expandido”:

(1) Que el “nodo hijo generado” correspondiente no haya sido generado o incluso expandido previamente. En este caso, el nodo hijo generado es introducido en ABIERTA y el enlace a su nodo padre recién expandido es incluido en TABLA_A.

(2) Que el nodo hijo generado ya haya sido generado pero no expandido previamente, es decir, el nodo hijo generado ya estaba en ABIERTA. En este caso hay que evaluar si hemos encontrado un camino mejor desde el nodo inicial al nodo hijo recién generado; en caso afirmativo, hay que enlazar en TABLA_A el nodo hijo recién generado con su nodo padre recién expandido. Este último proceso no es más que una reorientación de enlace.

(3) Que el nodo hijo generado ya haya sido expandido previamente, es decir, el nodo hijo no está en ABIERTA pero sus propios hijos ya han sido generados previamente. En este caso, además de los pasos llevados a cabo en el apartado anterior, hay que estudiar en un proceso recursivo descendente si se deben realizar posibles reorientaciones en los nodos descendientes del nodo hijo recién generado.

La búsqueda exhaustiva sin información del dominio explicada en este capítulo tiene el inconveniente de la ineficiencia. A medida que el tamaño del problema crece, la complejidad de estos métodos resulta ser exponencial con el tamaño del problema, lo cual dificulta encontrar un estado objetivo en un tiempo y con un espacio de memoria prudenciales. Por ello, resulta conveniente en la mayoría de casos reales emplear conocimiento del dominio que guíe la búsqueda por los estados más prometedores. Este otro tipo de búsqueda se denomina “búsqueda heurística” y es objeto de estudio en el capítulo 9 del texto base.

Los ejercicios propuestos y resueltos que el alumno puede encontrar tanto en el capítulo correspondiente del texto base como en el curso virtual de la asignatura resultan un complemento ideal para afianzar los conocimientos teóricos estudiados en este capítulo del texto base.

El resultado de aprendizaje que el alumno adquiere tras el estudio de este bloque de contenidos consiste en aplicar búsqueda en un espacio de estados a la resolución de problemas (RA3).

Bloque 3: Técnicas Basadas en Búsquedas Heurísticas

(Coincide con el capítulo 9 del texto base de la asignatura.)

La búsqueda heurística utiliza conocimiento del dominio para guiar la exploración de un espacio de estados. Los heurísticos son criterios que permiten establecer qué camino es supuestamente el mejor para alcanzar de forma más eficiente un estado objetivo; más concretamente, permiten determinar qué nodo candidato a ser expandido es el más prometedor. Es importante tener en cuenta que el conocimiento heurístico no suele producir valoraciones siempre correctas, sino que intenta proporcionar información lo más cercana posible a la realidad. Dicha cercanía determina la calidad del heurístico. A lo largo del capítulo 9 del texto base se explican algunos heurísticos simples para problemas como el 8-puzzle, las N -reinas, el viajante de comercio o el secuenciamiento de tareas, algunos de ellos obtenidos a partir de la relajación del problema original.

El primer método de búsqueda heurística descrito en este capítulo es la *búsqueda primero el mejor*. La definición de este método dada por los autores es muy genérica: es una especialización del algoritmo general de búsqueda en grafos, explicado en el capítulo anterior del texto base, en la que se usa una función heurística numérica para ordenar los elementos de ABIERTA.

El *algoritmo A^** , que es una especialización de la búsqueda primero el mejor, es el método sobre el que gira gran parte de los contenidos de este capítulo. Para su entendimiento conviene tener claras las definiciones de $f(n)$, $f^*(n)$, $g(n)$, $g^*(n)$, $h(n)$ y $h^*(n)$, donde n es un nodo cualquiera del espacio de búsqueda. Obsérvese que $f(n)$ consta de un sumando estático, $h(n)$, y de otro sumando $g(n)$ que puede cambiar a medida que avanza el proceso de búsqueda. La sección de este capítulo que describe las propiedades formales del algoritmo A^* en cuanto a terminación, completitud, admisibilidad y eficiencia contiene varias definiciones, lemas y teoremas que son importantes para entender el funcionamiento del algoritmo; no obstante, las demostraciones de estos lemas y teoremas, aunque es recomendable entenderlas, no es necesario memorizarlas.

Dos métodos que sacrifican la admisibilidad de A^* a cambio de una ganancia en eficiencia son la *ponderación dinámica* y el *algoritmo A^** . El primer método se basa en asociar unos pesos dinámicos a las funciones g y h , que permitan adaptar el tipo de exploración realizada en función de lo avanzado que se encuentre el proceso de búsqueda. Por otra parte, el segundo método utiliza una función heurística adicional para seleccionar uno de entre los nodos de una sublista de ABIERTA que contiene los nodos cuyo valor de f no difiere en más de un factor $1 + \epsilon$ del valor mínimo de f en ABIERTA.

Para finalizar con el estudio del método A^* , este capítulo describe algunas de sus variantes que emplean memoria limitada: el *algoritmo IDA** y el *algoritmo SMA**. El algoritmo IDA* es

una extensión del método de búsqueda en profundidad iterativa y, por tanto, la memoria necesaria en este algoritmo es proporcional a la profundidad de la solución y al factor de ramificación. La novedad consiste en que en cada iteración de la búsqueda en profundidad iterativa se establece un valor límite de f por encima del cual no se realiza exploración. En la primera iteración se fija $f(\text{nodo inicial})$ como límite de la exploración y en iteraciones sucesivas el nuevo límite de f se fija como el valor mínimo de f de los nodos descartados en la iteración previa. Por otra parte, el algoritmo SMA* acota la longitud de TABLA_A, de manera que cuando hay que expandir un nodo y no queda espacio libre en TABLA_A se descarta el nodo menos prometedor de ABIERTA. En cada nodo se anota el mejor f de los hijos descartados de ese nodo.

Finalizado el estudio del algoritmo A* y de algunas de sus variantes más importantes, este capítulo aborda diferentes métodos de búsqueda heurística ampliamente utilizados:

- (1) Algoritmos voraces
- (2) Algoritmos de ramificación y poda
- (3) Algoritmos de búsqueda local

Los *algoritmos voraces* toman decisiones irrevocables en la exploración, ya que no consideran alternativas al camino actual. A pesar de que no son algoritmos admisibles, son muy eficientes. La elección irrevocable de uno de los sucesores del nodo actual se hace en función de un heurístico, de manera que el sucesor más prometedor suele ser el elegido, aunque otras opciones de naturaleza probabilista son posibles. La explicación sobre algoritmos voraces dada en el capítulo 9 del texto base es bastante genérica y parece englobar tanto a los algoritmos de búsqueda local que eligen en cada paso el vecino más prometedor como a los algoritmos que manejan en cada paso una solución parcial del problema a resolver y van ampliando la solución parcial a partir del paso más prometedor. Un ejemplo del segundo caso se da en el problema del viajante cuando se parte de una ciudad y se elige la siguiente ciudad a visitar como la más cercana a la ciudad actual.

Los *algoritmos de ramificación y poda* interpretan cada estado como un subconjunto de soluciones del problema original. El proceso de ramificación consiste en descomponer un determinado conjunto de soluciones en la unión disjunta de varios subconjuntos suyos, con lo que el espacio de búsqueda adquiere forma de árbol. Cada nodo del árbol tiene asociado un valor heurístico que representa una cota inferior del coste de la mejor solución (considerada como la de menor coste) contenida en el nodo. Generalmente, cada vez que la cota inferior de un nodo rebasa el menor de los costes de los nodos hoja encontrados hasta el momento en el resto del árbol, dicho nodo es podado. (Un nodo hoja es aquel que no puede ser expandido por contener una única solución del problema original.) Por motivos de eficiencia, la estrategia de control para la exploración del árbol de búsqueda suele ser en profundidad, aunque otros métodos (heurísticos o no) son posibles.

Los *algoritmos de búsqueda local* se plantean en aquellos casos en que el camino solución desde el nodo inicial hasta un nodo meta es irrelevante y el nodo meta por sí solo contiene toda la información de interés. En estos casos se puede partir de una solución candidata inicial al problema y realizar pequeños cambios sucesivos en la misma para intentar llegar a la solución óptima. De esta manera, cada nodo tiene asociado un conjunto de nodos vecinos y la búsqueda local se encarga de explorar en cada paso uno de los nodos vecinos del nodo actual. En este capítulo se estudian tres tipos de búsqueda local:

- (1) *Búsqueda del gradiente*: Acepta una solución vecina (generalmente la mejor de todas las soluciones vecinas) sólo si es mejor o igual que la actual. Tiene el inconveniente de estancarse en óptimos locales, para lo cual la única alternativa es reiniciar la búsqueda.
- (2) *Temple simulado*: Realiza una elección aleatoria entre los vecinos del nodo actual y acepta el vecino elegido si es mejor que el nodo actual; en caso contrario y de cara a evitar óptimos locales, existe todavía cierta probabilidad de que el vecino sea aceptado. Esta probabilidad de aceptación es función de la diferencia en calidad de las dos soluciones y de un parámetro denominado “temperatura”. Dado que al principio la temperatura adquiere un valor alto y posteriormente es reducida en cada paso, inicialmente hay mayor probabilidad de que un vecino peor sea aceptado y al final del proceso de búsqueda dicha probabilidad es prácticamente nula. Es decir, al principio se da más énfasis a la exploración del espacio de búsqueda y, una vez que la convergencia está cercana, se favorece la búsqueda de las soluciones de calidad. Normalmente, la mayor dificultad de este método es la definición de un plan de enfriamiento adecuado, el cual depende fuertemente del tipo de problema que se está resolviendo (si contiene muchos óptimos locales o no).
- (3) *Búsqueda tabú*: A través de un mecanismo de memoria sobre el historial reciente de la búsqueda, este método descarta ciertos vecinos del nodo actual cuya exploración no resulta conveniente en principio. Los movimientos descartados se consideran “tabú”, aunque pueden ser excepcionalmente elegidos si conducen a una solución mejor que la mejor solución encontrada hasta el momento.

Los ejercicios propuestos y resueltos que el alumno puede encontrar tanto en el capítulo correspondiente del texto base como en el curso virtual de la asignatura resultan un complemento ideal para afianzar los conocimientos teóricos estudiados en este capítulo del texto base.

El resultado de aprendizaje que el alumno adquiere tras el estudio de este bloque de contenidos consiste en aplicar búsqueda en un espacio de estados a la resolución de problemas (RA3).

Bloque 4: Lógica y Representación del Conocimiento (con una introducción a los Conjuntos Borrosos)

(Coincide con el capítulo 2 y con las secciones 7.1, 7.2, 7.3 y 7.4 del texto base de la asignatura.)

Para la comprensión del capítulo 2 del texto base se asume que el alumno habrá cursado previamente las asignaturas del primer curso “Lógica y estructuras discretas” y “Autómatas, gramáticas y lenguajes”, tanto del grado en Ingeniería Informática como del grado en Ingeniería en Tecnologías de la Información. No obstante, incluye un repaso de los conceptos y contenidos básicos necesarios de estas materias, de modo que no resulta necesario que el alumno revise materiales didácticos complementarios para su seguimiento adecuado.

La lógica fue originariamente una rama de la filosofía que estudiaba los principios del razonamiento correcto desde un punto de vista formal, es decir, analizando la estructura de los argumentos abstraéndose de sus contenidos específicos y del lenguaje concreto en que se expresan.

