

SISTEMAS TUTORES INTELIGENTES: REDES NEURONALES PARA SELECCIÓN DEL PROTOCOLO PEDAGÓGICO

Fernando Salgueiro^{1,2}, Zulma Cataldi¹, Fernando Javier Lage¹, Ramón García-Martínez^{2,3}
liema@fi.uba.ar, rgm@itba.edu.ar

1. LIEMA - Laboratorio de Informática Educativa y Medios Audiovisuales. Facultad de Ingeniería. UBA. ^φ
2. Centro de Ingeniería del Software e Ingeniería del Conocimiento. Escuela de Postgrado. ITBA. ^θ
3. LSI - Laboratorio de Sistemas Inteligentes. Facultad de Ingeniería. UBA. ^φ
^φ Paseo Colón 850. 1063 - Ciudad de Buenos Aires.
^θ Av. Madero 399. Anexo 3º Piso. Ciudad de Buenos Aires. Argentina.

Resumen

Se presenta una posible solución al problema de la selección del estilo de enseñanza en el marco de los Sistemas Tutores Inteligentes, donde se propone que el módulo tutor disponga de un listado de protocolos pedagógicos o métodos de enseñanza y que pueda decidir, por medio de una red neuronal entrenada, cuál de ellos será el más adecuado para cada estudiante en particular. Además, se propone un procedimiento para el agrupamiento en clusters de estudiantes con características comunes partiendo de los datos de estilos de aprendizaje y personales relevados en los mismos. A partir de estos clusters se busca la relación con los distintos protocolos pedagógicos descriptos por la Teoría Uno.

Para ello, se debe efectuar un replanteo en los componentes fundamentales del Módulo del Tutor de un Sistema Tutor Inteligente, así como también de las interfaces del mismo. Finalmente, se presentan los resultados obtenidos que validan el modo de selección planteado.

Palabras Clave: Sistemas tutores inteligentes, Modelado del tutor, Redes Neuronales, Algoritmos de Inducción.

1. Introducción

Para desarrollar Sistemas Tutores Inteligentes (STI) desde una concepción multidisciplinaria y multilingüística, se debe partir de la arquitectura de tres módulos del modelo propuesto por Carbonell [Carbonell, 1970]. Algunos investigadores presentan una arquitectura [Carbonell, 1970; Salgueiro *et al.*, 2004; Costa *et al.*, 2004] que difiere de la estructura real implementada y esta diferencia se debe a la existencia de solapamiento de funcionalidades. Se observa que muchos de los conocimientos particulares del dominio (pertenecientes al módulo de dominio) se encuentran dentro de los módulos del *tutor* y del *estudiante* tal como se observa en la Figura 1 (a y b) con las consecuentes zonas de solapamiento entre módulos.

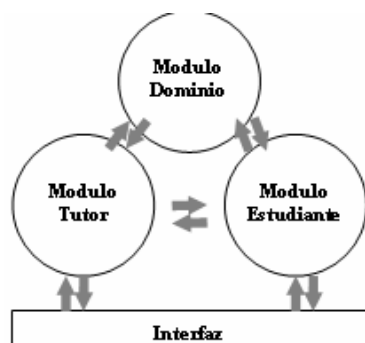


Figura 1.a

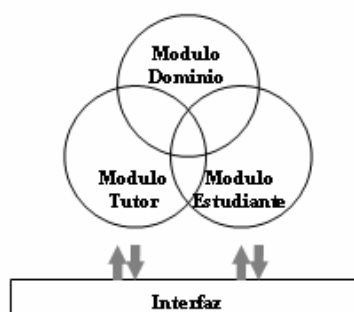


Figura 1.b

Figura 1: Diferencias entre el esquema modular teórico de un STI y su versión final implementada. **Figura 1.a:** Arquitectura de un STI teórico actual **Figura 1.b** Arquitectura de un STI real implementado.

Un análisis profundo de los solapamientos significa una definición más clara de las interfaces para diferenciar a cada uno de los módulos. Por ello, se debe identificar qué módulo será el encargado de realizar cada una de las funciones del STI a fin de quedar definido en su totalidad. De este modo se obtendrán *módulos completamente intercambiables e independientes del dominio de la aplicación*. Además de la modularidad e independencia, se busca STI centrados en las necesidades reales de los estudiantes. Esto significa contar con varios *protocolos*¹ pedagógicos que se ajusten de acuerdo a las necesidades y las preferencias de cada alumno en particular. Se trata entonces de un modelo que pueda soportar el uso de las nuevas tecnologías existentes.

