

Sistemas de Razonamiento Basado en Casos para el Soporte a la Toma de Decisiones.

Laza R., Fernandez F., Corchado J.M.
E.U.I.T. Informática de Gestión
Universidad de Vigo, Campus As Lagoas 32004, Ourense, España
Email {[corchado,rlaza,riverola](mailto:corchado,rlaza,riverola@ei.uvigo.es)}@ei.uvigo.es
Tlf.: +34 988387010 Fax.: +34 988387001

Palabras Clave: Agente, Sistemas de Razonamiento Basado en Casos, Red Neuronal de funciones de base radial.

RESUMEN: *En este artículo se presenta un sistema de razonamiento basado en casos (CBR, Case Based Reasoning) diseñado para aconsejar a los estudiantes universitarios sobre qué asignaturas optativas cursar. La finalidad es desarrollar una herramienta de apoyo en la elección de asignaturas optativas, que incorpora información acerca de los conocimientos y preferencias académicas de los estudiantes que han cursado al menos un curso en la universidad. La herramienta utiliza posteriormente esta información para aconsejar a nuevos estudiantes. El rendimiento del sistema obtenido se compara con los resultados proporcionados por una red neuronal artificial de funciones de base radial.*

1.- INTRODUCCIÓN.

En la Escuela Superior de Ingeniería Informática de la Universidad de Vigo, se ha desarrollado un sistema multi-agente para aconsejar a los más de 700 alumnos del centro, sobre qué asignaturas optativas cursar. Identificar y seleccionar las asignaturas que son más interesantes o adecuadas para un estudiante es una tarea compleja, la cual se ha automatizado con un sistema consejero multi-agente (S.A.E.A.O. Sistema de Ayuda a la Elección de Asignaturas Optativas).

SAEAO posee agentes profesor, agentes estudiantes, agentes de contacto (facilitadores) y un agente consejero. Cada agente tiene una base de conocimiento, el agente estudiante almacena la información del alumno que está solicitando consejo y actúa como su asistente, el agente profesor almacena las características de la asignatura que imparte el profesor y también actúa como su asistente. El agente consejero posee la capacidad de aconsejar a los estudiantes acerca de qué asignaturas optativas deben cursar. Los facilitadores son agentes especiales que se encargan de dirigir la comunicación entre los otros agentes. (Corchado *et al.*, 1999).

En la Figura 1 se puede ver la arquitectura propuesta para el sistema SAEAO. Los diversos agentes están organizados mediante coordinación asistida, es decir, los agentes no se comunican directamente, sino que lo hacen por medio de facilitadores. Los agentes utilizan ACL (*Agent Communication Language*) para comunicar sus necesidades y habilidades a su facilitador local, quien se encarga de encontrar la ruta correcta por la cual proporcionan solicitudes a otros facilitadores, quienes a su vez pasan las solicitudes a alguno de los agentes de su dominio que puede satisfacer la solicitud (Genesereth *et al.*, 1994).

El objetivo de este artículo es mostrar cómo el agente consejero del sistema SAEAO, empleando un sistema de razonamiento basado en casos (CBR – *Case Based Reasoning*) puede aconsejar a estudiantes de forma autónoma. El artículo presenta de forma general el concepto de CBR;

posteriormente se presenta su desarrollo en el ámbito de esta investigación y finalmente muestra los resultados obtenidos con él y los compara con los resultados arrojados por una red de funciones de base radial (RBF – *Radial Basis Functions*).

