

Inteligencia Artificial Distribuida y Razonamiento Basado en Casos en la Arquitectura de un Sistema Basado en el Conocimiento para la Educación a Distancia

Lima Lizarraga Elohin Davis
caleb_elohein@yahoo.es

RESUMEN

El presente trabajo explora los problemas de diseño e implementación de un Sistema Basado en el Conocimiento para Educación a Distancia (SBC-ED) desde el punto de vista de la Inteligencia Artificial Distribuida (IAD) y basándose en las metodología MAS- CommonKADS para la descripción de sus componentes y del modelo Multiagentes, es así como se han distribuido las tareas propias de un Sistema Tutorial Inteligente, en un conjunto de entidades independientes que cooperan entre sí con el único fin de guiar el proceso de enseñanza aprendizaje a través una adaptación de las estrategias de enseñanza utilizadas en el proceso.

Palabras Claves: Sistema Basado en el Conocimiento para Educación a Distancia, Modelación del Conocimiento, Agentes Inteligentes, Multiagentes, metodología MAS-CommonKADS, CommonKADS, estrategia basada en casos.

1. INTRODUCCIÓN

Cada día aumenta más la aparición de sistemas de software que pretenden ayudar a mejorar el proceso educativo y la administración de organizaciones educacionales que cuentan con un número muy elevado de alumnos a distancia. Estos sistemas parecieran no estar demostrando su utilidad, ya que la dependencia del docente frente al sistema sigue siendo igual o incluso superior a la que tenía antes de usarlo, el docente se ve en la obligación de generar el material que debe incluirse en el sistema, y preocuparse de la calidad y de los derechos de autor que lo involucran. A esto hay que agregar la dependencia que crean los sistemas de estar continuamente controlando al alumno y proponiendo actividades que le permitan el logro de los objetivos propuestos en la asignatura o curso.

Lo anterior ha llevado a generar el presente trabajo, con el objetivo principal de automatizar el proceso de enseñanza aprendizaje en un organismo educacional a distancia. Para lo cual ha sido necesario modelar el conocimiento del proceso de enseñanza aprendizaje que se realiza entre un profesor con experiencia en educación a distancia, y sus alumnos, encontrando los procedimientos generales y reutilizables para pasar desde esos modelos a un sistema que consiga apoyar el proceso de enseñanza / aprendizaje de una manera más real y efectiva. Es decir se pretende lograr obtener descripciones razonablemente claras, completas, precisas e inequívocas de las tareas que el educador y el alumno realizan cuando se someten a un proceso de enseñanza aprendizaje a distancia, y así obtener un modelo al nivel de conocimiento que articule las distintas actividades y relaciones descritas por el alumno y el educador, y que permita conseguir un modelo formal

subyacente a esa descripción que nos permita enlazarla con las primitivas computables de un entorno de programación, generando un Sistema Basado en el Conocimiento (SBC-ED) para la educación a distancia.

El SBC propuesto le permite por un lado al docente, generar un curso que basado en objetivos específicos permita detallar los materiales y la estrategias de enseñanza mas adecuadas a un alumno “común”, estrategias de enseñanza que se modificarán con el paso del tiempo, y que se adaptarán a los usuarios según sus necesidades. Por otro lado el sistema es capaz de guiar al estudiante a lo largo del dominio particular del conocimiento propuesto por el profesor, resolviendo durante este proceso tareas tales como: la elaboración de una estrategia de tutorización, la propuesta de materiales complementarios al estudio de un tema, el registro de las actividades del usuario y la generación de nuevas estrategias para nuevos casos.

El enfoque del proceso se presenta a través de un modelo de casos de uso (figura 1.1), donde los actores (profesor titular, profesor asistente, alumno, base de datos del curso y base de datos del usuario) interactúa con distintos procesos. Bajo este enfoque, cada módulo de los anteriormente mencionados representará un conjunto de uno o más agentes inteligentes. Estas entidades cooperarán entre sí, con el objetivo final de alcanzar el objetivo central del sistema: la transmisión del conocimiento al estudiante y la generación de cursos de manera más automatizada a través de la generación de estrategias de enseñanza/aprendizaje que se adapten a los alumnos.

A continuación se definen los términos de la Inteligencia Artificial Distribuida (IAD) que usaremos y la metodología MAS-CommonKADS para la construcción del sistema, y se terminará con la presentación de los elementos básicos del análisis y diseño del SBC.