Al adquirir estos estudios rigor y formalización simbólica se convierten en una rama de las ciencias matemáticas. En la lógica matemática surge el concepto de lógica como cálculo definido sobre unos símbolos mediante un conjunto de reglas que establecen las inferencias válidas (razonamientos correctos). Una lógica es pues en matemáticas un lenguaje, donde las reglas de inferencia son equivalentes a reglas sintácticas que determinan la validez de las proposiciones del lenguaje, y una semántica que asigna significados a los símbolos del lenguaje (establece correspondencias entre los símbolos y las entidades que estos representan en un mundo físico o abstracto). La lógica matemática encuentra su aplicación inmediata en ciencias de la computación y, naturalmente, en inteligencia artificial, ya que define un cálculo automatizable para la simulación de los procesos de razonamiento.

El propósito de este capítulo es mostrar el panorama general de aplicaciones de la lógica en inteligencia artificial y presentar las nociones básicas del razonamiento automático llamando la atención sobre los límites planteados por la complejidad computacional de los procedimientos algorítmicos implicados. A pesar de que proporciona una presentación bastante amplia tanto de la lógica matemática clásica como de las extensiones de ésta propuestas en el contexto de la inteligencia artificial, no es su intención proporcionar un manual completo, ni es exhaustivo en las demostraciones de los teoremas. Tampoco entra en el detalle del análisis de los problemas de computabilidad y decidibilidad.

No obstante, el carácter del capítulo es formal y riguroso, y está orientado a la práctica. Se recomienda a los alumnos que sigan al detalle los cálculos y los razonamientos de las demostraciones cualitativas que se proporcionan, y se ejerciten en la manipulación del lenguaje lógico rehaciéndolos manualmente. Esta práctica es imprescindible para la

comprensión en profundidad de este capítulo y del capítulo “Sistemas Basados en Reglas”, así como para su potencial aplicación futura en asignaturas del cuarto curso, en concreto, Ampliación de Sistemas Inteligentes (del Grado en Ingeniería Informática) e Ingeniería y Gestión del Conocimiento (del Grado en Ingeniería en Tecnologías de la Información).

De los capítulos del texto base de la asignatura, éste se encuentra entre los más didácticos y acertadamente ilustrado con ejemplos sencillos escogidos en campos de aplicación de la inteligencia artificial, particularmente la robótica. No obstante, a continuación redactamos una breve guía para su lectura.

En la sección 2.1 el autor comienza por definir conceptos clave tanto en el contexto del capítulo como para capítulos posteriores, particularmente aquellos que abordan el problema de la representación del conocimiento: “Sistemas basados en reglas” y “Redes semánticas y marcos”.

Introduce el problema de representación del conocimiento y las características fundamentales de una representación adecuada del conocimiento: el carácter esencial (no contiene información inútil) y formal (en un mismo contexto, símbolos iguales tienen igual significado o semántica). Introduce también el propio concepto de razonamiento (obtención de conclusiones correctas a partir de ciertas premisas consideradas correctas) y el concepto de método deductivo correcto y completo, y llama la atención sobre las características deseables de un método deductivo con miras a su automatización: la expresividad del lenguaje utilizado para expresarlo, sus propiedades computacionales y el grado de síntesis de las expresiones. Concluye finalmente las potencialidades de la lógica en este contexto, ya que proporciona un lenguaje para la representación del conocimiento, un cálculo para el razonamiento correcto, y una semántica para asignar significados en un cierto dominio de aplicación.

Esta introducción sienta las bases del resto del capítulo, donde primeramente se revisan la lógica proposicional clásica y de primer orden presentando su sintaxis, semántica y principales métodos deductivos, y destacando los límites de expresividad y problemas de complejidad asociados (secciones 2.2 y 2.3); para luego presentar las extensiones posibles de la lógica más utilizadas en el contexto de la inteligencia artificial (secciones 2.4 y 2.5).

Entendemos que las secciones 2.2 y 2.3 no plantearán al alumno ninguna dificultad, ya que suponen el repaso de conceptos ya aprendidos en la materia “Lógica y estructuras discretas”.

En la sección 2.2 se repasan los conceptos de lógica proposicional (proposición, modelo, tabla de verdad, equivalencia de fórmulas, tautología y contradicción). Se distingue entre métodos deductivos sintácticos y semánticos. Esta distinción tiene gran importancia desde un punto de vista práctico porque los últimos son los habitualmente empleados en las aplicaciones por ser más fácilmente implementables y utilizarse de forma extendida en el estudio de las nuevas lógicas. Entre estos métodos destacan el método del árbol semántico

y el método de resolución. El método del árbol semántico es un método no determinista polinomial de complejidad NP-completo. Por su parte, el método de resolución es el más utilizado en la práctica. Su complejidad se reduce cuando se limita la expresividad trabajando con un tipo particular de cláusulas, las cláusulas de Horn. Uno de los lenguajes emblemáticos de la inteligencia artificial, el lenguaje Prolog, se basa en este tipo de cláusulas. Dada su trascendencia, basamos en él la segunda de las actividades obligatorias de esta asignatura.

La sección 2.3 se centra en la lógica de primer orden o lógica de predicados. Primeramente se presentan su sintaxis y su semántica, introduciéndose sus conceptos básicos (variable, cuantificador, predicado) y haciendo énfasis en la mayor expresividad obtenida en relación a la lógica proposicional, ilustrándose, p.e., la representación del tiempo. A continuación se reflexiona sobre las complicaciones en términos de complejidad computacional y diseño de métodos deductivos que esta mayor expresividad acarrea. Finalmente se muestra cómo extender los métodos deductivos de la lógica proposicional a esta lógica. En el caso del método del árbol semántico, esta extensión comporta problemas de no terminación, cuyo origen está en el carácter no finito de los dominios (se trata de problemas semidecidibles o recursivamente enumerables). También el método de resolución comporta problemas nuevos respecto al caso proposicional, cuyo análisis conduce a la definición de los conceptos de “dominio de Herbrand” y “forma normal de Skolem”.

En la sección 2.3 se introducen también brevemente las lógicas de orden superior, mostrándose cómo la complejidad crece a la par que la potencia expresiva, dando lugar a lógicas no sólo semidecidibles sino no recursivamente enumerables. La sección se cierra definiendo el concepto de Fragmento de lógica proposicional, que supone restricciones mediante guardias, limitación de aridad de los predicados, o limitación de variables con el fin de combatir la indecidibilidad. Estas restricciones juegan un papel esencial en las extensiones a la lógica clásica estudiadas en las siguientes secciones del capítulo, en concreto, en las lógicas modales, temporales y monádica, respectivamente.

La sección 2.4 se dedica a las extensiones de la lógica clásica que, manteniendo un importante poder expresivo, no planteen problemas de decidibilidad. La sección comienza concretando las situaciones que sería deseable poder modelar (situaciones hipotéticas, situaciones espaciales y situaciones temporales) y los aspectos en que la lógica clásica difiere del razonamiento humano e incluso resulta menos eficaz en sus procedimientos de cálculo (habilidad para la resolución de problemas de carácter práctico frente a abstracto, uso de cantidades indefinidas sobre un dominio continuo, tolerancia a errores, ruido e incertidumbre, incorporación de razonamiento contextual e intencional, capacidad de anular conclusiones previas). Las extensiones a la lógica clásica presentadas a continuación incorporan algunas de estas propiedades del razonamiento humano.

A continuación, la sección se centra en la lógica modal, lógica que intenta acercarse más al razonamiento humano completando la lógica clásica al introducir el concepto modalidad. Esto significa que es posible indicar el modo en que es cierta o falsa una proposición (cuándo, dónde, bajo qué condiciones) y con ello expresar los conceptos de necesidad y posibilidad. Lo anterior se modela considerando un conjunto de mundos posibles relacionados por una relación de accesibilidad y cuantificadores existenciales que se aplican a las proposiciones permitiendo desplazarse entre mundos. Cada proposición tiene asignado un valor de verdad en cada mundo. Una fórmula es modalmente satisfacible si lo es en al menos un mundo. Existe una multiplicidad de sistemas modales, que se diferencian en el tipo de relación de accesibilidad y variantes de los operadores modales. Entre ellos se explican la lógica K y su sistema deductivo asociado, que es decidible y se basa en el método del árbol semántico.

Finalmente, en la sección 2.5 se presenta una interesante variante de la lógica K: las lógicas temporales. La interpretación de una lógica modal que permite utilizarla para modelar el tiempo es la siguiente: los mundos posibles se sustituyen por instantes de tiempo y la accesibilidad entre mundos por la sucesión temporal. Los operadores modales se convierten en operadores temporales. Se distingue entre lógicas temporales basadas en puntos (el tiempo se considera compuesto por un conjunto de puntos ordenados) y lógicas temporales basadas en intervalos (pares de puntos). El primer caso permite expresar eventos puntuales, situaciones pasadas y futuras con respecto al instante presente. Destacan en esta variedad la lógica LTL (Linear Time Logic), para la cual se utiliza el método deductivo decidible basado en árboles semánticos; y la lógica CTL, de gran expresividad y alta complejidad computacional, ampliamente aplicada en el campo de control de modelo o Model Checking (metodología de verificación automática de proyectos software). Las lógicas temporales basadas en intervalos se utilizan cuando no es preciso modelar eventos puntuales sino propiedades con una cierta duración. Destaca entre ellas la Lógica Proposicional de las Relaciones de Allen o HS. En este caso la deducción automática es mucho más compleja y plantea problemas de indecidibilidad. No obstante la capacidad expresiva de estas lógicas es muy alta, y resultan de gran interés en algunas áreas, particularmente en aplicaciones de tiempo real.

Para terminar, hemos considerado útil introducir también al alumno en otra interesante extensión de la lógica clásica: la lógica borrosa, constituida hoy en día en una importante área de desarrollo de la inteligencia artificial con múltiples aplicaciones industriales. En las secciones 7.1 a 7.4 del capítulo 7 se presentan los motivos que conducen al desarrollo de esta lógica: la necesidad de gestionar la imprecisión e incertidumbre del mundo real y la observación del modo en que los seres vivos afrontan estos problemas. Se definen también los conceptos básicos de teoría de conjuntos borrosos en referencia a la teoría convencional de conjuntos.

Los ejercicios propuestos y resueltos que el alumno puede encontrar tanto en este capítulo como en el curso virtual de la asignatura resultan un complemento ideal para afianzar los conocimientos teóricos estudiados y entender su aplicación.

El principal resultado de aprendizaje que el alumno adquiere tras el estudio de este bloque de contenidos consiste en utilizar las ampliaciones de la lógica clásica que resultan necesarias en inteligencia artificial (RA4).

Bloque 5: Sistemas Basados en Reglas

(Coincide con el capítulo 3 del texto base de la asignatura.)

Los Sistemas Basados en Reglas constituyen una de las principales áreas de la inteligencia artificial simbólica. Su surgimiento y la enorme difusión de sus aplicaciones en los años setenta, bajo el nombre de Sistemas Expertos y posteriormente Sistemas Basados en Conocimiento, suponen uno de los hitos más importantes de la inteligencia artificial. Entre las aplicaciones emblemáticas destacan en la década de su aparición Dendral y MYCIN, desarrollados en la universidad de Stanford en los campos de la biología molecular y el diagnóstico médico, respectivamente.

En la actualidad, la producción de Sistemas Basados en Reglas se ha constituido en un área relevante en el seno de la ingeniería informática, la “Ingeniería del Conocimiento”, que integra diferentes tecnologías en boga en el contexto actual, tales como la tecnología de agentes y orientación a objetos, y cuenta con metodologías de desarrollo y gestión de proyectos propias. De ahí la importancia de este capítulo en el contexto profesional.