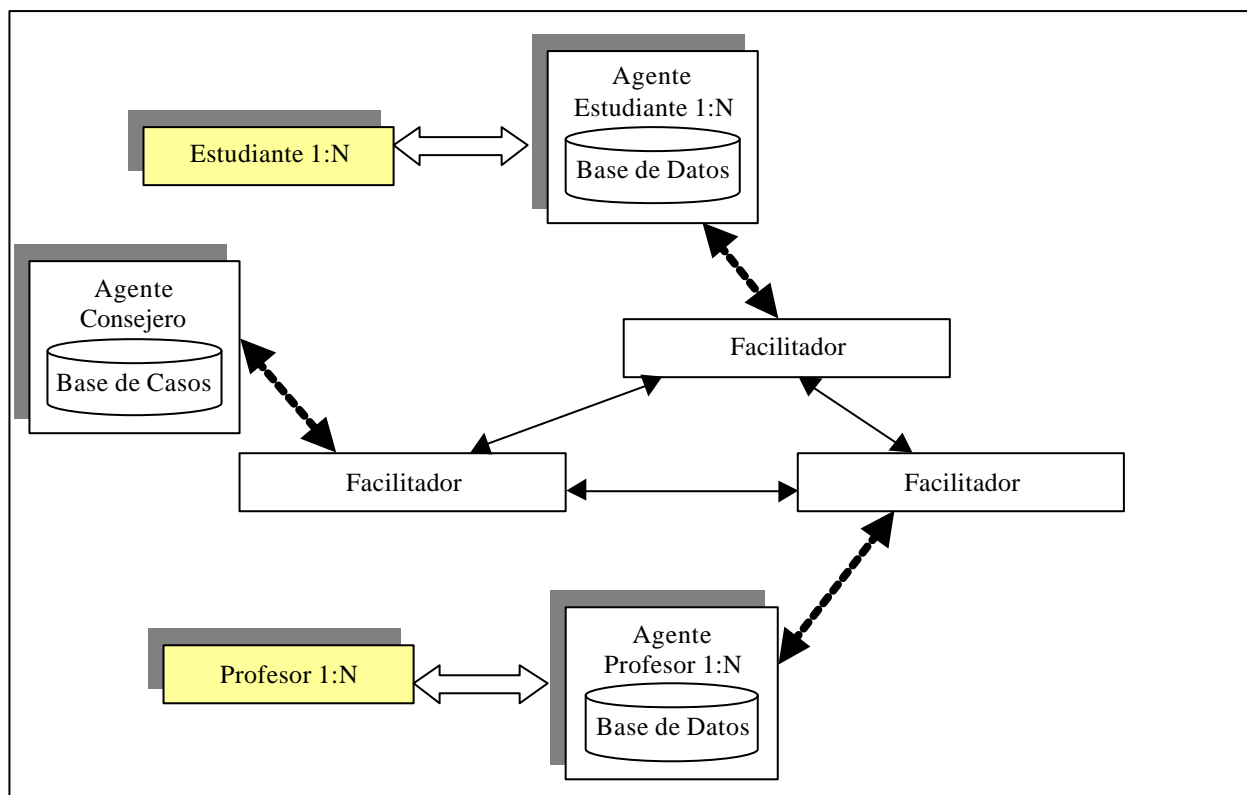


Figura 1: Sistema Multiagente SAEAO

2.- SISTEMAS DE RAZONAMIENTO BASADOS EN CASOS

Los sistemas expertos (KBS, *Knowledge Bases Systems*) constituyen una de las ramas de la Inteligencia Artificial (AI, *Artificial Intelligence*) que ha proporcionado más éxitos. Según Watson (1994) existen al menos 2000 sistemas expertos utilizados en la industria de hoy en día. Sin embargo, el desarrollo de estos sistemas se ha encontrado con varios problemas importantes (Watson *et al.*, 1994):

- En muchos casos es difícil extraer de expertos (humanos) el conjunto de leyes y normas que nos permitan crear un sistema inteligente.
- La implementación de sistemas expertos es compleja.
- Una vez implementados suelen ser lentos y tienen problemas para acceder a grandes volúmenes de información y manejarla.
- Son difíciles de mantener.