2. Objetivos generales y específicos

En este proyecto el objetivo *general* es *proponer un modelo de tutor inteligente* capaz de adaptarse a las necesidades y preferencias de los alumnos, según sus estilos de aprendizaje, para proveerles de una herramienta adicional a través de la cual puedan recibir tutorizado “uno a uno” independientemente del tiempo y la distancia en que éstos se encuentren.

El objetivo general se puede desglosar en los siguientes *objetivos específicos*:

- Determinar el estado actual de las investigaciones en el tema del tutorizado humano e inteligente, relevantes para la tarea que desempeña el módulo correspondiente al tutor en un Sistema Tutor Inteligente (STI).
 - Establecer los fundamentos que sustentan las teorías seleccionadas.
 - Seleccionar los componentes del módulo del tutor, mediante el uso de submódulos, que permitan guiar a los estudiantes en el proceso de aprendizaje de manera de poder responder a sus necesidades.
- Modelar el subsistema tutor enmarcado dentro de la arquitectura de los Sistemas Tutores Inteligentes (STI), con sus submódulos e interfaces, utilizando las herramientas que provee la Ingeniería de Software y la Inteligencia Artificial (IA).
- Validar el modelo planteado con datos característicos de los alumnos, tales como los estilos de aprendizaje.

3. Desarrollo

3.1. Etapas metodológicas

En una primera etapa se tomará el módulo del tutor para su análisis. En la Figura 2a y 2b se puede ver el esquema general de los submódulos del módulo del tutor, con sus componentes básicos: el analizador de perfil y la base de protocolos pedagógicos disponibles. Éste interactúa, con el resto de los submódulos del mismo y realiza además las peticiones de los datos al módulo del estudiante, para averiguar el perfil de éste y su estado de conocimientos.

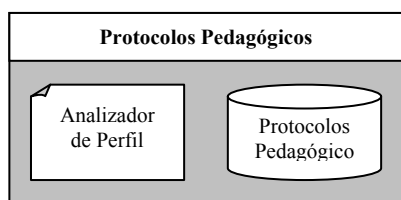


Figura 2.a: Submódulo de Protocolos Pedagógicos

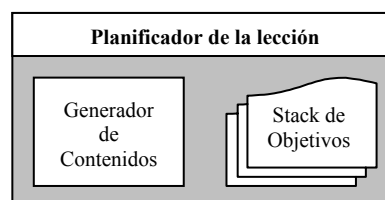


Figura 2.a: Submódulo de Protocolos Pedagógicos

El submódulo de Protocolos Pedagógicos es el encargado de seleccionar, según los datos y preferencias del estudiante, el protocolo pedagógico más adecuado para presentar los contenidos de la lección: consta de un analizador de perfil, encargado de evaluar las características del estudiante

¹ Se utiliza el término protocolo de enseñanza en referencia al método de enseñanza o método docente.

que se encuentran en el módulo del estudiante y una base de datos de los protocolos pedagógicos disponibles en el sistema.

Por otra parte, el Planificador de la lección es el encargado de diagramar los contenidos que se van a presentar en una sesión pedagógica determinada y consta con un generador de contenidos, que establece los objetivos globales de la lección y un *stack* o pila de objetivos, que mantiene el estado del objetivo de la lección y los distintos subobjetivos necesarios para cumplir con la meta global, o los que surjan a medida de que la lección avance. A continuación se resume el análisis efectuado para estos dos submódulos.

3.2. Analizador de Perfil

El Analizador de Perfil debe encontrar el protocolo pedagógico disponible en el STI que se adecue a las preferencias del estudiante para una dada sesión pedagógica. La eficacia resultante en la elección de protocolo estará influida en gran medida por la precisión en el método de selección utilizado.

Para obtener datos acerca del modo en que cada alumno aprende, se utilizarán las planillas de estilos de aprendizaje [Felder, 1998] que son instrumentos para la toma de datos. Se seleccionaron estas planillas entre las opciones disponibles, ya que la misma había sido utilizada con buenos resultados en estudios acerca de los estilos de aprendizaje de los estudiantes de Ingeniería Química e Ingeniería Informática. Es un herramienta cuya validez² y confiabilidad³ ha sido corroborada a través de su aplicación por diversos investigadores desde la fecha de su creación [Felder, 1990; Peña *et al.*, 2002; Figueroa, 2004], entre otros. La planilla utilizada consta de 44 preguntas agrupadas en 11 grupos de 4 preguntas cada uno, más datos personales del estudiante. Esto responde a los cuatro grupos de estilos de aprendizaje que según Felder [1990] se trabajan de a pares y se los puede definir según la: forma de procesar la información: tareas activas – tareas de reflexión, forma de percibir la información: sensorial – intuitiva (racional), forma de presentar la información: visual – verbal, forma del proceso del aprendizaje: secuencial – global⁴.