Estos sistemas constituyen una aplicación inmediata de la lógica y sus mecanismos de inferencia, de modo que este capítulo está estrechamente conectado con el capítulo precedente, “Lógica y representación del conocimiento”, cuyo estudio previo es esencial para abordarlo. Su potencia expresiva es menor que la de la lógica de predicados, en aras de la eficiencia computacional, pero mayor que la de la lógica proposicional. En concreto, los Sistemas Basados en Reglas son una aplicación de los sistemas de deducción en lógica proposicional de primer orden, restringidos a cláusulas de Horn (de ahí que se haya seleccionado como segunda actividad obligatoria de esta materia el desarrollo de un pequeño sistema en el lenguaje Prolog). El capítulo también presenta algunas conexiones con los bloques de estudio 2 y 3, relacionados con la búsqueda heurística, por su aplicación en los algoritmos de selección de reglas en una Base de Conocimiento.

El estudio de los Sistemas Basados en Reglas no plantea dificultades particulares desde un punto de vista conceptual una vez se han comprendido los mecanismos de la lógica. Adicionalmente, el texto base de la asignatura presenta el tema con sencillez e introduciendo los conceptos en base a ejemplos, particularmente en el dominio médico. Es de destacar también que presta especial atención a dos objetivos de aprendizaje clave de esta materia:

la reflexión sobre los límites de aplicabilidad e idoneidad de las técnicas de la inteligencia artificial con respecto a otras áreas de la ingeniería del software, y la conexión con otras técnicas más convencionales de esta ingeniería.

A continuación presentamos una visión general del planteamiento del capítulo 3 del texto base. La sección 3.1 comienza introduciendo los Sistemas Basados en Reglas como sistemas que capturan “la experiencia humana en la resolución de problemas con el fin de alcanzar decisiones coherentes y repetibles”. El conocimiento se representa en forma declarativa, por traducción inmediata de heurísticas y formas de proceder de los expertos del campo de aplicación. De aquí se deriva otra característica clave de estos sistemas: la separación entre el conocimiento (factual o heurístico) y los mecanismos de deducción (motores de inferencia) que utilizan esos conocimientos para resolver problemas creando líneas de razonamiento.

En la sección 3.2 se introducen los elementos básicos de un Sistema Basado en Reglas (Base de Hechos, Base de Conocimientos y Motor de Inferencias) y los conceptos más relevantes asociados (equiparación, conjunto conflicto), en los que se profundiza en las secciones siguientes. En la sección 3.3 se presentan las técnicas básicas de inferencia: encadenamiento hacia delante (búsqueda de un conjunto de objetivos que se verifican dado un conjunto de entradas), hacia atrás (comprobación de la verificación de un objetivo dado con ciertas entradas), y mixto. Las técnicas se explican con detallados ejemplos ilustrados con representaciones gráficas inspiradas en la notación electrónica, lo que consideramos un acierto desde el punto de vista pedagógico dada la familiaridad del alumno con la teoría de los sistemas digitales, estudiada en asignaturas previas. La sección concluye indicando estrategias generales para determinar el tipo de encadenamiento más adecuado en cada caso.

En la sección 3.4 se presentan las técnicas de equiparación (selección de reglas compatibles con la Base de Hechos por sus condiciones y acciones), profundizando en el caso del algoritmo de equiparación RETE. El algoritmo Rete ha sido ampliamente utilizado en la construcción de Sistemas Expertos por la indudable eficiencia que aporta. No obstante, la velocidad de procesamiento se consigue a costa de consumir ingentes cantidades de memoria.

En la sección 3.5 se presentan las técnicas de resolución de conflictos (selección de una regla de entre las resultantes del proceso de equiparación) más usadas indicando las ventajas y desventajas de cada una de ellas y su adecuación en función de las características de la Base de Conocimientos.

Finalmente, las secciones 3.6 y 3.7 tienen un gran interés por presentar una visión crítica del área y situarla en el contexto de la ingeniería del software y de las aplicaciones de la inteligencia artificial. Destacamos a continuación las ideas clave presentadas en estas secciones:

- (1) Los Sistemas Basados en Reglas, al consistir en un conjunto de reglas independientes, son muy modulares y por tanto presentan buenas propiedades de mantenibilidad si su tamaño no es muy grande.
- (2) En sistemas grandes se requieren métodos de estructuración de la Base de Conocimientos para facilitar la depuración y evitar efectos colaterales en las fases de actualización y mantenimiento.
- (3) La representación declarativa del conocimiento facilita la incorporación en estos sistemas de facilidades de autoexplicación.
- (4) Actualmente las herramientas de desarrollo de Sistemas Basados en Reglas integran otras técnicas de programación convencionales.
- (5) Los Sistemas Basados en Reglas tienen menor potencia expresiva que la lógica de predicados, en aras de la eficiencia computacional, pero mayor que la lógica proposicional; e incorporan aspectos de diferentes extensiones de la lógica clásica.
- (6) Los campos de aplicación idóneos de los Sistemas Basados en Reglas son aquellos que se pueden modelar como un conjunto de múltiples estados, y el conocimiento se puede separar claramente de la forma en que se usa.

Los ejercicios propuestos y resueltos que el alumno puede encontrar tanto en este capítulo como en el curso virtual de la asignatura resultan un complemento ideal para afianzar los conocimientos teóricos estudiados y entender su aplicación.

El principal resultado de aprendizaje que el alumno adquiere tras el estudio de este bloque de contenidos consiste en entender el funcionamiento de un Sistema Basado en Reglas (RA5).

Bloque 6: Redes Semánticas y Marcos

(Coincide con el capítulo 4 del texto base de la asignatura.)

El capítulo 4 del texto base explica dos de los métodos de representación de conocimiento e inferencia más ampliamente utilizados: las *redes semánticas* y los *marcos*. Mientras que las primeras se centran en la representación de “relaciones” entre entidades del dominio, los segundos se centran en la representación de “conceptos” referidos a dichas entidades. Para cada uno de estos dos métodos se estudia en este capítulo, por un lado, cómo representan el conocimiento y, por otro lado, cómo realizan inferencia a partir de dicho conocimiento.

Las redes semánticas, creadas en los años 60, utilizan grafos dirigidos para representar conocimiento, donde tanto los nodos como los arcos están etiquetados. Mientras que los nodos representan conceptos o instancias que son ejemplo de dichos conceptos, los arcos representan relaciones binarias entre nodos. Existen dos tipos de arcos en una red semántica: los descriptivos y los estructurales. Los arcos descriptivos representan atributos y unen el objeto al que hace referencia el atributo con el valor asociado al atributo; de esta

manera, un objeto puede ser descrito mediante un conjunto de arcos descriptivos. Los arcos estructurales unen conceptos y determinan la arquitectura de la red. Su semántica es independiente del dominio tratado y suelen representar procesos de generalización, instanciación o agregación. En este capítulo se hace referencia a una serie de aspectos importantes asociados a la representación de conocimiento mediante redes semánticas:

- (1) El conocimiento expresado en una red semántica puede ser representado utilizando lógica de predicados de primer orden.
- (2) Los predicados lógicos de aridad superior a dos se pueden representar en una red semántica.
- (3) Oraciones simples o compuestas pueden ser representadas mediante redes semánticas tomando el verbo como elemento principal de la representación.
- (4) El carácter disjunto de diferentes entidades del dominio se puede representar en redes semánticas.

La inferencia en redes semánticas se realiza principalmente por medio de dos métodos: *equiparación* y *herencia de propiedades*. La equiparación consiste en comparar un fragmento de red con una red semántica para comprobar si el fragmento está incluido en la red. En caso de que la comparación resulte positiva, se podrá asignar a los nodos variable del fragmento de red los valores que toman sus nodos asociados en la red semántica. Respecto al método de inferencia mediante herencia de propiedades, se lleva a cabo a partir de los enlaces “Subclase-de” e “Instancia” de la red semántica. Consiste en que a los nodos más específicos de la red se les pueda asignar los valores de las propiedades asociadas a nodos más genéricos. Las excepciones a los valores genéricos deberán ser incluidas explícitamente en la red (como valores de propiedades de los nodos específicos) si se quiere que sean tenidas en cuenta.

Los marcos, creados en la década de los 80, utilizan para representar conocimiento grafos dirigidos en cada uno de cuyos nodos se estructura la información correspondiente a las características de cierta entidad del dominio. Estos grafos son construidos por especialización de conceptos generales en conceptos más específicos. Asociado a este conocimiento declarativo de carácter jerárquico y estereotipado, existe generalmente conocimiento procedimental adicional, de manera que un sistema de marcos utiliza cuatro elementos fundamentales para representar conocimiento: *marcos*, *relaciones*, *propiedades* y *facetas*. Hay dos tipos de marcos, clases e instancias, y mientras que las primeras expresan conceptos genéricos, las segundas constituyen ejemplos concretos de dichos conceptos genéricos. Los diferentes marcos quedan enlazados entre sí por medio de relaciones. Las relaciones estándar más importantes son las de “Subclase-de”, que une una clase específica con otra clase más genérica que la engloba, y la de “Instancia”, que une una instancia con la clase a la que pertenece. Existen otras relaciones no estándar que expresan dependencias entre conceptos del dominio: la relación “Fraternal” une dos clases cuya clase padre

coincide, la relación “Disjunta” une dos clases que no pueden tener una instancia común y, finalmente, diferentes relaciones “A medida” o “Ad-hoc” entre dos clases pueden ser definidas por el usuario. Las propiedades de un marco permiten describir el concepto asociado al mismo, de manera que cada propiedad represente un atributo o característica genérica del marco. Las propiedades se crean en las clases de un sistema de marcos, pero pueden ser definidas bien para que los valores que toman sólo puedan ser rellenados en clases bien para que sólo puedan ser rellenados en instancias de clases. Cada propiedad y cada relación de un sistema de marcos tiene un conjunto de facetas asociadas. Una faceta puede definir cierta característica como: el tipo de dato con el que se rellenará la propiedad o relación, la cardinalidad mínima o máxima del dato y si es multivaluado o no. Existen otras facetas que afectan a propiedades de instancias y que indican el conjunto de valores válidos de la propiedad, el valor por omisión de la propiedad, o qué procedimiento o regla hay que ejecutar si se necesita, modifica, añade o borra el valor de la propiedad.

La inferencia en marcos se lleva a cabo mediante tres tipos principales de procesos: *equiparación*, *herencia* y *demonios*. La equiparación es un método de inferencia utilizado en sistemas de marcos para la clasificación de entidades en la base de conocimiento. Dados los valores conocidos de un subconjunto de propiedades de un marco pregunta, la equiparación identifica qué marco de la base de conocimiento corresponde al marco pregunta. En muchas ocasiones, este proceso no es un proceso exacto y debe llevarse a cabo en tres fases: selección de marcos candidatos, cálculo de valores de equiparación y toma de la decisión final. Otro método de inferencia en sistemas de marcos es la herencia de propiedades, que permite compartir propiedades y valores de propiedades usando las relaciones “Subclase-de” e “Instancia”. La herencia es simple cuando el sistema de marcos tiene forma de árbol, considerando exclusivamente las relaciones “Subclase-de” e “Instancia”. En este caso, el valor de una propiedad de una instancia se busca en primer lugar en dicha instancia y, en caso de fracaso, se toma de la clase más cercana a la instancia donde figure dicho valor. Si un marco es subclase o instancia de más de una clase, se dice que la herencia es múltiple. En este caso, dado que una instancia puede tener más de una clase padre que contenga la propiedad buscada, el valor de la propiedad que hereda la instancia depende del método de búsqueda utilizado para recorrer el grafo: en profundidad, en anchura o utilizando el concepto de “distancia inferencial”. Por último, otra técnica de inferencia en marcos son los demonios, que son procedimientos asociados a propiedades de marcos y cuya función es recuperar, almacenar o borrar información en las mismas.