2. INTELIGENCIA ARTIFICIAL DISTRIBUIDA

En [Iglesias,98] se define la Inteligencia Artificial Distribuida (IAD) como “aquella parte de la IA que se centra en comportamientos inteligentes colectivos que son producto de la cooperación de diversos agentes”. Estos agentes, son las entidades que colaboran.

La descripción de un sistema desde el punto de vista de la IAD, puede hacerse de acuerdo a dos elementos fundamentales [Dem,90]. El primero se concentra en las componentes fundamentales del sistema como una composición de estos, o sea como la sociedad en que viven nuestros agentes (IAD clásica).

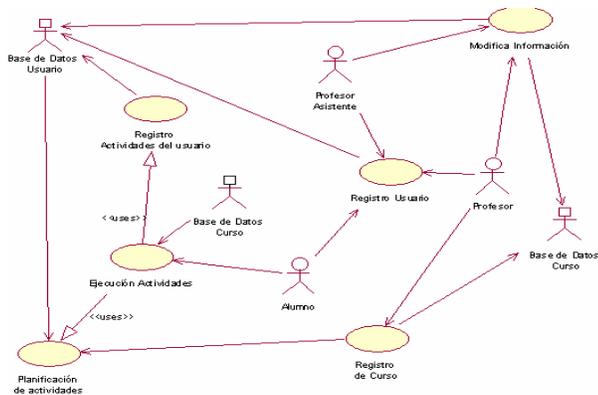


Fig. 1.1 Modelo de casos de uso para el SBC-ED

Un Agente puede definirse como una entidad, semi o completamente autónoma, la cual actúa racionalmente de acuerdo a sus percepciones del exterior y el estado de su conocimiento.

Desde un punto de vista estructural, y de acuerdo a [Ber,92], el agente puede dividirse en dos partes principales. La primera está relacionada con el conocimiento que el agente posee acerca del mundo exterior, las habilidades de otros agentes y las suyas propias, así como sus tópicos de interés; el segundo aspecto abarca el procesamiento dinámico que el agente realiza de los mensajes y que definirá su actividad de razonamiento ante problemas e informaciones nuevas. Por esto se basa en un lenguaje de comunicación externo, para interactuar con otros agentes, uno interno, para la comunicación entre sus propias componentes, y un conjunto de métodos que permite el procesamiento de estos mensajes.

La sociedad de agentes está organizada como una red en la cual los nodos representan dichas entidades, y los enlaces los puentes de intercambio de información entre los mismos. A partir de la organización de los enlaces, el intercambio de información puede hacerse o bien directamente, o bien indirectamente a través de un intermediario.

A partir del control esbozado en el párrafo anterior, se puede decir que existen dos tipos fundamentales de sociedades en la Inteligencia Artificial Distribuida: la centralizada y la descentralizada.

MAS-CommonKADS

MAS-CommonKADS es una metodología orientada a Sistemas MultiAgentes [Iglesias,98], una extensión a CommonKADS [KADS,93]. Con ella es posible describir todo el sistema basándose en representaciones gráficas y plantillas que el usuario debe rellenar, utiliza representaciones gráficas y plantillas para representar los modelos que cubren cada uno de los aspectos del sistema.

MAS-CommonKADS utiliza los siguientes modelos para el desarrollo de sistemas multiagentes: Modelo de Agente (AM), Modelo de Organización (OM), Modelo de Tareas (TM), Modelo de experiencia (EM), Modelo de Comunicación (CM), Modelo de Coordinación (CoM), Modelo de Diseño (DM) [Iglesias,98].

3. DESARROLLO DEL SISTEMA

En el desarrollo del sistema entre otros podemos identificar los siguientes modelos y agentes, los cuales representan a nuestro juicio la componente principal de un SBC-ED.