Con la intención de superar este problema, Kolodner (1983a, 1983b) propuso un nuevo modelo: Los Sistemas de Razonamiento Basados en Casos (CBR) que utilizan como base el modelo de razonamiento humano (Joh, 1997). La idea que impulsó el desarrollo de esta metodología se basa en el hecho de que los humanos utilizan lo aprendido en experiencias previas para resolver problemas presentes. Este hecho se experimenta de forma diaria y ha sido probado empíricamente por varios psicólogos (Klein *et al.*, 1988; Ross, 1989). Desde entonces, los CBR se han aplicado con éxito a un amplio espectro de problemas (Kolodner, 1993).

Un sistema de razonamiento basado en casos resuelve un problema por medio de la adaptación de soluciones dadas con anterioridad a problemas similares (Riesbeck *et al.*, 1989). La memoria del CBR almacena un cierto número de problemas junto a sus correspondientes soluciones (caso). La solución de un nuevo problema se obtiene recuperando casos (o problemas) similares almacenados en la memoria del CBR.

2.1.- Estructura General del Ciclo de Vida de un CBR.

Los CBR analizan problemas y obtienen la solución a estos problemas por medio de algoritmos de indexación, recuperación de problemas previamente almacenados, técnicas de comparación y adaptación de problemas a una determinada situación. Para ello se basan en el conocimiento almacenado en su memoria en la forma de casos o problemas. Todas esas acciones están estructuradas y se pueden representar por una secuencia cíclica de procesos que requiere generalmente la intervención humana. El razonamiento basado en casos puede ser usado por sí mismo o como parte de otro sistema convencional o inteligente.

Un sistema típico CBR está compuesto por cuatro etapas secuenciales que se invocan siempre que sea necesario resolver un problema (Kolodner, 1993; Aamodt, 1994; Watson *et al.*, 1997). La Figura 2 muestra el ciclo de vida de un sistema típico de Razonamiento basado en Casos. Dicho proceso cíclico está compuesto por cuatro pasos fundamentales:

- **Recuperación** de los casos o problemas más relevantes.
- **Adaptación** de los casos o problemas con la intención de solucionar el problema presente.
- **Revisión** de la solución propuesta si es necesario.
- **Almacenamiento (Aprendizaje)** de la solución propuesta como parte de un nuevo caso.

La misión del algoritmo de *recuperación* consiste en buscar y seleccionar en la memoria del CBR los casos más similares al problema presente. Las soluciones de los casos seleccionados son *adaptadas* para generar una posible solución (que en este contexto es el conjunto de asignaturas optativas a cursar). La solución propuesta en esa etapa se *revisa* y finalmente se crea un nuevo caso que se *almacena* en la memoria. Un CBR es un sistema de aprendizaje incremental puesto que cada vez que se resuelve el problema, es posible crear un nuevo caso y almacenarlo en la memoria del CBR para su posterior utilización.