En la sección de resumen de este capítulo aparece un interesante estudio comparativo entre redes semánticas y marcos. Algunos puntos destacados de esta comparación son los siguientes:

- (1) Los marcos permiten agrupar y estructurar la información asociada a cada entidad del dominio. En cambio, en las redes semánticas la información correspondiente a una misma

entidad está dispersa en la red.

(2) Las redes semánticas son declarativas aunque no procedimentales, mientras que los marcos contienen una parte declarativa y otra procedimental.

(3) Las redes semánticas no contienen valores por omisión, mientras que los marcos sí.

(4) Una base de conocimiento formada por un sistema de marcos es más fácilmente ampliable que otra formada por una red semántica.

Los ejercicios propuestos y resueltos que el alumno puede encontrar tanto en el capítulo correspondiente del texto base como en el curso virtual de la asignatura resultan un complemento ideal para afianzar los conocimientos teóricos estudiados en este capítulo del texto base.

Los resultados de aprendizaje que el alumno adquiere tras el estudio de este bloque de contenidos son: por un lado, caracterizar los distintos tipos de redes semánticas utilizadas para representar conocimiento en inteligencia artificial (RA6) y, por otro lado, usar marcos como método de representación de conocimiento estructurado y como método de inferencia (RA7).

METODOLOGÍA

Al constar esta asignatura de 6 créditos y corresponder cada crédito a 25 horas de trabajo del alumno, la carga total de trabajo se corresponde con 150 horas.

Las actividades formativas realizadas por el alumno se distribuyen entre el trabajo autónomo (estimado en 120 horas) y el tiempo de interacción con el equipo docente o con el tutor a través de los distintos medios existentes (estimado en 30 horas). El trabajo autónomo del alumno se divide entre el trabajo con contenidos teóricos (estimado en 60 horas) y el desarrollo de tareas prácticas (estimado en 60 horas) como, por ejemplo, la realización del plan de actividades o la resolución de problemas.

El alumno estudiará los contenidos teóricos presentes en el texto base de la asignatura. Como complemento a lo anterior, podrá realizar los ejercicios prácticos (propuestos o resueltos) incluidos tanto en el texto base de la asignatura como en el curso virtual de la misma. Además, deberá realizar dos actividades evaluables preparadas por el equipo docente y corregidas por un profesor tutor.

El enfoque pedagógico estará guiado por los objetivos y resultados de aprendizaje señalados en secciones anteriores. Estos implican en primer lugar la comprensión en profundidad de distintos paradigmas, técnicas y aplicaciones de la Inteligencia Artificial en el contexto general de la ingeniería informática, y la práctica de su aplicación en problemas sencillos. Suponen también el desarrollo de una actitud crítica y un juicio inteligente de sus ventajas e inconvenientes, la formación de criterios de evaluación de su idoneidad y límite de aplicabilidad en diferentes ámbitos. Todo lo anterior requiere un aprendizaje activo y significativo.

El aprendizaje activo se estimulará mediante la experimentación con ejercicios prácticos que

ilustrarán el fundamento y objeto de los diferentes conceptos teóricos. Por otro lado, los recursos didácticos utilizados estarán orientados al aprendizaje significativo, con énfasis en situar los contenidos en el contexto general de aprendizaje del alumno. De este modo, los conceptos teóricos se situarán en el contexto global de las ciencias de la computación, y los contenidos de carácter práctico en el contexto general de los paradigmas, técnicas y aplicaciones de la ingeniería informática. Se fomentará asimismo la argumentación en torno a las ventajas de las soluciones tecnológicas propuestas.

Las habilidades técnicas específicas de la materia se desarrollarán mediante los ejercicios prácticos de carácter optativo y mediante las actividades evaluables. Con estas últimas se ejercitarán particularmente las competencias profesionales genéricas mencionadas previamente. Estas actividades tendrán el formato de pequeños proyectos informáticos e implicarán la redacción de una breve memoria y la práctica de competencias cognitivas superiores, competencias de expresión y comunicación, y usos de recursos de la Sociedad del Conocimiento. Los objetivos de formación integral se atenderán también particularmente al abordar los primeros capítulos de temario, “Perspectiva Histórica y Conceptual” y “Aspectos Metodológicos en Inteligencia Artificial”. Los ejercicios prácticos relativos a estos temas instarán al alumno a la reflexión sobre las dimensiones filosóficas, y humanistas en general, de la disciplina mediante técnicas pedagógicas constructivistas, basadas en la consulta de documentación diversa y el trabajo en grupo.

Las dudas de estudio, y las relativas a los ejercicios y actividades prácticas, podrán consultarse durante las tutorías presenciales que tienen lugar en los centros asociados, o haciendo uso de los foros del curso virtual. Cada alumno tendrá asociado un profesor tutor que atenderá a sus dudas y orientará su estudio. Adicionalmente, los profesores de la asignatura proporcionarán asimismo pautas de estudio e intervendrán para la aclaración de los aspectos más complejos de la materia.

PLAN DE TRABAJO

En el cómputo de horas se incluyen el tiempo dedicado a las horas lectivas, horas de estudio, tutorías, seminarios, trabajos, prácticas o proyectos, así como las exigidas para la preparación y realización de exámenes y evaluaciones.

BLOQUE: 1: Aspectos Conceptuales de la Inteligencia Artificial y la Ingeniería del Conocimiento - 15 Horas

[TEMPORALIZACIÓN: SEMANAS 1-2 DE 14]

[RESULTADOS DE APRENDIZAJE: RA1 y RA2]

Se estima que la lectura reflexionada en profundidad del capítulo I requiere aproximadamente 2 horas de dedicación, y la de los textos complementarios requiere un total de 5 horas. Adicionalmente, se sugiere al alumno que dedique 8 horas adicionales a la realización de las actividades descritas a continuación, con el fin de satisfacer los objetivos de aprendizaje del capítulo. Se recomienda realizar las actividades siguiendo la siguiente secuencia:

1. Lectura de las secciones 1.1, 1.2, 1.5 y 1.6.
2. Lectura de textos complementarios y realización de la actividad 1.a: La inteligencia artificial bajo debate.
3. Lectura de la sección 1.3.
4. Lectura de textos complementarios y realización de la actividad 1.b: Historia de la inteligencia artificial.
5. Lectura de la sección 1.4.
6. Realización de la actividad 1.c: Evaluación de idoneidad de técnicas y paradigmas.

Este plan de actividades tiene como fin los siguientes resultados de aprendizaje:

- Comprensión del concepto de inteligencia artificial y visión crítica del campo desde múltiples perspectivas multidisciplinares; elaboración de una síntesis personal en función del contexto personal de aprendizaje, los conocimientos previos y la experiencia profesional.
- Adquisición de una visión general de la historia de la inteligencia artificial y de los distintos paradigmas, técnicas y aplicaciones en el contexto de la ingeniería informática.
- Práctica en el juicio y crítica de opciones tecnológicas en el ámbito de la inteligencia artificial (criterios personales formados para la evaluación de idoneidad y límite de aplicabilidad de los diferentes paradigmas, técnicas y métodos).
- Reconocimiento de los retos y logros de la inteligencia artificial en su justa medida.
- Desarrollo de las competencias generales G2, G4 y G5, particularmente de análisis y síntesis, razonamiento crítico, comunicación y expresión escrita en registro científico y tecnológico.

Actividad 1.a: La inteligencia artificial bajo debate

Dado que no existe consenso en una visión única de la problemática de la inteligencia artificial, no tendría cabida presentar unos contenidos cerrados para su estudio memorístico. Es por esta razón que estos primeros temas se abordan con una metodología de trabajo particular. En su comprensión es particularmente pertinente un enfoque pedagógico activo y participativo, constructivista, con objeto de que el alumno construya su propia perspectiva del campo en base a su contexto de aprendizaje, sus conocimientos previos y experiencia profesional.

Se ha optado pues por ofrecer al alumno lecturas adicionales que complementen al capítulo I del texto base. En ellas se manifiestan diferentes visiones del campo y se hace particular énfasis en diferentes cuestiones. Las lecturas requieren una dedicación aproximada de 3 horas y se proporcionarán a través del entorno virtual de aprendizaje al inicio del curso. Podrán tratar cuestiones adicionales a las presentadas en el capítulo 1 tales como:

- Tesis de Turing, teorema de incompletitud de Gödel, caja china de Searle.

- Dimensiones filosóficas de la inteligencia artificial.
- Idiosincrasia de la inteligencia artificial en el contexto de la ética informática.
- Implicaciones socio-políticas.

Nuestra sugerencia es que los alumnos, tras haber leído detenidamente estos textos, respondan a un cuestionario acerca de lo leído. A continuación intercambien sus respuestas para la evaluación por pares y, finalmente, entablen debates al respecto bien en los espacios de debate virtual bien de forma presencial en las horas de tutoría de los centros.

La actividad se realiza pues en dos fases:

1. *Cumplimentado de un cuestionario* (dedicación estimada: 2 horas). Las preguntas mueven a reflexionar sobre las distintas visiones ofrecidas, las cuestiones subyacentes comunes y matices diferenciadores, las relaciones entre conceptos, las conexiones con las experiencias y conocimientos previos, los dilemas éticos que plantea la ingeniería inteligente...
2. *Evaluación por pares del cuestionario y debate final* (dedicación estimada: 3 horas). En este caso no se proporcionan las soluciones del cuestionario con el fin de motivar el debate, bien sea virtual o presencial.

Actividad 1.b: Historia de la inteligencia artificial

En el capítulo I del texto base se proporciona una visión panorámica, a grandes rasgos de la historia de la inteligencia artificial. Hemos juzgado interesante proporcionar una lectura adicional (dedicación estimada: 2 horas) que aporta detalles de interés y a continuación un cuestionario de autoevaluación (dedicación estimada: 1 hora) que ayude a los alumnos a fijar los datos más significativos.

Se recomienda al alumno que realice el cuestionario después de la lectura y sin buscar las respuestas en los solucionarios, de modo que pueda estimar hasta qué punto ha asimilado los datos esenciales, apreciar los matices en la asimilación incorrecta de conceptos y repasar aquellos aspectos que considere necesario.

Actividad 1.c: Evaluación de idoneidad de técnicas y paradigmas

Se trata de una actividad de resolución de casos prácticos. El enunciado consiste en la descripción de una serie de problemas que se pretende solucionar mediante una aplicación informática que utilice técnicas de inteligencia artificial. Para cada uno de ellos se debe:

- Reflexionar sobre el interés o no de aplicar soluciones de inteligencia artificial.
- Proponer un paradigma y técnicas específicas, justificando su adecuación al problema; o bien argumentar las ventajas e inconvenientes de diferentes paradigmas y técnicas alternativos.
- Proporcionar indicaciones metodológicas sobre cómo se abordaría el desarrollo de la aplicación.

- Señalar las fortalezas, debilidades y límites de aplicabilidad de las soluciones propuestas.
- Reflexionar sobre las cuestiones éticas y socio-políticas implicadas en el desarrollo de la aplicación.

Se estima que la realización de esta última actividad requiere una dedicación de 2 horas.

BLOQUE: 2: Introducción a las Técnicas de Búsqueda - 15 Horas

[TEMPORALIZACIÓN: SEMANAS 2-3 DE 14]

[RESULTADOS DE APRENDIZAJE: RA3]

La estrategia de aprendizaje propuesta para este bloque consiste en estudiar primeramente los contenidos teóricos del texto base de la asignatura (capítulo 8) y, como complemento, realizar los ejercicios prácticos (propuestos o resueltos), tanto los incluidos en el texto base como los proporcionados en el curso virtual.

