

Inteligencia Artificial: teoría y prácticas

Álvarez Llorente, Luis Antonio.
Departamento de Informática. Escuela Politécnica.
Universidad de Extremadura.
Avda. de la Universidad s/n. 10071 Cáceres.
e-mail: luisalva@unex.es

En el curso 95/96 arranca el ciclo superior de la Ingeniería Informática en la Universidad de Extremadura. La asignatura de Inteligencia Artificial forma parte de la troncalidad de esta carrera, recogida en los nuevos planes de estudio que se implantaron hace ahora cinco años. En el presente trabajo se trata de dar una visión general de la asignatura, reflexionando sobre diferentes aspectos que influyen en la misma: temario impartido, prácticas realizadas, número de alumnos matriculados, objetivos marcados, relaciones con otras asignaturas afines, etc.

I. Objetivos.-

La asignatura de *Inteligencia Artificial* se imparte en la Universidad de Extremadura desde el curso académico 95/96 en los estudios de Ingeniería Informática. Se trata de una materia troncal de cuarto curso, recogida en los nuevos planes de estudio que se implantaron en la Escuela Politécnica de Cáceres hace ahora cinco años. Esto significa que en este curso académico 96/97 saldrá la primera promoción de Ingenieros Informáticos de la Universidad de Extremadura.

Hasta el año 1992, los únicos estudios de Informática que contemplaba esta Universidad eran los que conducían a la obtención del título de Diplomado en Informática. La Escuela Politécnica realizó un gran esfuerzo, tanto material como humano, para poder incorporar un segundo ciclo y satisfacer la gran demanda existente en la sociedad. En la actualidad es posible cursar dos Ingenierías Técnicas y la mencionada Ingeniería Superior.

Al ciclo superior se puede acceder directamente dentro de la Ingeniería Informática o como Ingeniero Técnico de Sistemas, o bien como Diplomado en el caso de alumnos provenientes del plan antiguo. Esta forma de acceso implica que los conocimientos con los que llegan los alumnos sean muy heterogéneos.

La experiencia del año anterior nos dice que, de los 77 alumnos matriculados, aproximadamente un 40% son Diplomados que desean completar sus estudios; pero entre ellos, hay quien cursó la carrera en esta misma Escuela y quien estudió en otras Universidades; los hay que terminaron el curso anterior y los hay con más de 5 años de experiencia laboral...

Otro aspecto importante a destacar es el perfil dado a este ciclo superior. Como hemos mencionado anteriormente, la asignatura de *Inteligencia Artificial* es troncal. Sin embargo, es la única de obligado estudio relacionada con esta disciplina. Es cierto que para los más interesados en el tema se ofertan algunas materias optativas, tales como *Redes Neuronales*, *Robótica* o *Programación Declarativa*, pero en cualquier caso, no ofrecen un panorama demasiado amplio para lo extensa que puede llegar a ser la Inteligencia Artificial como disciplina.

Estos dos factores condicionan bastante los objetivos de la asignatura. Por un lado, no podemos olvidar que la mayoría de los alumnos desconocen aspectos básicos de la IA; y por otro lado, debe servir de base para todos los que deseen ampliar sus conocimientos cursando materias optativas.

Por tanto, el objetivo fundamental será dotar al alumno de unos conocimientos generales que le permitan tener una visión global de lo que es la Inteligencia Artificial. De esta forma, tendrán la base necesaria para continuar cursando otras materias optativas afines. O si deciden seguir otros perfiles, al menos conocerán las cuestiones básicas de la IA.

II. Temario de la asignatura.-

Visto el contexto en el que se desenvuelve la asignatura de *Inteligencia Artificial*, vamos a centrarnos en la misma. La carga lectiva de la asignatura es de 9 créditos anuales, repartidos en 6 créditos teóricos y 3 créditos prácticos, lo que traducido a horas significa que los alumnos reciben dos horas de teoría y una hora de prácticas semanalmente. El temario es muy amplio, pero básicamente está dividido en tres partes fundamentales :

- Adquisición y Representación del conocimiento.
- Resolución de problemas.
- Introducción a los Sistemas Expertos.

Dentro de la primera parte, se estudian diferentes formas de aprendizaje: *memorístico*, *por inducción*, *por analogía*, *por descubrimiento*, etc [1] y [2]. En todos los casos se toma como base ejemplos de la forma de aprendizaje en cuestión, de los que se extraen los conceptos básicos, la metodología general del procedimiento y diferentes conclusiones.