1.3 Modelo de Organización

El constituyente proceso indica el orden temporal en el que son realizadas las funciones del constituyente función y se muestra en la fig. 3.1

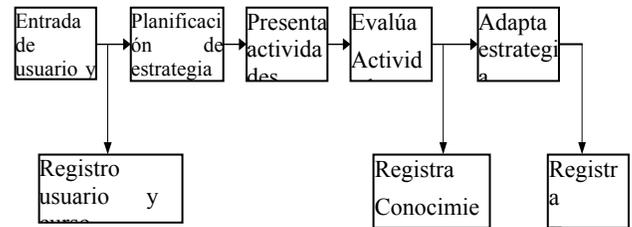


Fig. 3.1 Procesos de control de proceso de desarrollo y enseñanza / aprendizaje del SBC-EDADIS

La introducción del sistema multiagente acelerará la detección de la mejor estrategia para el objetivo y el usuario. Así mismo agilizará el proceso de desarrollo de un curso a distancia, y el proceso de aprendizaje del usuario.

1.4 Conceptuación

3.2.1 Identificación de los actores

En nuestro sistema podemos identificar varios actores que interactúan con el sistema: el Profesor, el Alumno, la Base de Datos de Usuario, la Base de Datos de Cursos y el Profesor Asistente. Aunque también podemos considerar la necesidad de acceso al sistema por personal de desarrollo para actualizarlo o mantenerlo.

1.5 Modelo de Agente

El propósito del modelo de agente es describir los agentes que participan en la resolución del problema y la repercusión del sistema en los agentes humanos.

3.3.1 Identificación de los agentes (primera-iteración)

Partiendo de los actores de la fase de conceptualización, se pueden identificar los agentes externos al sistema: Alumno, Profesor Titular, Profesor Asistente, Base de Datos del Curso, Base de Datos Usuario. Las Bases de Datos serán controladas por el sistema, y para el resto de los agentes, que son humanos, será necesario crear una interfaz gráfica diferente para cada uno, ya que en el caso del Alumno este solo interactuará con las actividades propuestas, en cambio el profesor titular ingresará objetivos, actividades y estrategias, y el profesor asistente solo modificará información.

Todos estos agentes se diferencian entre sí, exclusivamente, por la componente cognitiva, la cual permite determinar las tareas a ellos encargadas dentro de la sociedad, así como poder mostrar un comportamiento determinado. Si describimos al Agente de Pedagógico tendremos:

4. APLICACIÓN

El objetivo principal del Agente Pedagógico es la generación de planes instruccionales (estrategias de enseñanza), que se adapten a las necesidades actuales del alumno; es decir, el agente debe ser capaz de trabajar tan dinámicamente que detecte lo más

rápido posible las necesidades del alumno, y, en base a estas, variar la estrategia de enseñanza

La idea general para la construcción de las capacidades de razonamiento de este agente, tiene sus orígenes en el Razonamiento Basado en Casos (RBC). El RBC puede definirse básicamente como: “generación de soluciones a problemas actuales sobre la base de planes pasados, los cuales se almacenan como casos en una memoria, el caso cuyas características se asemejen más a las situación actual, y aplicándole estrategias que lo adapten a las necesidades actuales. Para esta selección se hace necesaria, por lo tanto, una métrica de similaridad.

En nuestro caso, la memoria de trabajo está constituida por un conjunto de elementos, a los que hemos denominado trayectorias, los cuales describen el discursar del estudiante a lo largo de todas las sesiones de trabajo. La estructura de estas trayectorias es la siguiente:

(ID,(Obj i,l, SubPlan i,l, Aprov, Term),...,(Obj i,n, SubPlan i,n, Aprov, Term))

donde:

ID: Identificador del estudiante

Objj,k: Tópico de trabajo

Aprov: Relación entre actividades correctas y total de estas propuestas

Term: Determina si el tópico de trabajo fue completamente vencido, o si el trabajo se interrumpió.

SubPlanik: Vector de la forma (present concept1, propose medios, ..., present conceptk, propose medk)

concepti: Concepto a enseñar

mediosi: Lista de ejercicios a proponer

Estas trayectorias se pueden dividir en dos clases: trayectorias standard, incluidas en la memoria de trabajo desde un inicio y aplicables a cualquier alumno, y trayectorias generadas durante el trabajo.