BLOQUE: 3: Técnicas Basadas en Búsquedas Heurísticas - 15 Horas

[TEMPORALIZACIÓN: SEMANAS 4-5 DE 14]

[RESULTADOS DE APRENDIZAJE: RA3]

La estrategia de aprendizaje propuesta para este bloque consiste en estudiar primeramente los contenidos teóricos del texto base de la asignatura (capítulo 9) y, como complemento, realizar los ejercicios prácticos (propuestos o resueltos), tanto los incluidos en el texto base como los proporcionados en el curso virtual.

PEC: 1: Búsqueda en un Espacio de Estados - 15 Horas

[TEMPORALIZACIÓN: SEMANAS 5-6 DE 14. FECHA TOPE DE ENTREGA APROXIMADA: PRIMEROS DE ABRIL]

[RESULTADOS DE APRENDIZAJE: RA3]

La actividad evaluable sobre búsqueda en un espacio de estados consistirá en la realización por parte del alumno de seis ejercicios prácticos. El alumno deberá seguir los contenidos del texto base de la asignatura a la hora de dar respuesta a estos ejercicios, cuya temática y criterios de evaluación se especifican a continuación.

Ejercicio 1:

Tiene que ver con los espacios de búsqueda y vendría a responder al siguiente esquema genérico:

Dibuje mediante un grafo dirigido o describa detalladamente mediante una tabla el espacio de estados (o espacio de búsqueda) completo para el *<problema-real>*. Para ello especifique: el conjunto de todos los estados posibles, el estado inicial, el o los estados meta, los operadores aplicables a cada estado y el coste asociado a cada operador. En el <

problema-real>, <explicación del *problema-real*>. ¿Cuál es la solución menos costosa para el <*problema-real*>?

La evaluación sobre 10 puntos de este ejercicio se realizaría atendiendo a los siguientes criterios:

- Los estados se especifican correctamente: 2.5 puntos
- Los operadores se especifican correctamente: 2.5 puntos
- Los costes de los operadores se especifican correctamente: 1 punto
- El espacio de búsqueda se dibuja (mediante un grafo dirigido) o se describe (mediante una tabla) correctamente: 3 puntos
- La solución de menor coste se especifica correctamente: 1 punto

Ejercicio 2:

Tiene que ver con el orden de expansión de nodos bajo diferentes métodos de búsqueda sin información del dominio y vendría a responder al siguiente esquema genérico:

Considere el espacio de búsqueda de la <figura>, que tiene forma de árbol, donde el nodo raíz del árbol es el nodo inicial, existe un único nodo meta y cada operador tiene asociado un coste. Explique razonadamente en qué orden se expandirían los nodos de dicho árbol de búsqueda a partir de cada uno de los métodos siguientes de búsqueda sin información del dominio:

1. Búsqueda Primero en Anchura (<de izquierda a derecha>)
2. Búsqueda Primero en Profundidad (<de derecha a izquierda>)
3. Búsqueda de Coste Uniforme
4. Búsqueda en Anchura Iterativa (<de derecha a izquierda>)
5. Búsqueda en Profundidad Iterativa (<de izquierda a derecha>)

<figura que contiene el espacio de búsqueda en forma de árbol>

La evaluación sobre 10 puntos de este ejercicio se realizaría atendiendo a los siguientes criterios:

- Cada uno de los cinco apartados, correspondiente a un método de búsqueda concreto, se puntúa sobre 2 puntos.
- Si en cualquiera de los cinco apartados el algoritmo correspondiente no expande los nodos en el orden debido: la puntuación del apartado bajaría 1.6 puntos si el orden dado como respuesta varía significativamente del correcto, mientras que si el orden dado como respuesta varía del correcto como consecuencia de un despiste no conceptual entonces la puntuación del apartado bajaría sólo 0.4 puntos en vez de los 1.6 puntos mencionados.
- Si en cualquiera de los cinco apartados el algoritmo correspondiente no finaliza cuando es debido, la puntuación del apartado bajaría 0.4 puntos. (Esto es independiente del orden de

expansión de nodos dado como respuesta, que ya ha sido valorado anteriormente con un máximo de 1.6 puntos.)

Ejercicio 3:

Tiene que ver con la descripción del contenido de la lista ABIERTA a lo largo del proceso de búsqueda bajo diferentes métodos de búsqueda sin información del dominio y vendría a responder al siguiente esquema genérico:

Considere el espacio de búsqueda de la <figura>, que tiene forma de árbol, donde el nodo raíz del árbol es el nodo inicial, existe un único nodo meta y cada operador tiene asociado un coste. Describa cuál es el contenido de ABIERTA, previamente a cada extracción de un nodo de la misma, a partir de cada uno de los métodos siguientes de búsqueda sin información del dominio:

1. Búsqueda Primero en Anchura (<de izquierda a derecha>)
2. Búsqueda Primero en Profundidad (<de derecha a izquierda>)
3. Búsqueda de Coste Uniforme
4. Búsqueda en Anchura Iterativa (<de derecha a izquierda>)
5. Búsqueda en Profundidad Iterativa (<de izquierda a derecha>)

<figura que contiene el espacio de búsqueda en forma de árbol>

La evaluación sobre 10 puntos de este ejercicio se realizaría atendiendo a los siguientes criterios:

- Cada uno de los cinco apartados, correspondiente a un método de búsqueda concreto, se puntúa sobre 2 puntos.
- Si en cualquiera de los cinco apartados el algoritmo correspondiente no gestiona ABIERTA en la forma debida: la puntuación del apartado bajaría 1.6 puntos si la respuesta dada varía significativamente de la correcta, mientras que si la respuesta dada varía de la correcta como consecuencia de un despiste no conceptual entonces la puntuación del apartado bajaría sólo 0.4 puntos en vez de los 1.6 puntos mencionados.
- Si en cualquiera de los cinco apartados el algoritmo correspondiente no finaliza cuando es debido, la puntuación del apartado bajaría 0.4 puntos. (Esto es independiente de la gestión de ABIERTA dada como respuesta, que ya ha sido valorada anteriormente con un máximo de 1.6 puntos.)

Ejercicio 4:

Tiene que ver con la descripción del contenido de TABLA_A a lo largo del proceso de búsqueda bajo diferentes métodos de búsqueda sin información del dominio y vendría a responder al siguiente esquema genérico:

Considere el espacio de búsqueda de la <figura>, que tiene forma de árbol, donde el nodo raíz del árbol es el nodo inicial, existe un único nodo meta y cada operador tiene asociado un coste. Describa cuál es el contenido de TABLA_A, posteriormente a cada expansión de un nodo, a partir de cada uno de los métodos siguientes de búsqueda sin información del dominio:

1. Búsqueda Primero en Anchura (<de izquierda a derecha>)
2. Búsqueda Primero en Profundidad (<de derecha a izquierda>)
3. Búsqueda de Coste Uniforme
4. Búsqueda en Anchura Iterativa (<de derecha a izquierda>)
5. Búsqueda en Profundidad Iterativa (<de izquierda a derecha>)

<figura que contiene el espacio de búsqueda en forma de árbol>

La evaluación sobre 10 puntos de este ejercicio se realizaría atendiendo a los siguientes criterios:

- Cada uno de los cinco apartados, correspondiente a un método de búsqueda concreto, se puntúa sobre 2 puntos.
- Si en cualquiera de los cinco apartados el algoritmo correspondiente no gestiona TABLA_A en la forma debida: la puntuación del apartado bajaría 1.6 puntos si la respuesta dada varía significativamente de la correcta, mientras que si la respuesta dada varía de la correcta como consecuencia de un despiste no conceptual entonces la puntuación del apartado bajaría sólo 0.4 puntos en vez de los 1.6 puntos mencionados.
- Si en cualquiera de los cinco apartados el algoritmo correspondiente no finaliza cuando es debido, la puntuación del apartado bajaría 0.4 puntos. (Esto es independiente de la gestión de TABLA_A dada como respuesta, que ya ha sido valorada anteriormente con un máximo de 1.6 puntos).

Ejercicio 5:

Tiene que ver con la descripción de los pasos seguidos por el algoritmo A* al aplicarlo sobre un espacio de búsqueda concreto con información heurística del dominio y vendría a responder al siguiente esquema genérico:

Considere el grafo de la <figura>, donde el nodo inicial es <nodo-inicial> y donde los nodos meta son <nodos-meta>. Cada arco u operador lleva asociado su coste y en cada nodo aparece la estimación de la menor distancia desde ese nodo a una meta. Aplique paso a paso el algoritmo A* al grafo dado, indicando de forma razonada la siguiente información en cada paso del algoritmo:

1. Qué nodo es expandido.
2. Cuál es el contenido de ABIERTA tras la expansión del nodo, indicando el valor de la función de evaluación heurística para cada nodo de ABIERTA.

3.Cuál es el contenido de TABLA_A tras la expansión del nodo. Para cada nodo de TABLA_A incluya la siguiente información:

1. Su nodo padre que indique el camino de menor coste hasta el nodo inicial encontrado hasta el momento
2. El coste del camino de menor coste hasta el nodo inicial encontrado hasta el momento
3. Sus nodos hijos (si el nodo de TABLA_A actual ya ha sido expandido)

Por último, ¿cuál es el camino solución hallado y su coste?

<figura que contiene el espacio de búsqueda con información heurística>

La evaluación sobre 10 puntos de este ejercicio se realizaría atendiendo a los siguientes criterios:

- El orden seguido en la expansión de los nodos se puntúa sobre 1.5 puntos.
- La forma en que se gestiona ABIERTA se puntúa sobre 3.75 puntos. Se hará especial énfasis en comprobar qué nodos hay en ABIERTA en cada paso del algoritmo y qué valores de la función de evaluación heurística se les asignan.
- La forma en que se gestiona TABLA_A se puntúa sobre 3.75 puntos. Se hará especial énfasis en comprobar qué nodos hay en TABLA_A en cada paso del algoritmo y qué padre mejor se les asigna (teniendo en cuenta las posibles reorientaciones o rectificaciones de enlaces)
- La correcta terminación del algoritmo se puntúa sobre 1 punto. Se hará especial énfasis en comprobar cuándo termina el algoritmo y qué camino solución devuelve.

Ejercicio 6:

Tiene que ver con la descripción de los pasos seguidos por el algoritmo de escalada o máximo gradiente al aplicarlo sobre un espacio de búsqueda concreto con información heurística del dominio y vendría a responder al siguiente esquema genérico:

Considere el grafo de la figura, donde el nodo inicial es <nodo-inicial> y donde los nodos meta son desconocidos. Cada arco u operador lleva asociado su coste y en cada nodo aparece su valor de la función de evaluación heurística (que hay que minimizar). Aplique paso a paso el algoritmo de escalada o máximo gradiente al grafo dado. Para ello indique de forma razonada qué nodo se expande en cada paso y cuál es el nodo final devuelto por el algoritmo. Utilice como criterio de selección el de mejor vecino. Utilice como criterio de terminación <criterio-de-terminación>.

<figura que contiene el espacio de búsqueda con información heurística>

La evaluación sobre 10 puntos de este ejercicio se realizaría atendiendo a los siguientes criterios:

- La correcta aplicación en cada paso del algoritmo del *criterio de selección* del vecino o hijo del nodo actual (qué vecino o hijo del nodo actual es considerado como candidato para sustituirlo) se puntúa sobre 4.5 puntos.
- La correcta aplicación en cada paso del algoritmo del *criterio de aceptación* del vecino o hijo seleccionado (si el vecino o hijo candidato sustituye o no finalmente al nodo actual) se puntúa sobre 4.5 puntos.
- La correcta aplicación del *criterio de finalización* del algoritmo se puntúa sobre 1 punto.