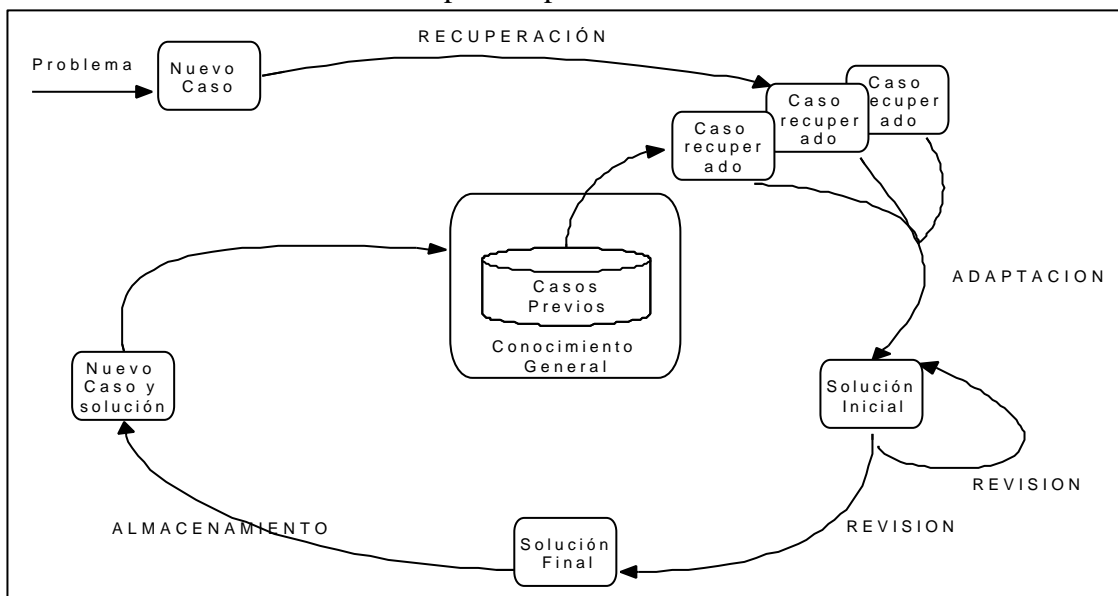


Figura 2: Ciclo de Vida del CBR.

3.- CBR PARA ACONSEJAR A ESTUDIANTES UNIVERSITARIOS.

En el sistema CBR (Figura 2) cada caso se representa por dos vectores: el *vector Problema* y el *vector Solución*, los cuales se comentan en la siguiente sección.

3.1.- Representación de un Caso.

El *vector Problema* está formado por un número de valores que proporcionan una idea de los conocimientos y preferencias académicas de un estudiante determinado.

Los parámetros almacenados en dicho vector se obtienen de las respuestas proporcionadas por un alumno al cubrir el cuestionario (Perfil del alumno) presentado en la Tabla 1.

1. ¿Qué asignaturas has aprobado en años anteriores? Ej. Bases de Datos, Ingeniería del Software, Introducción a la programación, etc.	6. ¿Qué áreas de interés te gustan más? Ej. Algoritmos y Estructuras de Datos, Arquitectura, Inteligencia Artificial, Bases de Datos, etc.
2. ¿Has asistido a algún seminario, si es así, a cuál? Ej. Programación Avanzada, Introducción a red Novel, etc.	7. ¿Hablas algún idioma extranjero? Ej. Inglés, Francés, Alemán.
3. ¿Qué lenguajes de programación son los que mejor conoces? Ej. Pascal, C/C++, Java, etc.	8. Te gustaría trabajar como: Ej. Programador, Analista de sistemas, Director de proyectos.
4. ¿Qué sistemas operativos son los que mejor conoces? Ej. Unix, MsDos, Windows, etc.	9. ¿Quieres continuar tus estudios con el segundo ciclo? Ej. Sí, No.
5. ¿Qué tipo de ordenador prefieres? Ej. Pc, Wokstation, Mackintosh, etc	

Tabla 1: Cuestionario utilizado para identificar el perfil de un alumno.

El vector Problema está representado por valores booleanos, de forma que si la respuesta ha sido seleccionada por el alumno tomará el valor 1 y sino tomará el valor 0.

$$\text{Vector Problema: } P = \{p_1, \dots, p_n\}$$

El valor del subíndice “n” variará dependiendo del número de respuestas, es decir, el cuestionario puede sufrir modificaciones (hay asignaturas que pueden desaparecer, surgir otras nuevas, se pueden modificar las conferencias,...).

El *vector solución* está formado por las asignaturas optativas aprobadas por un estudiante. Por lo tanto, estas asignaturas deben ser diferentes de las seleccionadas en el vector Problema que son asignaturas aprobadas en años anteriores.

$$\text{Vector Solución: } S = s_j$$

Como hay muchas posibilidades a la hora de cursar asignaturas optativas, el vector solución contendrá un número secuencial (s_j) que representa un conjunto de asignaturas optativas.

Los diferentes casos que están almacenados en la Base de Casos, están indexados para facilitar su recuperación. La indexación se realiza teniendo en cuenta las asignaturas obligatorias que ya ha aprobado el alumno. Los casos están agrupados de la siguiente forma:

- Todo primero aprobado.
- Todo primero y segundo aprobado.

superior a 0.7 (valor estimado heurísticamente) los cuales serán casos bastante similares al nuevo caso.