La métrica de similaridad cumple una función de filtraje en dos pasos. Primero se eliminan del espacio de búsqueda aquellas trayectorias que arrojen un aprovechamiento general menor del 75%, o bien no incluyan el tópico de trabajo. El segundo filtraje se encarga de seleccionar, entre el conjunto obtenido en el paso anterior, las trayectorias más semejantes a la del alumno actual, este proceso puede describirse de la siguiente manera:

1. Para la trayectoria del alumno obtener el vector V0 tal que:

$$V0 = (Obj1, concept1,1, \dots, concept1,m, \dots, Objk, conceptk,1, \dots, concept k,m)$$
2. Para cada una de las trayectorias obtenidas a partir del segundo filtro, obtener un vector Vi, semejante al V0 y formar con cada uno de ellos la relación:

$$R = \frac{\text{Total_de_indices_coincidentes}}{\text{Longitud_del_vector}}$$

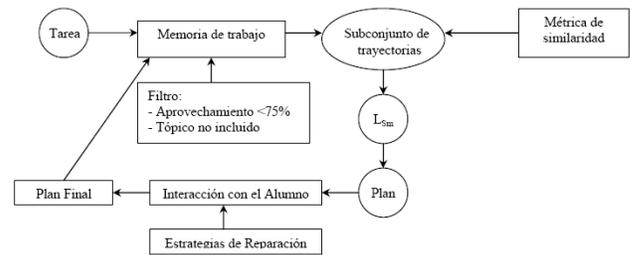
donde un índice representa una componente del vector, y su longitud es la cantidad total de componentes.

3. Ordenar descendientemente las RV0,Vi anteriores en una lista resultado LSIM.

De acuerdo a este algoritmo, queda claro que la trayectoria más similar a la analizada, será la correspondiente al primer elemento Lsim.

De esta trayectoria tomaremos entonces el subplan instruccional correspondiente al tópico de trabajo, el cual se comenzará a aplicar al estudiante.

Durante la sesión de trabajo, la cual constituye un test de efectividad para el subplan seleccionado, el mismo estará expuesto a variaciones, que se realizan a través de estrategias de reparación. Los cambios se realizan a partir de las componentes de las trayectorias standar y las contenidas en Lsim, las cuales se combinan teniendo en cuenta el grado de similaridad determinado y la efectividad que alcance el estudiante.



De esta manera, el plan instruccional de un alumno cualquiera se genera dinámicamente a partir de: subplanes contenidos en trayectorias de la memoria de trabajo, y variaciones a estos planes de acuerdo a estrategias. El subplan final se almacena en la trayectoria del alumno. La figura 3.2 muestra el proceso completo.

Fig. 3.2 Proceso de formación de una estrategia de enseñanza

1.6 Modelo de Diseño

Gracias a la posibilidad que tiene esta plataforma de adaptar la estrategia de enseñanza, los cursos definidos con ella se adaptan a los estudiantes teniendo en cuenta tanto sus características propias psicosociales, como el conjunto de acciones que realizan durante el proceso de aprendizaje. Existe una estructura asociada con cada estudiante en su interacción con el sistema, que se restaura al inicio de cada sesión. Los cursos gestionados por el sistema se definen en términos de Tareas Docentes y Reglas. Las Reglas especifican la(s) relación(es) entre tareas que, a su vez, corresponden a unidades conceptuales definidas por el profesor del curso.

La arquitectura del sistema está basada en el modelo Web estándar, en el que el servidor recibe peticiones de los estudiantes a través de los navegadores.

Hay un proceso por cada estudiante conectado al sistema, que controla el aprendizaje del estudiante durante toda la sesión. Si un mismo estudiante está siguiendo más de un curso, entonces existirá un proceso por cada uno de los cursos.

Los principales módulos del sistema son los que administra el Agente Coordinador, y se encuentran situados en el servidor al que se conectan los estudiantes, y son accesibles a través de un programa CGI.

Para que el sistema esté operativo, el Gestor de Procesos debe estar siempre activo, esperando peticiones procedentes de los estudiantes.

Cuando el CGI recibe una petición del cliente, envía los parámetros recibidos al Gestor de Procesos y se mantiene a

espera de una respuesta por parte del Agente de Interfaz. Cuando recibe las páginas HTML generadas se las envía al estudiante.

Tanto la estrategia preferida por el estudiante para el aprendizaje como los datos personales del mismo se obtienen a través de un test que el estudiante realiza la primera vez que accede a un curso, y son almacenados por el sistema para su uso tanto en la sesión actual como en sesiones posteriores. El Agente Coordinador proporciona la información sobre las acciones llevadas a cabo por el estudiante y sus resultados (el número de páginas visitadas, el número de actividades logradas, el número de ejercicios resueltos correctamente, etc.) al Agente Pedagógico el cual genera la estrategia en base a un razonamiento basado en el caso más efectivo a sus características (RBC). El Agente Coordinador a su vez en comunicación con el Agente Material determinará las actividades a seguir del estudiante en un momento y espacio dado según su perfil.