IMPORTANTE:

La nota sobre 10 de esta actividad evaluable se calcula del siguiente modo:

$$\text{NOTA} = 0.125 \times \text{nota}_1 + 0.1 \times \text{nota}_2 + 0.2 \times \text{nota}_3 + 0.2 \times \text{nota}_4 + 0.3 \times \text{nota}_5 + 0.075 \times \text{nota}_6,$$
donde nota_i representa la nota sobre 10 del ejercicio i .

BLOQUE: 4: Lógica y Representación del Conocimiento (con una Introducción a los Conjuntos Borrosos) - 15 Horas

[TEMPORALIZACIÓN: SEMANAS 7-8 DE 14]

[RESULTADOS DE APRENDIZAJE: RA4]

La estrategia de aprendizaje propuesta para este bloque consiste en estudiar primeramente los contenidos teóricos del texto base de la asignatura (capítulo 2 y cuatro primeras secciones del capítulo 7) y, como complemento, realizar los ejercicios prácticos (propuestos o resueltos), tanto los incluidos en el texto base como los proporcionados en el curso virtual.

BLOQUE: 5: Sistemas Basados en Reglas - 15 Horas

[TEMPORALIZACIÓN: SEMANAS 8-9 DE 14]

[RESULTADOS DE APRENDIZAJE: RA5]

La estrategia de aprendizaje propuesta para este bloque consiste en estudiar primeramente los contenidos teóricos del texto base de la asignatura (capítulo 3) y, como complemento, realizar los ejercicios prácticos (propuestos o resueltos), tanto los incluidos en el texto base como los proporcionados en el curso virtual.

BLOQUE: 6: Redes Semánticas y Marcos - 15 Horas

[TEMPORALIZACIÓN: SEMANAS 9-10 DE 14]

[RESULTADOS DE APRENDIZAJE: RA6 y RA7]

La estrategia de aprendizaje propuesta para este bloque consiste en estudiar primeramente los contenidos teóricos del texto base de la asignatura (capítulo 4) y, como complemento, realizar los ejercicios prácticos (propuestos o resueltos), tanto los incluidos en el texto base como los proporcionados en el curso virtual.

PEC: 2: Construcción en Prolog de un Sistema Basado en Reglas Básico - 25 Horas

[TEMPORALIZACIÓN: SEMANAS 10-12 DE 14. FECHA TOPE DE ENTREGA

APROXIMADA: PRIMEROS DE MAYO]

[RESULTADOS DE APRENDIZAJE: RA3 a RA7]

El Prolog (PROgramation en LOGique) es un lenguaje de programación emblemático de la inteligencia artificial, muy utilizado aún en nuestros días. Se basa en la lógica de predicados de primer orden restringida a cláusulas de Horn para representar datos y conocimiento, y utiliza encadenamiento hacia atrás y una estrategia de reversibilidad sin información heurística conocida como backtracking. Por estas razones se ha considerado un lenguaje ideal para comprender desde una perspectiva práctica los conceptos presentados en los capítulos que tratan sobre representación del conocimiento e inferencia (Lógica y representación del conocimiento, Sistemas basados en reglas y Redes semánticas y marcos), así como para experimentar con el diseño de Sistemas Basados en Reglas sencillos e introducirse en el campo de las metodologías de desarrollo de la inteligencia artificial.

La actividad consistirá en la ejecución de un pequeño proyecto de desarrollo de un Sistema Basado en Reglas, desde la fase inicial de compilación del conocimiento experto necesario, hasta la fase final de pruebas. El enunciado de la actividad definirá las funcionalidades que debe cubrir la aplicación. Se dará al alumno la opción de realizar el trabajo individualmente o en grupo, proporcionándose diferentes enunciados para cada caso. Los profesores determinarán las dimensiones de la base de conocimiento a modelar en función de la aplicación y del número de componentes del grupo de trabajo.

Los alumnos habrán de documentarse sobre el conocimiento experto necesario y posteriormente codificarlo en Prolog de forma que la aplicación exhiba las funcionalidades deseadas. El conocimiento experto podrá proceder de recursos bibliográficos, expertos humanos, pericia de los propios alumnos, etc.

Para la realización de esta actividad se proporcionarán a través de los cursos virtuales:

1. Referencias de sitios Web donde podrán encontrar recursos sobre el lenguaje Prolog, en particular, tutoriales introductorios e intérpretes de libre distribución. Señalamos que, para la realización del proyecto, se requerirá sólo la utilización de mecanismos básicos del lenguaje: definición de hechos y reglas, formulación de preguntas u objetivos, y comprensión de las estrategias básicas de unificación y control, y del efecto del operador de corte.
2. Una colección de ejercicios que guiará el aprendizaje de los conceptos de Prolog relevantes para codificar sistemas basados en reglas sencillos.

Para la planificación del proyecto y redacción de la memoria final, se recomienda la lectura del capítulo 19 del texto base: Ingeniería del conocimiento. Obviamente las metodologías de

desarrollo que aquí se describen están concebidas para proyectos de mucha mayor envergadura. No obstante, consideramos de interés que el alumno proceda con cierto rigor metodológico en el abordaje del problema, y adicionalmente aprenda por propia experiencia lo que estas metodologías aportan.

La memoria del proyecto habrá de incluir:

1. La descripción del conocimiento del dominio modelado en forma estructurada, clara, precisa e inequívoca.
2. La descripción breve de la metodología de desarrollo simplificada que se ha aplicado y, en el caso de proyecto de grupo, de la estrategia de trabajo en grupo, identificando las partes realizadas por los diferentes miembros en aquellos casos en que puedan delimitarse. (En una página como máximo).
3. La descripción de la estructura de base de reglas y de las consideraciones de eficiencia realizadas.
4. El código comentado en Prolog, en forma de base de reglas y hechos.
5. La descripción de casos de prueba necesarios para mostrar que el sistema responde a las funcionalidades requeridas.
6. Un informe breve de las dificultades encontradas, y una valoración de las aportaciones y/o inconvenientes relacionados con la aplicación de la metodología de desarrollo.

Criterios de Evaluación

La evaluación sobre 10 puntos de este ejercicio se realizaría atendiendo a los siguientes criterios:

1. La descripción del conocimiento del dominio modelado en forma estructurada, clara, precisa e inequívoca se puntúa sobre 1 punto (se evaluarán la completitud: 0.4 puntos; la estructura: 0.2 puntos; la precisión: 0.2 puntos; y la claridad: 0.2 puntos)
2. La descripción breve de la metodología de desarrollo simplificada que se ha aplicado y, en el caso de proyecto de grupo, de la estrategia de trabajo en grupo se puntúa sobre 0.5 puntos (se evaluará la adecuación de la metodología y estrategia).
3. La descripción de la estructura de base de reglas y de las consideraciones de eficiencia realizadas se puntúa sobre 2 puntos (se evaluarán la adecuación de la estructura: 1 punto; y las consideraciones de eficiencia: 1 punto).
4. El código comentado en Prolog, en forma de bases de reglas y hechos se puntúa sobre 4 puntos (se evaluarán la corrección: 2 puntos; y la completitud: 2 puntos).
5. La descripción de casos de prueba necesarios para mostrar que el sistema responde a las funcionalidades requeridas se puntúa sobre 2 puntos (se evaluarán la completitud de los casos de prueba: 1 punto; y la pertinencia de los casos de prueba: 1 punto).
6. El informe breve de las dificultades encontradas, y una valoración de las aportaciones y/o inconvenientes relacionados con la aplicación de la metodología de desarrollo se puntúan

con 0.5 puntos (se evaluará la pertinencia de los argumentos).

En el caso de los trabajos realizados en grupo, en caso de haberse identificado las partes realizadas por los diferentes miembros la nota de cada alumno se calculará promediando una nota individual y una nota colectiva.

OTRAS ACTIVIDADES: - Preparación de la Prueba Presencial - 18 Horas

[TEMPORALIZACIÓN: SEMANAS 13-14 DE 14]

[RESULTADOS DE APRENDIZAJE: RA1 a RA7]

De cara a preparar la prueba presencial, recomendamos al alumno releer detenidamente el apartado de METODOLOGÍA de esta guía.

PRUEBA PRESENCIAL: 2 horas

Total Horas ECTS introducidas aquí : 150

SISTEMA DE EVALUACIÓN

TIPO DE PRUEBA PRESENCIAL

Tipo de examen	Examen de desarrollo
Preguntas desarrollo	3
Duración del examen	120 (minutos)
Material permitido en el examen	

Ninguno.

Criterios de evaluación

La prueba presencial constará de alrededor de tres preguntas de carácter teórico o práctico. En las preguntas teóricas se busca que el alumno demuestre que ha adquirido una visión general de cierta parte de la materia, así como que sepa relacionar y comparar diferentes conceptos o técnicas de interés en inteligencia artificial. Por otra parte, en las preguntas prácticas se pide que el alumno aplique correctamente los métodos explicados para la resolución de problemas concretos de búsqueda o de representación de conocimiento.

% del examen sobre la nota final	80
Nota del examen para aprobar sin PEC	6,2
Nota máxima que aporta el examen a la calificación final sin PEC	8
Nota mínima en el examen para sumar la PEC	4
Comentarios y observaciones	

La autoevaluación constituye un aspecto primordial en cualquier proceso de aprendizaje. A través del curso virtual de la asignatura se ofrece al alumno la posibilidad de realizar tests, contestar a preguntas teóricas y desarrollar la solución a problemas prácticos. Dado que las soluciones estarán disponibles para los alumnos, éstos podrán autoevaluarse de forma continua a lo largo del desarrollo de la asignatura, de cara a abordar con garantías de éxito la prueba presencial final.

PRUEBAS DE EVALUACIÓN CONTINUA (PEC)

¿Hay PEC?

Descripción

(1) PEC1: "Búsqueda en un Espacio de Estados"

(2) PEC2: "Construcción en Prolog de un Sistema Basado en Reglas Básico"

Criterios de evaluación

Debido a su extensión, los criterios de evaluación detallados de las PECs se especifican en los enunciados de las mismas.

Ponderación de la PEC en la nota final 20%

Fecha aproximada de entrega (1) PEC1: Primeros de Abril. (2) PEC2: Primeros de Mayo.

Comentarios y observaciones

En esta asignatura se realizan dos PECs (actividades evaluables) para las que no será necesario que el alumno acuda al Centro Asociado, ya que podrán realizarse a distancia. Las PECs se entregarán a través del curso virtual y serán corregidas por un profesor tutor. En principio, la nota final de las PECs será la media de las puntuaciones obtenidas en cada una de las dos PECs y constituirá un 20% de la nota final de la asignatura.

Es importante tener en cuenta que sólo se corregirán las PECs una vez durante el curso (previamente a la convocatoria de junio). Por tanto, la nota asignada a las PECs de cara a junio será la única válida tanto para la convocatoria de junio como para la de septiembre. En caso de que el alumno no realice la entrega de PECs de cara a la convocatoria de junio, se le asignará un cero al 20% de la nota final correspondiente a las PECS, tanto para la convocatoria de junio como para la de septiembre. La nota obtenida en las PECs es válida únicamente para las convocatorias del curso académico en que se entregan.

OTRAS ACTIVIDADES EVALUABLES

¿Hay otra/s actividad/es evaluable/s?

Descripción

No hay otras actividades evaluables.

Criterios de evaluación

Ponderación en la nota final 0

Fecha aproximada de entrega

Comentarios y observaciones

¿CÓMO SE OBTIENE LA NOTA FINAL?

La nota final sobre 10 se obtiene aplicando la siguiente formula:

NOTA FINAL = (0.8 x NOTA EXAMEN) + (0.2 x NOTA PECS),

donde NOTA EXAMEN es la nota de la prueba presencial sobre 10 y NOTA PECS es la nota de las PECS sobre 10.