En la Figura 3 se observa, a modo de ejemplo una parte de la planilla descripta. Esta planilla se administró a una muestra de 121 estudiantes de una población de 800⁵, pertenecientes a La asignatura Algoritmos y programación I, de la Facultad de Ingeniería de la Universidad de Buenos Aires durante 2005. El *error de generalización* de los resultados está en el orden del 3%. Con los datos obtenidos se trata de agrupar los estilos de aprendizaje de acuerdo a las técnicas de enseñanza requeridas. *Esto permitirá correlacionar la preferencia del alumno con la disponibilidad del protocolo pedagógico más adecuado existente en el sistema.*

Se busca contrastar, por lo tanto, los resultados experimentales con la Teoría Uno de Perkins [1995] que no es un modelo, ni un método de enseñanza, sino un conjunto de recomendaciones compatibles con cualquier teoría. Ella estipula que “*La gente aprende más cuando tiene una oportunidad razonable y una motivación para hacerlo*”. Si se combinan las condiciones que estipula la Teoría Uno con cada uno de los programas de estudio, se obtienen los métodos apropiados respectivos. En otras palabras, la Teoría Uno se ajusta de distintas maneras según el programa del momento siendo los modelos fundamentales que se enumeran a continuación:

² Validez significa que la definición se ajusta al concepto y ésta debe referirse justamente a ese concepto y no a algo similar, para asegurar que se está midiendo justamente lo que pretende y no otra cosa.

³ Confiabilidad se refiere a que si se repite la medición o el registro de información, el resultado será siempre el mismo, independientemente del investigador.

⁴ Sensitivos (concretos, prácticos, orientados hacia los hechos y los procedimientos) o intuitivos (conceptuales, innovadores, orientados hacia las teorías), visuales (prefieren la presentación visual del material tal como películas, cuadros, o diagramas de flujo) o verbales (prefieren las explicaciones escritas o habladas); inductivos (prefieren la información que deviene desde lo específico hacia lo general) o deductivos (prefieren la información que deviene desde lo general hacia lo específico); activos (aprenden manipulando las cosas y trabajando con otros) o reflexivos (aprenden pensando acerca de las cosas y trabajando solos) y secuenciales (aprenden poco a poco en forma ordenada) o globales (aprenden de forma holística).

⁵ Se debe señalar que para una población de 800 estudiantes, trabajando con fórmula que presenta Hernández Sampieri (2001) para el cálculo del tamaño muestral se tiene que: V es el error estándar, V^2 es el error estándar al cuadrado, σ^2 es la desviación estándar, S^2 es la varianza de la muestra n la cual podrá determinarse en términos de la probabilidad p donde: $V=0.03$ y $V^2=(0.03)^2=0.0009$, por lo que el tamaño de la muestra sin ajustar a la población es: $n=(S^2/\sigma^2)=0.09/0.0009=100$, corrigiendo en función de la población N se tendrá el tamaño ajustado de la muestra: $n=(n/(1+n/N))=100/(1+100/800)=89$ estudiantes. El error de generalización está por debajo de 0.03.

ESTILOS DE APRENDIZAJE DE LOS ALUMNOS DE INGENIERÍA	
Carrera:.....	Universidad:.....
Año de ingreso:.....	Año que cursa:.....
Edad:.....	Sexo: <input type="checkbox"/> F <input type="checkbox"/> M
<p>1. Entiendo mejor un tema después de:</p> <input type="checkbox"/> Probarlo /ejercitarlo. <input type="checkbox"/> Pensarlo. <p>2. Prefiero ser considerado como:</p> <input type="checkbox"/> Realista. <input type="checkbox"/> Innovador. <p>3. Cuando pienso acerca de lo que hice ayer, la mayoría de las veces pienso en:</p> <input type="checkbox"/> Imágenes. <input type="checkbox"/> Palabras.	<p>11. En un libro con muchos dibujos y esquemas probablemente:</p> <input type="checkbox"/> Mire los dibujos y esquemas cuidadosamente. <input type="checkbox"/> Me focalice sobre el texto escrito. <p>12. Cuando resuelvo un problema en matemática:</p> <input type="checkbox"/> Trabajo a mi manera, para resolver un paso por vez. <input type="checkbox"/> La mayoría de las veces solo veo la solución y tengo que esforzarme para darme cuenta de los pasos para llegar a ella. <p>13. En el aula:</p>

Figura 3: Sección de la planilla de estilos de aprendizaje.