Por otro lado es el Agente Evaluador el que en comunicación con los contenidos del curso y con el Agente Coordinador, determinará las evaluaciones más adecuadas a un objetivo determinado, según el perfil del usuario como no logrado.

Finalmente el Agente Coordinador envía al Agente de Interfaz los parámetros asociados con la tarea activa, que se utilizarán para generar dinámicamente las páginas HTML que recibirá el estudiante. Dichos parámetros están relacionados con el perfil y con las acciones del estudiante.

5. CONCLUSIONES

El SBC-ED permite la creación y seguimiento de cursos adaptativos a través de Internet. La estructura y el contenido de los cursos se gestionan de forma independiente, facilitando los procedimientos de adaptación a cada estudiante, así como la reutilización y el mantenimiento de la información. La generación automática de páginas hace posible el mantenimiento de la coherencia en las páginas de menús, incluso después de producirse cambios en la estructura del curso.

La estructura de cada curso se compone de un conjunto de objetivos, tareas y reglas docentes, mientras que los contenidos están formados por actividades a cumplir y elementos multimedia que se utilizan para generar dinámicamente las páginas HTML que se presenta a los estudiantes.

Este sistema representa un avance importante en la capacidad de adaptación de la estrategia de enseñanza y en la capacidad de adaptación de los cursos desarrollados, al permitir la elección dinámica de actividades que persiguen un objetivo a lograr.

Este enfoque resulta de vital importancia para la creación de una plataformas inteligentes de educación, pues provee de un entorno interactivo de gran capacidad de reacción. Esto se alcanza debido a la distribución de tareas a agentes que trabajan de forma concurrente; esto significa, por ejemplo, que mientras el alumno realiza los ejercicios finales de un tema, el agente pedagógico planificador esta decidiendo la estrategia de enseñanza a seguir con los próximos conceptos.

6. BIBLIOGRAFIA

- [1] Y. Demazeau & Mueller, "Decentralized Artificial Intelligence", en "Decentralized AI", Demazeau & Mueller eds., North-Holland/Elsevier, pp 3-13, 1990.
- [2] Y. Demazeau & O. Boissier, "A Distributed Artificial Intelligence View on General Purpose Vision Systems", en "Decentralized AI 3", Demazeau & Werner eds., North-Holland/Elsevier, pp 311-330, 1992.
- [3] Y. Demazeau, "Distributed Artificial Intelligence & Multi-Agent Systems". Conferencia impartida en el X Simposio Brasileño de Inteligencia Artificial, 1993.
- [4] C.A. Iglesias, J.C. González and J.R. Velasco. "MIX: A general purpose multiagent architecture" en Intelligent Agent II: Agent Theories, Architectures, and Languages, volume 1037 of Lecture Notes in Artificial Intelligence, pages 251-266. Springer-Verlag, 1996.
- [5] C.A. Iglesias, "Definición de una Metodología para el Desarrollo de Sistemas Multiagentes. Tesis Doctoral", Febrero 1998, Departamento de Ingeniería de Sistemas Telemáticos de la Universidad Politécnica de Madrid.
- [6] R. De Hoog, R. Martil, B. Wielinga, R. Taylor, C. Bright, and W. van de Velde. The CommonKADS model set. ESPRIT Project P5248 KADS-II/M1/DM..1b/UvA/018/5.0, University of Amsterdam, Lloyd's Register, Touche Ross Management Consultants & Free University of Brussels, Dec. 1993.
- [7] Morales, "Sistemas Tutoriales Basados en Casos", en Informática 94, Cuarta Convención Internacional y Feria de Software, La Habana, Cuba, Febrero 1994.
- [8] F. Morales, "Implementando Cooperación en un Mundo MultiAgente". Reporte Interno de Investigaciones. Universidad de La Habana, Fac. Matemática y Computación. 1995.
- [9] Pedro Salcedo Lagos Departamento de Metodología de la Investigación e Informática Educativa, Facultad de Educación, Universidad de Concepción CHILE