También se plantean los fundamentos de los *algoritmos genéticos* [2] y las *redes neuronales* [1], cuestiones en las que posteriormente se puede profundizar y ampliar conocimientos en distintas materias optativas.

De igual forma, se estudia la *lógica matemática* como método básico de representación del conocimiento [1]. La estrategia que se sigue es la de dar una idea de qué es la lógica matemática, para qué sirve y cuál es su uso dentro de la IA. Básicamente se estudian 3 algoritmos:

- Transformación de sentencias a forma clausal.
- Unificación.
- Resolución.

Para los alumnos más interesados en el tema se oferta en quinto curso una asignatura optativa, en la que pueden ampliar sus conocimientos en esta disciplina.

Como complemento a la lógica, se tratan las *redes semánticas* y los *frames (marcos)* [1] como forma de representación estructurada del conocimiento. En ambos casos, los ejemplos son básicos para entender y manejar estas dos técnicas de representación.

Finalmente, como forma de adquisición y representación de información un tanto especial, se aborda el tema del *procesamiento de lenguaje natural*. Se trata de una materia muy amplia, a la que bien se podría dedicar una asignatura optativa completa. En un recorrido básico por la IA no podemos pasar por alto este tema, pero dado que el objetivo es dotar al alumnos de conocimientos generales, se tratan los contenidos sin profundizar en las distintas técnicas que se emplean en el uso del lenguaje natural como interfaz.

La parte de **resolución de problemas**, es quizás la más amplia en cuanto a dedicación y profundidad. Básicamente está dividida en tres bloques :

- En una primera parte se revisan diferentes algoritmos que emplean la búsqueda guiada mediante heurísticas como forma habitual de trabajo, frente a otros métodos clásicos de búsqueda no guiada. Así, quedan recogidos dentro de este bloque los *algoritmos de escalada*, *enfriamiento simulado*, *A**, *AO**, etc [1]. De entre todos ellos cabe destacar por el especial interés que tiene para los alumnos el tema de la resolución de juegos, al que se dedica un capítulo.
- En un segundo bloque se estudian algunos programas clásicos dentro de la Inteligencia Artificial, como el LT (*Logic Theorist*) o el GPS (*General Problem Solver*) [3]. Se trata de programas que, como alternativa a la búsqueda heurística, emplean procedimientos que emulan ciertas técnicas del ser humano: dividir el problema a resolver en subproblemas de menor entidad, e ir solucionando cada uno de ellos comenzando por los de mayor dificultad.
- Finalmente nos centramos en otros sistemas más evolucionados que emplean la planificación como alternativa a la búsqueda, y para ello se toma como base el sistema STRIPS [1] y [3]. Se trata de un planificador muy general para el que se definen operadores básicos con los que pueda trabajar en un universo simplificado. En concreto se trabaja con un brazo de robot que apila y desapila bloques en un mundo ideal.

Con respecto a la última parte, **introducción a los sistemas expertos**, el terreno ya ha sido allanado convenientemente, puesto que ya se tiene una visión general de los aspectos fundamentales de un sistema experto: la adquisición de nuevos conocimientos, el almacenamiento y organización de los mismos, y su posterior uso para resolver problemas [4] y [5]. Se trata de dar una definición general que pueda servir para cualquier sistema experto: sus diferentes partes, características y ámbitos de uso.

III. Las prácticas como complemento fundamental.-

Hasta el momento no hemos hablado de los 3 créditos de prácticas asociados a la asignatura. Hemos pretendido cubrir en el laboratorio las diferentes partes que se abordan en las clases de teoría. La experiencia del curso pasado nos hizo ser más realistas, este objetivo que *a priori* resultaba posible a medida que el curso avanza se hace un tanto utópico.

El laboratorio en el que se desarrollaron las prácticas en el curso pasado estaba compuesto por 15 ordenadores PC 80386, con 2MB de memoria RAM y 40MB de disco duro. A pesar de no ser las máquinas más adecuadas, hay que reconocer que se les sacó un buen rendimiento... A pesar de contar con estos medios materiales tan reducidos, teniendo en cuenta que el número de matriculados fue de 77 alumnos, las prácticas de la asignatura resultaron muy interesantes y fueron seguidas por los estudiantes con mucho interés, aunque nos obligó a que se realizaran en parejas, para permitir un paso por el laboratorio más fluido.