BIBLIOGRAFÍA BÁSICA

ISBN(13):9788448156183

Título:INTELIGENCIA ARTIFICIAL: TÉCNICAS, MÉTODOS Y APLICACIONES (2008)

Autor/es:Marín Morales, Roque ; Palma Méndez, José Tomas ;

Editorial:Mc-Graw Hill

La asignatura consta de un libro de texto base. Los capítulos específicos que hay que estudiar de este libro se detallan en la versión extendida de esta guía.

Texto base:

Inteligencia Artificial: Técnicas, Métodos y Aplicaciones

José Tomás Palma Méndez y Roque Marín Morales

McGraw-Hill, 2008

ISBN: 9788448156183

La elección de este texto como texto base de la asignatura se ha visto motivada por el objetivo esencial de formar a profesionales críticos, capaces de evaluar con juicio inteligente las diferentes opciones tecnológicas y aplicaciones características de la Inteligencia Artificial. El texto se caracteriza por su visión crítica del campo, reconociendo sus retos y valorando sus logros en su justa medida. Se ha valorado también que los diferentes capítulos hayan sido redactados por expertos reputados en sus respectivos campos, profesores experimentados de diferentes universidades españolas, que transmiten tanto su clara visión como su erudito conocimiento, proporcionando referencias clave para la profundización en los tópicos implicados. Estos aspectos son coherentes con un enfoque pedagógico activo, constructivista y significativo. Se trata de un texto didáctico aunque exhaustivo, que incluye adicionalmente una amplia colección de ejercicios propuestos de gran interés. Por último, indicar que este texto está disponible como e-book.

BIBLIOGRAFÍA COMPLEMENTARIA

ISBN(13):9788420540030

Título:INTELIGENCIA ARTIFICIAL: UN ENFOQUE MODERNO (2ª)

Autor/es:Norvig, Peter ; Russell, Stuart ;

Editorial:PRENTICE-HALL

ISBN(13):9788478290178

Título:PROBLEMAS RESUELTOS DE INTELIGENCIA ARTIFICIAL APLICADA. BÚSQUEDA Y REPRESENTACIÓN. (1ª)

Autor/es:Fernández Galán, Severino ; González Boticario, Jesús ; Mira Mira, José ;

Editorial:PEARSON ADDISON-WESLEY

Como material complementario con el que profundizar de forma voluntaria en los contenidos del texto que conforma la bibliografía básica, el equipo docente recomienda los siguientes libros:

- *Problemas Resueltos de Inteligencia Artificial Aplicada: Búsqueda y Representación*

Severino Fernández Galán, Jesús González Boticario y José Mira Mira

Addison-Wesley, 1998

ISBN: 9788478290178

Este libro contiene una extensa colección de problemas resueltos sobre búsqueda en un espacio de estados y representación de conocimiento mediante lógica, reglas, redes semánticas y marcos. Por un lado, la consulta de estos ejercicios ilustra al alumno en la aplicación de los contenidos teóricos del texto base y, por otro lado, complementa los ejercicios propuestos y resueltos del propio texto base y del curso virtual.

- *Inteligencia Artificial: Un Enfoque Moderno*

Stuart Russell y Peter Norvig

Prentice Hall, 2004, 2ª Edición

ISBN: 9788420540030

Este libro es un clásico de la literatura del campo, utilizado desde su primera edición en una importante mayoría de universidades del mundo para la docencia de la Inteligencia Artificial. De naturaleza similar a la del texto base de la asignatura, ofrece una visión alternativa, centrada en el concepto de Agente Inteligente. Constituye un compendio muy completo que incorpora las áreas más innovadoras y constituye una referencia obligada, pero para el propósito de esta asignatura se ha considerado menos adecuado, desde el punto de vista pedagógico, que el mencionado texto base.

RECURSOS DE APOYO Y WEBGRAFÍA

La presente asignatura está virtualizada a través de un curso virtual en el que se ponen a disposición del alumno los siguientes materiales de apoyo al estudio:

- (1) Una versión extendida de la presente guía de estudio
- (2) Una lista de preguntas frecuentes sobre la asignatura
- (3) Tests y ejercicios resueltos que el alumno puede usar para su autoevaluación

Además, el curso virtual constituye una canal de comunicación de gran utilidad para el intercambio de ideas entre alumnos y profesores.

GLOSARIO

Esta sección contiene un glosario de términos relevantes para la asignatura, ordenados alfabéticamente según el bloque temático correspondiente.

BLOQUE 1: Aspectos Conceptuales de la Inteligencia Artificial y la Ingeniería del Conocimiento

Abductiva, inferencia. Razonamiento que deriva explicaciones de los hechos conocidos.

Aprendizaje. Proceso de modificación de las funciones cognitivas con el fin de adaptar un organismo a su entorno.

Cibernética. Ciencia que estudia la construcción de sistemas electrónicos y mecánicos a partir de su comparación con los sistemas de comunicación y regulación automática de los seres vivos.

Conexionista, paradigma de la inteligencia artificial. Aquel en que el conocimiento se representa mediante líneas numéricas etiquetadas para la entrada y salida de una red y la inferencia se realiza mediante un clasificador numérico de naturaleza paramétrica en el que el valor de esos parámetros se ajusta mediante un algoritmo de aprendizaje supervisado o no supervisado.

Conocimiento, Ingeniería del. Disciplina de la Inteligencia Artificial cuyo fin es el diseño y desarrollo de Sistemas Expertos (o Sistemas Basados en el Conocimiento).

Deductiva, inferencia. Razonamiento que deriva conclusiones a partir de unas premisas.

Deliberativo, agente. Aquel que contiene un modelo simbólico del mundo, explícitamente representado, en donde las decisiones se toman utilizando mecanismos de razonamiento lógico basados en la concordancia de patrones y la manipulación simbólica.

Formal, lenguaje. Aquel que se define mediante unas reglas fijas de formación de expresiones y significados (una sintaxis y una semántica formales).

Híbrido, paradigma de la inteligencia artificial. Aquel que incorpora elementos de distintos paradigmas.

IA débil. Postura adoptada sobre los avances de la inteligencia artificial que sostiene que las computadoras sólo pueden aspirar a mimetizar la inteligencia, comportarse como si fueran inteligentes, sin serlo realmente.

IA fuerte. Postura adoptada sobre los avances de la inteligencia artificial que sostiene que la

inteligencia que posee un ser vivo procede de un programa y que es reproducible en todas sus dimensiones mediante un computador.

Inductiva, inferencia. Razonamiento que deriva conclusiones generales a partir de premisas que contienen datos particulares.

Inferencia. Razonamiento, obtención de una conclusión.

Inteligencia. Medida global de la calidad de todos los procesos cognitivos de un ser vivo y de su capacidad de adaptación a los cambios del medio en el que existen otros seres vivos de complejidad comparable.

No situado, conocimiento. El asociado a aquellas situaciones en las que la interfaz entre el sistema de inteligencia artificial y el medio es humana, por lo que no hay que preocuparse de sensores y de efectores.

Ontología. Formulación de un exhaustivo y riguroso esquema conceptual de un dominio considerado.

Paradigma de la inteligencia artificial. Aproximación metodológica a la inteligencia artificial consensuada entre un amplio grupo de profesionales del campo que establece una forma de modelar conocimiento, formalizar los modelos, programar los operadores formales e implementar físicamente el soporte de esos programas, junto con la hipótesis de partida acerca de qué se entiende por conocimiento.

Proceso cognitivo. Proceso psicológico de orden superior que implica principalmente a las funciones de percepción, memoria, razonamiento y aprendizaje.

Reactivo, agente. Aquel que no incluye un modelo simbólico del mundo, actuando en base a un patrón estímulo- respuesta.

Relacional, paradigma de la inteligencia artificial. Ver simbólico, paradigma de la inteligencia artificial.

Semántica. Correspondencia entre expresiones de símbolos y entidades de un mundo físico o abstracto.

Simbólico, paradigma de la inteligencia artificial. Aquel en que se considera que todo el conocimiento necesario para resolver una tarea puede representarse usando descripciones declarativas y explícitas en lenguaje natural formada por un conjunto de conceptos, los hechos, y otro conjunto de reglas de inferencia que describen las relaciones estáticas y dinámicas conocidas entre esos hechos.

Símbolo. Representación de una entidad de un mundo físico abstracto.

Sintaxis. Conjunto de reglas de una gramática que define las combinaciones válidas de símbolos.

Situado, paradigma. Aquel en que el sistema de inteligencia artificial se considera en interacción con su entorno por medio de sensores y efectores.

BLOQUE 2: Introducción a las Técnicas de Búsqueda

ABIERTA. Lista que contiene los nodos generados pero no expandidos.

Algoritmo de búsqueda. Estrategia de control que decide el orden en que se exploran los estados de un espacio de búsqueda.

Búsqueda admisible. La que siempre encuentra una solución óptima.

Búsqueda bidireccional. La que simultanea la búsqueda del estado objetivo desde el estado inicial y viceversa.

Búsqueda completa. La que siempre encuentra solución en el caso de que exista.

Búsqueda de coste uniforme. La que expande el nodo cuyo camino al nodo inicial es el de menor coste.

Búsqueda en anchura iterativa. La que realiza iterativamente búsquedas primero en anchura aumentando en cada iteración en una unidad el número de sucesores de cada nodo que son generados.

Búsqueda en profundidad iterativa. La que realiza iterativamente búsquedas primero en profundidad aumentando en cada iteración en una unidad la profundidad límite.

Búsqueda heurística. La que utiliza información del dominio.

Búsqueda no informada. La que no utiliza información del dominio.

Búsqueda primero en anchura. La que usa ABIERTA como una cola.

Búsqueda primero en profundidad. La que usa ABIERTA como una pila.

Espacio de búsqueda. Conjunto de estados, operadores aplicables a cada estado y coste de cada operador, que están asociados a un problema de búsqueda.

Reorientación. Cambio del enlace que parte de un nodo hacia aquel padre suyo que está contenido en el camino parcial más corto desde el nodo al nodo inicial, debido a que se ha encontrado un nuevo padre que acorta el camino hacia el nodo inicial.

TABLA_A. Tabla que permite obtener el mejor camino parcial encontrado desde cada nodo al nodo inicial.

BLOQUE 3: Técnicas Basadas en Búsquedas Heurísticas

Algoritmo A*. Especialización de la búsqueda primero el mejor, basada en usar una función heurística que estima para cada nodo n el coste menor para ir del nodo inicial a un nodo meta pasando por el nodo n .

Algoritmo A*. Variante del algoritmo A*, que utiliza una función heurística adicional para elegir el siguiente nodo a expandir de entre un subconjunto de ABIERTA formado por sus elementos más prometedores que difieren del mejor elemento de ABIERTA como máximo en un factor $(1+)$.

Algoritmo de búsqueda local. Aquel que persigue generar el estado óptimo a partir de un estado inicial cualquiera al que se realizan pequeños cambios sucesivos.

Algoritmo de ponderación dinámica. Variante del algoritmo A*, que utiliza pesos dinámicos en la definición de la función heurística.

Algoritmo de ramificación y poda. Aquel que interpreta cada estado como un subconjunto de soluciones del problema original, ramifica dichas soluciones en un árbol, asocia un valor heurístico a cada nodo del árbol que representa una cota inferior del coste de la mejor solución contenida en el nodo y poda aquellos nodos que se sabe que no pueden mejorar la mejor solución encontrada hasta el momento.

Algoritmo IDA*. Variante del algoritmo A*, basado en realizar una búsqueda en profundidad iterativa acotada en cada iteración por ciertos valores conservadores de la función heurística.

Algoritmo SMA*. Variante del algoritmo A*, que utiliza memoria limitada y descarta el nodo menos prometedor de ABIERTA cuando se necesita liberar memoria.