3.2.2 Adaptación.

Las soluciones de los casos recuperados en la fase anterior se analizan y ordenan de mayor a menor en número de repeticiones. Cada una de las asignaturas que forman la solución inicial debe ser analizada para comprobar si la puede cursar el alumno que solicita consejo, para ello se compara el perfil del alumno (nuevo caso) con la caracterización de la asignatura. Si alguna asignatura no es adecuada para que la curse el alumno se elimina y se selecciona otra asignatura de la siguiente solución en el orden establecido anteriormente.

A cada asignatura se le asocian una serie de atributos que la definen. Los valores de esos atributos los determina el profesor responsable de la asignatura. Se usa el cuestionario mostrado en la Tabla 2 para dicha caracterización:

<p>1. ¿Qué asignaturas de años anteriores están relacionadas con esta asignatura? Ej. Bases de Datos, Ingeniería del Software, Introducción a la programación, etc.</p> <p>2. ¿Sería recomendable asistir a algún seminario impartido en la Universidad, si es así, a cuál? Ej. Programación Avanzada, Introducción a red Novel, etc.</p> <p>3. ¿Qué lenguajes de programación son necesarios conocer para superar la asignatura? Ej. Pascal, C/C++, Java, ninguno, etc.</p> <p>4. ¿Qué sistemas operativos son convenientes conocer? Ej. Unix, MsDos, Windows, ninguno, etc.</p> <p>5. ¿Qué tipo de hardware es necesario?</p>	<p>Ej. Pc, Wokstation, Mackintosh, ninguno, etc.</p> <p>6. ¿En qué áreas de interés centrarías esta asignatura? Ej. Algoritmos y Estructuras de Datos, Arquitectura, Inteligencia Artificial, Bases de Datos, etc.</p> <p>7. ¿Sería conveniente conocer algún idioma extranjero? Ej. Inglés, Francés, Alemán.</p> <p>8. ¿Qué trabajo se podría desempeñar al cursar esta asignatura? Ej. Programador, Analista de sistemas, Director de proyectos.</p> <p>9. Horario de la asignatura.</p>
---	--

Tabla 2: Cuestionario utilizado para caracterizar asignaturas.

Las respuestas a este cuestionario formarán el vector que define el perfil de una asignatura optativa. La función de similitud utilizada para comparar el perfil de la asignatura optativa y el perfil del alumno es la misma que la empleada en la fase de recuperación, la función coseno:

$$\text{Cos}(P, N) = \frac{\sum_{i=1}^n (p_i * n_i)}{\sqrt{\sum_{i=1}^n p_i^2} * \sqrt{\sum_{i=1}^n n_i^2}}$$

Donde P es el vector que caracteriza a una asignatura y N es el vector que caracteriza al alumnos que solicita consejo.

Si al aplicar la función coseno entre los vectores P y N ésta toma un valor inferior a 0.7 se desechará la opción de cursar la asignatura optativa que caracteriza el vector P y se tomará otra asignatura de las soluciones recuperadas en la fase anterior. Las asignaturas que pasen la fase de adaptación serán las que se propongan al alumno que ha solicitado el consejo.

La función coseno se emplea porque lo que interesa es seleccionar las características comunes entre dos vectores. Por ejemplo si al cubrir el cuestionario de la Tabla 2, el profesor decide que

es conveniente para cursar la asignatura optativa ‘*Entornos de Programación*’, que el alumno haya aprobado las asignaturas de ‘*Estructuras de Datos y de la Información*’, ‘*Fundamentos de Programación*’, ‘*Metodología y Tecnología de la Programación*’ y ‘*Ampliación de Estructuras de Datos y de la Información*’; sólo interesan esas asignaturas y no el resto.

3.2.3.- Revisión.