En lo referente al software empleado, en cada una de las prácticas se trabajó con programas diferentes :

- En la primera práctica, dedicada al lenguaje Prolog, se usó un intérprete de libre distribución llamado *Arity Prolog* (figura 1). Se trata de una versión simplificada que no genera código ejecutable. Dispone de un entorno amigable, donde se puede editar la base de conocimiento, interpretar las reglas y hechos, hacer consultas sobre la base de reglas, o recibir ayudas sobre los comandos disponibles. En cuanto al lenguaje en sí, es una versión muy parecida al Prolog estándar, no como ocurre con otros lenguajes como *Turbo Prolog*, que presenta algunos aspectos propios que lo diferencian claramente del lenguaje original.



Figura 1.-

- Para la segunda práctica había que elegir un lenguaje conocido por los alumnos, que permitiera programar con cierta facilidad los algoritmos de juego estudiados en clase y en el que fuera sencillo integrar las rutinas que se entregan para la implementación de la interfaz de usuario. De entre los compiladores disponibles, se optó por la versión 3.1 de *Borland C++* (figura 2). Posee un buen entorno de desarrollo, donde se puede editar, compilar y depurar cualquier programa escrito en C tradicional o en C++. Además cuenta con la ventaja de ser conocido por todos los alumnos, puesto que ya lo han usado en prácticas de otras asignaturas.

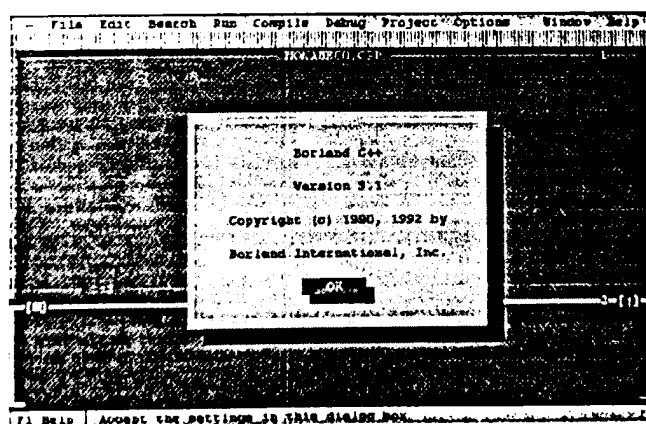


Figura 2.-

Centrándonos en las prácticas en sí, no parecía lógico que los alumnos pudieran terminar sus estudios sin conocer un lenguaje típico de la Inteligencia Artificial como *Prolog*. Además encaja perfectamente en el temario de la asignatura, ya que permite llevar a la práctica los conceptos estudiados en representación del conocimiento mediante lógica matemática o en la resolución de problemas con técnicas de *backtracking*. Por esta razón, las primeras sesiones se dedican a explicar los fundamentos de este lenguaje, que para la mayoría de los alumnos supone romper con todos los esquemas de lenguajes de programación conocidos, tales como Pascal o C++, que emplean una filosofía de trabajo muy distinta a la de Prolog.

En los primeros meses se pretende que el alumnado se familiarice con Prolog [6], comenzando con la redacción de pequeños programas y acabando con la resolución de problemas de cierta entidad. Estas experiencias se llevan a cabo en el laboratorio con la ayuda del profesor. Para finalizar con este bloque, se plantea una práctica de mayor entidad, que los alumnos deben completar por sí mismos.

En concreto, una de las prácticas que debían entregar esta año consiste en resolver en Prolog el conocido *problema del zorro, el pollo y el grano*. Se trata de mostrar la secuencia de pasos que debe llevar a cabo un granjero para atravesar un río con los tres acompañantes que lleva: un zorro, un pollo y un poco de grano. El problema es que sólo dispone de una barca en la que caben él y un acompañante, y no puede dejar solos en ninguna de las orillas al zorro con el pollo (porque se lo comerá) o al pollo con el grano (por la misma razón).