Algoritmo voraz. Aquel que toma decisiones irrevocables en la exploración y no considera alternativas al camino actual.

Búsqueda del gradiente. Tipo de búsqueda local que acepta una solución vecina sólo si es mejor o igual que la actual.

Búsqueda primero el mejor. Especialización del algoritmo general de búsqueda en grafos,

en la que se usa una función heurística numérica cualquiera para ordenar ABIERTA.

Búsqueda tabú. Tipo de búsqueda local que descarta ciertos vecinos del nodo actual a partir del historial reciente de la búsqueda.

Heurístico. Criterio para establecer la calidad de un estado en un problema de búsqueda.

Temple simulado. Tipo de búsqueda local que realiza una elección aleatoria entre los vecinos del nodo actual y puede llegar a elegir un vecino peor con cierta probabilidad que decrece progresivamente a lo largo del proceso de búsqueda.

BLOQUE 4: Lógica y Representación del Conocimiento (con una introducción a los Conjuntos Borrosos)

Árbol semántico, método deductivo del. Método de deducción no determinista de la validez de una fórmula, basado en la reducción al absurdo.

Autoepistémica, lógica. Extensión de la lógica clásica basada en la hipótesis del mundo cerrado.

Basada en intervalos, lógica temporal. Aquella en que el tiempo se define mediante pares de puntos, permitiendo modelar propiedades de cierta duración.

Basada en puntos, lógica temporal. Aquella en que el tiempo se considera compuesto por un conjunto de puntos ordenados, permitiendo expresar eventos puntuales, y situaciones pasadas y futuras con respecto al instante presente.

Borrosa, lógica. Extensión de la lógica clásica donde las proposiciones tienen un grado de verdad que se asigna mediante una función de pertenencia que toma valores en el intervalo real $[0,1]$.

Circunscripción de predicados. Ver autoepistémica, lógica.

Cláusula. Fórmula consistente en una disyunción de literales.

Complejidad NP. Denominación de la clase de problemas que pueden ser resueltos en tiempo polinómico por una máquina de Turing no determinista.

Contradicción. Proposición que es falsa en todos los modelos posibles.

CTL, lógica. Lógica temporal basada en puntos de gran expresividad y alta complejidad computacional, ampliamente aplicada en el campo de control de modelo o Model Checking.

Difusa, lógica. Ver borrosa, lógica.

Equivalencia, entre fórmulas. Tiene lugar si y sólo si las fórmulas tienen el mismo conjunto de modelos que las satisfacen.

Fragmento de lógica proposicional. Subconjunto de la lógica proposicional, que supone restricciones mediante guardias, limitación de aridad de los predicados, o limitación de variables con el fin de combatir la indecibilidad.

Hipótesis del mundo cerrado. Hipótesis asumida en la lógica Autoepistémica o la Circunscripción de predicados, consistente en suponer como falso todo aquello que no esté explícitamente afirmado.

Horn, cláusula de. Cláusula consistente en un átomo, una implicación donde el antecedente es una conjunción de literales positivos y el consecuente un solo literal positivo, o bien una implicación donde el antecedente es una conjunción de literales negativos, y el consecuente es vacío.

Intuicionista, lógica. Extensión de la lógica clásica que redefine el concepto de tautología.

K, lógica. Variante de uso extendido de la lógica modal que utiliza un conjunto simplificado de axiomas.

Lógica. Cálculo definido sobre unos símbolos mediante un conjunto de reglas que establecen las inferencias válidas (razonamientos correctos). Una lógica es en matemáticas un lenguaje, donde las reglas de inferencia son equivalentes a reglas sintácticas que determinan la validez de las proposiciones del lenguaje, y una semántica que asigna significados a los símbolos del lenguaje.

Lógica proposicional. Es un sistema formal diseñado para analizar ciertos tipos de argumentos. En lógica proposicional, las fórmulas representan proposiciones y las conectivas lógicas son operaciones sobre dichas fórmulas, capaces de formar otras fórmulas de mayor complejidad.

LTL, lógica. Lógica temporal basada en puntos para la cual se utiliza el método deductivo decidible basado en árboles semánticos.

Modal, lógica. Extensión de la lógica clásica donde es posible indicar el modo en que es cierta o falsa una proposición (cuándo, dónde, bajo qué condiciones) y con ello expresar los conceptos de necesidad y posibilidad.

Modelo. Asignación de valores de verdad a todos los símbolos proposicionales que aparecen en una fórmula lógica.

Multivaluada, lógica. Extensión de la lógica clásica donde se permiten más de dos valores de verdad.

No monótona, lógica. Extensión de la lógica clásica caracterizada por la posibilidad de volver atrás en las conclusiones.

Orden superior, lógica de. Extensión de la lógica de primer orden que se caracteriza por tener variables relacionales de uno o varios órdenes, todas las cuales pueden cuantificarse.

Por defecto, lógica. Extensión de la lógica clásica donde es posible establecer conclusiones y excepciones por defecto.

Predicados, lógica de. Extensión de la lógica proposicional en la cual se introducen variables para denotar elementos del dominio, cuantificadores y predicados.

Primer orden, lógica de. Ver Predicados, lógica de.

Proposición. Expresión lógica que puede ser falsa o verdadera.

Relaciones de Allen o HS, lógica de las. Lógica temporal basada en intervalos de alta capacidad expresiva y aplicabilidad en problemas de tiempo real.

Resolución, método de. Método deductivo basado en averiguar la satisfacibilidad de un conjunto de fórmulas.

Semántico, método deductivo. Aquel que busca un contraejemplo para una fórmula, intentando demostrar que la fórmula no es satisfacible.

Sintáctico, método deductivo. Aquel que busca una demostración final de la validez de una fórmula a través de reglas de deducción.

Tableau. Ver árbol semántico, método deductivo del.

Tablero semántico. Ver árbol semántico, método deductivo del.

Tautología. Fórmula que independientemente del modelo en que se evalúe siempre es verdadera.

Temporal, lógica. Interpretación de una lógica modal que permite utilizarla para modelar el tiempo sustituyendo los mundos posibles por instantes de tiempo y la accesibilidad entre mundos por la sucesión temporal, y convirtiendo los operadores modales en operadores temporales.

BLOQUE 5: Sistemas Basados en Reglas

Base de Conocimientos. Componente de un Sistema Basado en Reglas que almacena segmentos de conocimiento relacional entre datos y conceptos.

Base de Hechos. Memoria de trabajo de un Sistema Basado en Reglas que acumula un conjunto de hechos establecidos que se usan para determinar qué reglas puede aplicar el mecanismo de inferencias.

Búsqueda bidireccional. Ver reversibilidad.

Ciclo de reconocimiento-acción. Proceso iterativo de aplicación del encadenamiento hacia delante.

Conjunto conflicto. Conjunto de reglas seleccionadas por un proceso de equiparación.

Declarativa, representación del conocimiento. Aquella en que está especificado el conocimiento pero no el modo en que debe ser usado.

Encadenamiento hacia atrás de reglas. Modo de inferencia de un Sistema Basado en Reglas que parte de un conjunto de hipótesis e intenta verificar estas hipótesis usando datos de la Base de Hechos o datos externos (obtenidos, por ejemplo, del usuario).

Encadenamiento hacia delante de reglas. Modo de inferencia de un Sistema Basado en Reglas que partiendo de una colección de hechos o afirmaciones de partida aplica las reglas de la Base de Conocimiento repetidas veces hasta que no se generan nuevos hechos.

Equiparación. Selección de reglas compatibles con la Base de Hechos por sus condiciones y acciones.

Heurística. Conjunto de técnicas para la resolución de un problema.

Heurístico, conocimiento. Aquel conocimiento no riguroso ni consensuado por los expertos de un campo sino basado en heurísticas propias de un experto o grupo de expertos.

Indexación, técnica de equiparación de. Aquella que consiste en añadir a las reglas nuevas condiciones relacionadas con el punto de inferencia.

Intérprete de reglas. Ver Motor de inferencias.

Motor de Inferencias. Mecanismo que implementa la estrategia de control de un Sistema Basado en Reglas, examina la Base de Hechos y determina qué reglas se deben disparar.

Procedimental, representación del conocimiento. Aquella que incluye la información de control necesaria para utilizar el conocimiento.

Progresivo, razonamiento. Ver Encadenamiento hacia delante de reglas.

Regla. Par que consta de un antecedente o condición, y un consecuente o acción. Equivale a un condicional IF-THEN de los lenguajes de programación.

Regresivo, razonamiento. Ver encadenamiento hacia atrás de reglas.

Resolución de conflictos. Selección de una regla de entre las resultantes de un proceso de equiparación.

Rete, algoritmo de equiparación. Técnica de aceleración del proceso de equiparación ampliamente utilizada que aumenta la velocidad de procesamiento a costa de consumir ingentes cantidades de memoria.

Reversibilidad. Modo de inferencia mixto que combina encadenamiento de reglas hacia adelante y hacia atrás.

Sistema basado en reglas. Aplicación de los sistemas de deducción en lógica proposicional restringidos a cláusulas de Horn que utiliza reglas de inferencia de la lógica para obtener conclusiones lógicas. Tiene una base de conocimiento con reglas y algún mecanismo de inferencias que selecciona las reglas que se pueden aplicar y las ejecuta con el objetivo de obtener alguna conclusión.

BLOQUE 6: Redes Semánticas y Marcos

Arco descriptivo. Arco de una red semántica que representa un atributo y une el objeto al que hace referencia el atributo con el valor asociado al atributo.

Arco estructural. Arco de una red semántica que une dos conceptos y determina la arquitectura de la red.

Clase. Marco que expresa un conocimiento genérico.

Demonio. Procedimiento asociado a una propiedad de un marco, que se encarga de recuperar, almacenar o borrar información asociada a dicha propiedad.

Equiparación. Método de inferencia en redes semánticas o marcos consistente en la comparación de datos, identificación de aquella red semántica o marco que mejor los contiene y asignación de valores.

Faceta. Característica de una propiedad o relación asociada a un marco.

Herencia de propiedades. Método de inferencia en redes semánticas o marcos consistente en que los elementos más específicos reciben propiedades y valores de los elementos más genéricos.

Instancia. Marco que constituye un ejemplo concreto de un concepto genérico.

Marco. Método de representación del conocimiento e inferencia que se centra en la representación de conceptos referidos a entidades del dominio.

Propiedad. Atributo asociado a una característica genérica de un marco, que ayuda a describir el concepto representado por dicho marco.

Red semántica. Método de representación del conocimiento e inferencia que se centra en la representación de relaciones entre entidades del dominio.

TUTORIZACIÓN EN CENTROS ASOCIADOS

En el enlace que aparece a continuación se muestran los centros asociados y extensiones en las que se imparten tutorías de la asignatura. Estas pueden ser:

- Tutorías de centro o presenciales:** se puede asistir físicamente en un aula o despacho del centro asociado.
- Tutorías campus/intercampus:** se puede acceder vía internet.

La información ofrecida respecto a las tutorías de una asignatura es orientativa. Las asignaturas con tutorías y los horarios del curso actual estarán disponibles en las fechas de inicio del curso académico. Para más información contacte con su centro asociado.

Consultar horarios de tutorización de la asignatura 71902060

IGUALDAD DE GÉNERO

En coherencia con el valor asumido de la igualdad de género, todas las denominaciones que en esta Guía hacen referencia a órganos de gobierno unipersonales, de representación, o miembros de la comunidad universitaria y se efectúan en género masculino, cuando no se hayan sustituido por términos genéricos, se entenderán hechas indistintamente en género femenino o masculino, según el sexo del titular que los desempeñe.