Los datos extraídos del cuestionario presentado en la fase de adaptación también se usan para identificar incompatibilidades entre asignaturas. Puede ocurrir que algunas de las asignaturas optativas se impartan a la misma hora, o que con la solución propuesta el alumno no curse los créditos de optativas necesarios para conseguir su titulación. Es posible que en la solución sólo se propongan dos asignaturas optativas y, sin embargo, este nuevo alumno necesita cursar tres asignaturas para alcanzar los créditos necesarios. Por lo tanto, sería en esta fase en donde se eliminan asignaturas optativas si se solapan en el horario y se pasa a la fase de adaptación para seleccionar otra asignatura; y así garantizar los créditos que necesita el alumno y que no haya incompatibilidades de horario entre las asignaturas aconsejadas.

También podría ocurrir que al alumno se le propusiesen asignaturas de más, en cuyo caso se eliminarían las que excedieran en el número de créditos. Para eliminar asignaturas se borrarían aquellas que menos veces aparecen en las soluciones obtenidas en la fase de recuperación.

3.2.4.- Aprendizaje.

Al CBR se incorporan detalles relativos al éxito de la solución propuesta. Una vez que el estudiante ha finalizado su año académico puede ocurrir que haya superado las asignaturas optativas o no. En el caso de que las haya superado, este caso se almacena en la Base de Casos del CBR así podrá ser utilizado para aconsejar a otros alumnos cuyo perfil sea parecido a este.

Los casos almacenados son eliminados de la base de casos cuando contienen asignaturas que desaparecen y no se pueden reemplazar por otras nuevas; también cuando se modifican los cuestionarios que aparecen en las tablas 1 y 2. Cuando aparecen asignaturas nuevas el profesor solicita crear casos prototipos, por eso las nuevas asignaturas también se pueden recomendar a los estudiantes.

En este sistema el CBR sólo aprende del éxito de la solución propuesta, esto es debido a la naturaleza del problema. El hecho de que un alumno apruebe una asignatura depende de sus conocimientos, de sus gustos pero sobre todo del esfuerzo que él realiza, lo cual es muy difícil de medir; por ello se decidió que el CBR sólo tuviese en cuenta aquellos casos cuya solución propuesta son asignaturas optativas aprobadas. El fracaso de un estudiante puede ser por muchos motivos (problemas familiares, estado de ánimo, ...); sin embargo, el éxito de un estudiante es conseguido gracias a su esfuerzo diario. Por lo tanto, a un alumno se le pueden aconsejar unas asignaturas acordes a sus conocimientos y gustos, pero el hecho de que las apruebe sólo depende de él.

4.- UTILIZACIÓN DE LA RED NEURONAL RBF.

Las redes neuronales artificiales de Funciones de Base Radial (RBF) son aproximadores universales. Se han utilizado para realizar la comparación porque son algoritmos que modelan problemas a partir de conjuntos de datos, especialmente aplicables a problemas de clasificación donde se manejan datos y conocimiento existente del problema. Englobadas dentro de las redes de aprendizaje supervisado, la estructura una RBF es muy similar a la arquitectura del perceptrón

multi-capas (MLP, *MultiLayer Perceptron*), sin embargo, su forma de aprendizaje es sensiblemente diferente (Haykin, 1999).

La topología de las redes RBF está representada en la Figura 2 y se caracteriza por:

- La capa de entrada de la red de funciones de base radial es un receptor para los datos de entrada (*perfil del alumno*).
- La capa intermedia realiza una transformación no lineal del espacio de entrada al espacio del nivel intermedio. Las neuronas del nivel intermedio son las funciones base para los vectores de entrada. Estas funciones son normalmente Gaussianas, y sus medias y desviaciones estándar pueden calcularse teniendo en cuenta los vectores del espacio de entrada.
- La capa de salida calcula la combinación lineal ponderada de las salidas de las neuronas del nivel intermedio (*número de asignaturas optativas que el alumno debe cursar*).