Durante este tiempo, en las clases de teoría se ha avanzado en el programa para explicar uno de los temas que los alumnos siguen con mayor interés: la resolución de juegos. En este tema se explican los dos algoritmos fundamentales de la teoría de juegos: la *regla Min-Max* y el *algoritmo de poda Alfa-Beta* [1] y [3]. De igual modo, aunque sin tanto detalle, se enumeran diferentes mejoras que se pueden introducir en estos algoritmos según el tipo de juego a resolver.

Conocidos los fundamentos teóricos de la resolución de juegos, se plantea la segunda práctica: implementar un juego inteligente, que permita que un jugador humano desafíe a la computadora. La dificultad principal era motivar al alumno, por un lado con un juego que despertara su curiosidad y fuera lo suficientemente sencillo como para que su aprendizaje y estudio no necesitasen un tiempo excesivo; y por otro que fuese innovador y en cierta medida original.

Así surgió la idea del **juego de los ceros**. Se trata de un juego relativamente conocido, y que por su sencillez y facilidad de aprendizaje, se puede empezar a jugar al instante de conocerlo. El tablero de juego (figura 3) está formado por una matriz cuadrada de $4 \times 4 = 16$ *ceros*. Entre los ceros quedan huecos que deben ser completados con *rayas*, horizontales y verticales. El propósito del juego es rellenar todos los huecos libres. Para ello, cada jugador, en su turno, coloca una *raya* (horizontal o vertical) entre dos *ceros*. Cuando un jugador, al colocar una *raya*, completa un cuadrado formado por 4 rayas, conseguirá un punto y tendrá derecho a un turno extra. El proceso se repite hasta completar el tablero. El jugador que más puntos obtenga será el ganador de la partida.

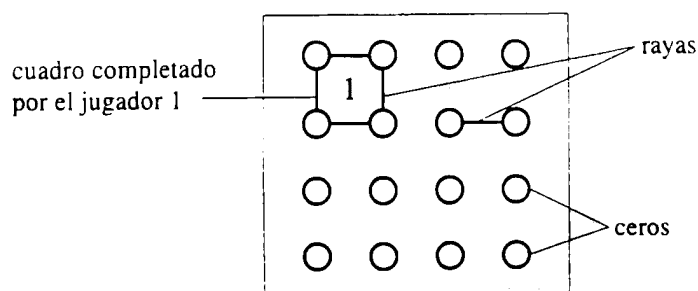


Figura 3.-

Dado que el tablero es de dimensiones 4x4, los jugadores deberán repartirse los 9 puntos que se pueden conseguir tras realizar 24 movimientos (si llamamos movimiento a colocar una *raya*). En la figura 3 podemos apreciar el tablero de juego con los 16 *ceros*. Sobre él aparecen situadas *rayas* verticales y horizontales, y un cuadro formado por sus cuatro *rayas* correspondientes.

Pero quizás todo esto no era suficiente motivación. Una vez conseguido que el programa no perdiera después de los primeros movimientos, se podía caer en un cierto desánimo, sólomente los alumnos más interesados se preocuparían por mejorar el nivel de juego. Por esta razón se pensó en una competición. En esta competición, las distintas implementaciones del juego se enfrentarían entre sí en diferentes rondas eliminatorias, de manera que la supuestamente más inteligente debía ser aquella que pudiera ganar a las demás.

El tema de la competición pasó a ser casi el fin que justificaba los medios. Conocido es en el hombre esa tendencia natural a intentar hacer las cosas mejor que los demás, y en este caso no iba a ser menos... El problema ahora era planear bien la competición para que fuera totalmente transparente y nadie se sintiera perjudicado.

Sin perder de vista los objetivos de este trabajo: poner en práctica lo aprendido en las clases de teoría sobre resolución de juegos, y teniendo en cuenta que la competición ya había provocado un cierto *pique* entre los alumnos, el hacer un programa más vistoso, con una interfaz de usuario más llamativa, podía acabar por hacer olvidar ese primer propósito. Lo ideal sería que todos los programas fueran iguales en cuanto a forma de manejo, de esa manera se conseguían dos propósitos :

1. El alumno se centra en los algoritmos que hacen que el programa sea capaz de jugar a los ceros, y se olvida de la interfaz, que no es del máximo interés en esta asignatura.
2. La prueba de las prácticas, y la propia competición, resulta más sencilla cuando todas siguen el mismo modelo de funcionamiento.