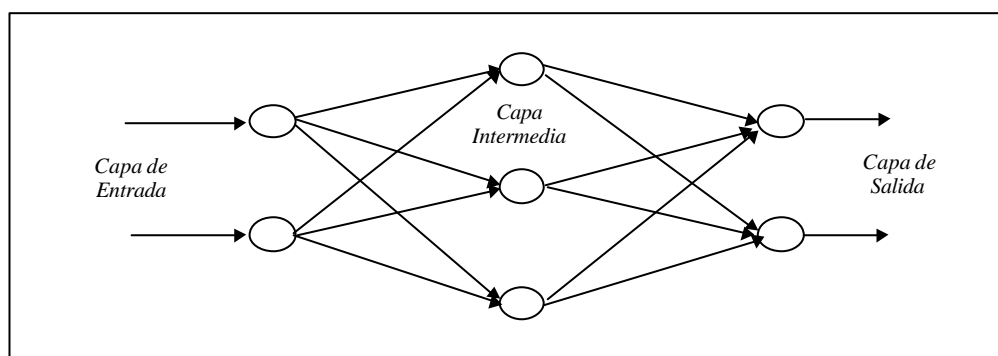


Figura 2: Topología Redes de Funciones de Base Radial.

5.- RESULTADOS Y CONCLUSIONES.

Para la implementación del problema definido anteriormente, se ha utilizado una red neuronal artificial RBF con las características que se mencionan en la Tabla 3.

Capa de Entrada:	50 neuronas.
Capa Intermedia:	75 neuronas.
Capa de Salida:	5 neuronas.
Ratio de aprendizaje:	0,45. (disminuyendo hasta 0 a lo largo del aprendizaje).
Momento:	0,9. (disminuyendo hasta 0 a lo largo del aprendizaje).

Tabla 3: Parámetros de configuración para la RBF.

Donde cada neurona de la capa de entrada se corresponde con un dato del *vector Problema*, y las neuronas de la capa de salida representan la codificación en binario del *vector Solución*. La red se entrenó con 600 vectores durante más de 5000 iteraciones hasta que el error medio de la red fue menor del 10% en un conjunto de validación de 30 vectores.

Los resultados obtenidos con el CBR son prometedores ya que empíricamente se ha comprobado que esta herramienta ha disminuido el fracaso escolar en casi un 12% y que proporciona mejores resultados que otros modelos, como las redes neuronales de funciones de base radial. En el experimento realizado con 21 estudiantes, éstos se dividieron en tres grupos de 7 alumnos de forma que la media del expediente académico de los tres grupos fuera idéntica entre sí. Se ha observado que los alumnos que siguieron el consejo del CBR han suspendido un 12% menos de asignaturas que los que no recibieron ningún consejo del CBR ni de la red neuronal y que los que

han seguido el consejo de la red han suspendido un 3% menos de asignaturas de los que no han recibido ningún consejo. Aunque los resultados obtenidos proceden de una muestra muy pequeña, creemos que son muy prometedores debido al alto grado de satisfacción de los alumnos que han utilizado esta herramienta. Los consejos propuestos por el CBR han sido contrastados con consejos pedidos a profesores del departamento y han coincidido en una media de un 83%.

La herramienta empleada por los alumnos y profesores del centro se denomina S.A.E.A.O. (Sistema de Ayuda a la Elección de Asignaturas Optativas), la aplicación (Figura 3) se utiliza para crear o modificar el cuestionario del alumno y de la asignatura, para identificar al profesor responsable de una asignatura, etc. Y desde la página web se establece la comunicación entre el agente y el alumno, el agente y el profesor (Figura 4).

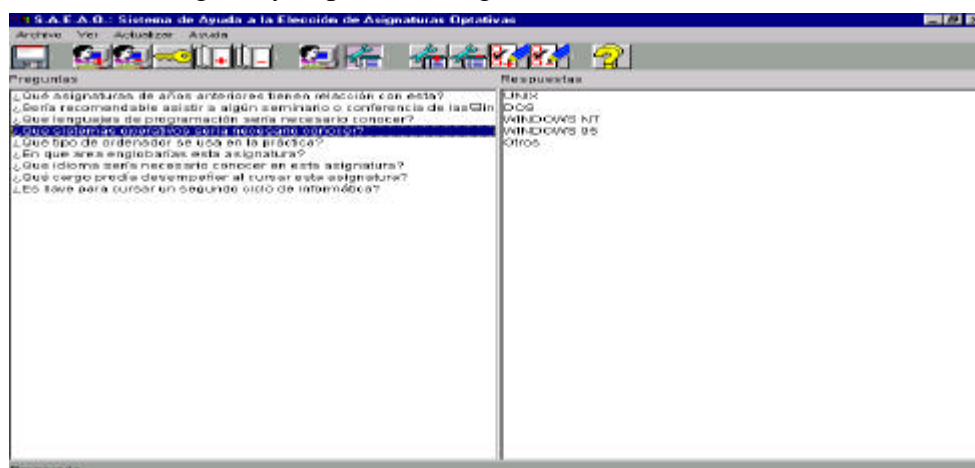


Figura 3: Aplicación gestionada por el administrador.



Figura 4: Página para establecer la comunicación entre el agente-alumno, agente-profesor.

En este artículo se ha presentado un método para la ayuda a la toma de decisiones sobre el problema expuesto anteriormente. Esta técnica se muestra válida para concluir resultados positivos. El sistema de razonamiento basado en casos muestra una mayor precisión a la hora de seleccionar el conjunto de asignaturas que la opción basada exclusivamente en la aplicación de

una red neuronal artificial. Las diferencias se encuentran principalmente en el método utilizado por el CBR, el cual filtra la información disponible en su base de casos, centrándose solamente en aquellos que pertenecen al mismo grupo que el caso de entrada, además el CBR en su fase de adaptación garantiza que las asignaturas aconsejadas son adecuadas para que las curse el alumno. La información utilizada por el CBR para la conclusión de resultados es más ajustada que en el paradigma neuronal, donde la red es entrenada con el total de casos disponibles.

REFERENCIAS.

- Aamodt A., Plaza E. (1994): *Case-Based Reasoning: foundational Issues, Methodological Variations, and System*
Corchado J.M, Laza R., Cuesta P., Barreiro E., González E., Carrión P.(1999): *Adaptative Multi-Agent System Architecture for Co-operative Problem Solving*. Modeling Autonomous Agents in a Multi-Agent World (MAAMAW99).
- Genesereth, M.R., Ketchpel, S.P. (1994). *Software agents*. *Communications of the ACM* 37, 7, pp. 48-53.
- Haykin, S., (1999): *Neural Networks: a Comprehensive Foundation*. 2nd Edition, Prentice Hall.
- Joh D. Y. (1997): *CBR in a Changing Environment*. Case Based Reasoning Research and Development. ICCBR-97. Providence, IR, USA.
- Klein G. A., Calderwood R. (1988): *Using Analogues to Predict and Plan*. Proceedings of a Workshop on Case-Based Reasoning. Pp. 224-232.
- Kolodner J. (1983a): *Maintaining organization in a dynamic long-term memory*. *Cognitive Science*. Vol. 7. Pp. 243-280.
- Kolodner J. (1983b): *Reconstructive memory, a computer model*. *Cognitive Science*. Vol. 7. Pp. 281-328.
- Kolodner J. (1993): *Case-Based Reasoning*. San Mateo. CA, Morgan Kaufmann. 1993.
- Riesbeck C.K., Schank R.C. (1989): *Inside Case-Based Reasoning*. Lawrence Erlbaum Ass. Hillsdale.
- Ross B.H. (1989): *Some psychological results on case-based reasoning*. K. J. Hammond, Proceedings of the Case-Based Reasoning Workshop. pp. 318-323. Pensacola Beach, Florida. Morgan Kaufmann.
- Watson I., Marir F. (1994): *Case-Based Reasoning: A Review*. Cambridge University Press, 1994. The knowledge Engineering Review. Vol. 9. N°3.