De esta forma, a cada alumno se le entrega un disquete con el enunciado de la práctica, donde se detallan los objetivos, las reglas del juego de los ceros, etc, y una serie de archivos para que los utilicen en el programa que tiene que implementar. El contenido de estos ficheros se resume en una serie de rutinas para implementar la interfaz, tanto a nivel de aspecto del tablero de juego como de uso del ratón para colocar las *rayas*. En la figura 4 se puede apreciar el aspecto que presenta el tablero de juego usando estos archivos.

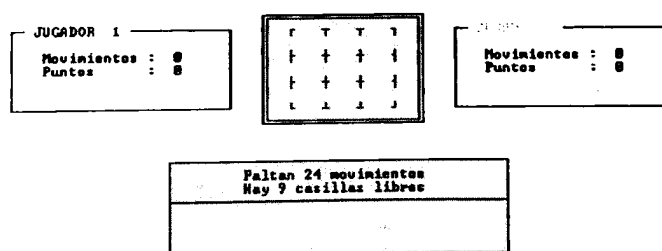


Figura 4.-

En general, las prácticas presentadas por los alumnos consiguieron un nivel de juego bastante bueno, aunque lógicamente aquellas que llegaron hasta las últimas eliminatorias eran las mejores. Los dos programas finalistas eran lo suficientemente *inteligentes* como para que a un jugador no experto le cueste trabajo derrotarlos.

Finalmente, aunque se trata más de una experiencia que de una práctica, se dedicaron algunas sesiones a conocer el funcionamiento de una red neuronal. Hemos visto en el apartado de teoría que uno de los temas estudiados estaba dedicado a las redes neuronales. Teniendo en cuenta que parte de los alumnos no volverá a estudiar este tipo de modelos, parecía lógico que pudieran experimentar con algo que visto sobre el papel puede parecer muy abstracto. Se facilita al alumno un programa que implementa una red tipo *backpropagation*. A través de un fichero de configuración pueden establecerse los patrones que queremos que aprenda. Los resultados se pueden ver directamente en pantalla.

IV. Conclusiones.-

En este trabajo hemos presentado una visión general de la asignatura de Inteligencia Artificial, que se imparte en el cuarto curso de los estudios de Ingeniería Informática en la Universidad de Extremadura. Se ha realizado un análisis del temario y de las prácticas que se desarrollaron en esta asignatura y creemos que :

- ◇ Las prácticas que suponen una interacción con el trabajo desarrollado, como es el caso de este juego, motivan mucho a los alumnos, ya que en todo momento están viendo el resultado de su esfuerzo.
- ◇ Pero además siempre es útil emplear otros alicientes, como la competición descrita en este trabajo, que lleva al alumno a mejorar sus prácticas aun cuando ya se han cumplido los objetivos básicos que se pretendían alcanzar.

Tras la experiencia de este curso, nos proponemos repetirla en años sucesivos, añadiendo y mejorando los siguientes aspectos :

- Incrementar el tiempo que los alumnos tienen para llevar a cabo las prácticas, especialmente la dedicada a la programación de juegos.
- Mejorar la interfaz de usuario facilitada en esta segunda práctica.
- En función del número de prácticas presentadas, modificar el modelo de competición.
- Impartir un seminario dedicado al otro gran lenguaje de la Inteligencia Artificial, el *LISP*. Además, sería interesante realizar alguna práctica sobre Lisp, empleando para ello algún intérprete sencillo, similar al descrito para Prolog.

V. Referencias bibliográficas.-

- [1] *Inteligencia Artificial. Segunda Edición.* Elaine Rich & Kevin Knight.
Editorial McGraw-Hill. 1994.
- [2] *Inteligencia Artificial. Tercera Edición.* Patrik Henry Winston.
Editorial Addison-Wesley. 1994.
- [3] *Principios de Inteligencia Artificial.* Nils J. Nilsson.
Editorial Díaz de Santos. 1987.
- [4] *Principios de Inteligencia Artificial y Sistemas Expertos.* D.W. Rolston.
Editorial McGraw-Hill. 1993.
- [5] *Sistemas Expertos.* Dieter Nebendahl.
Editorial Marcombo. 1988.
- [6] *Prolog: Introducción a la Programación de Sistemas Expertos.* J.M.Orenga y J.P.Sánchez.
Editorial Ra-Ma. 1987.