- *La instrucción didáctica* satisface una necesidad que surge en el marco de la instrucción a fin de expandir el repertorio de conocimientos del alumnado.
- *El entrenamiento* satisface la necesidad de asegurar una práctica efectiva.
- *La enseñanza socrática* se aplica para ayudar al alumno a comprender ciertos conceptos por sí mismo y darle la oportunidad de investigar y de aprender cómo hacerlo, entre otros..

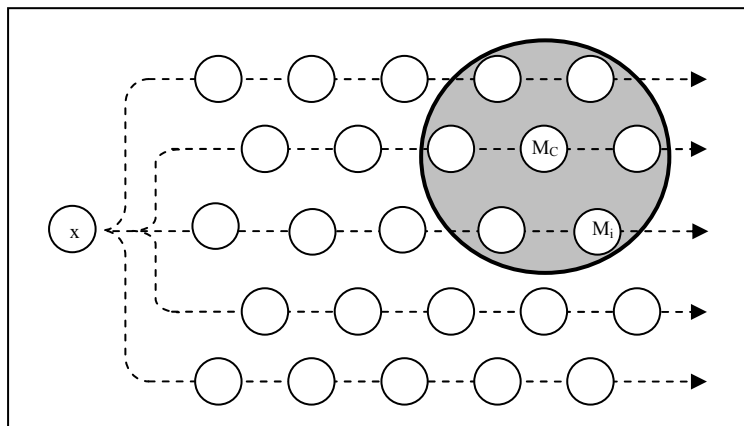


Figura 3: Conjunto autoorganizado. Cuando se ingresa un mensaje "x", este se envía a todos los conjuntos M_i del modelo. De todos estos conjuntos, solo M_c es el que mejor concuerda con "x". A su vez, todos los modelos que se encuentran en las cercanías de M_c (marcados con el círculo grande) tienen una buena concordancia con el mensaje "x" [Kohonen, 2001].

De este modo, se desea agrupar a los estudiantes en familias con características comunes, pero desconocidas *a priori*. Esto se puede llevar a cabo utilizando las redes neuronales del tipo *Self Organizing Maps* (S.O.M.) (también conocidas con el nombre de mapas de Kohonen) [Kohonen, 2001] que realizan una "clusterización" o agrupamiento según determinadas características que son comunes en el conjunto de individuos original. Las redes S.O.M. utilizan aprendizaje no supervisado y las salidas de la red no poseen un valor esperado durante la etapa de entrenamiento, por lo tanto los "clusters"⁶ resultantes de aplicar la red S.O.M. a los datos de entrada, si bien poseen características comunes, estas no se pueden ver a simple vista. Una vez finalizado el proceso de formación del mapa topológico obteniendo los clúster o familias con características comunes, se agrega a cada uno de los individuos pertenecientes a la población original se agrega el identificador del grupo generado por la red S.O.M. para continuar con el análisis.

En la Figura 3 se puede ver el esquema general de una red neuronal del tipo S.O.M., mientras que en la Figura 4 se observa en forma grafica cómo una red S.O.M. arma distintos grupos de elementos con los datos de entrada, asignando cada observación a un cluster de salida específico

Obtenidos los agrupamientos resultantes de la red S.O.M. se utilizará un algoritmo de inducción para hallar las reglas que caracterizan a cada uno de estos grupos. Para ello, se utilizarán algoritmos que pertenecen a la familia *Top-Down Induction Tress* (TDIT) (o en castellano *Árboles Inductivos de Arriba Hacia Abajo*).

⁶ Un cluster es una familia de datos o atributos que poseen características en común. Estas características pueden ser explícitas o implícitas a los usuarios de los datos de la red.

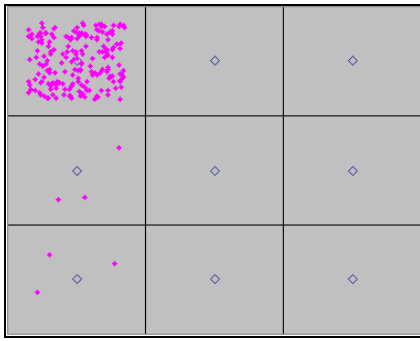


Figura 4.a

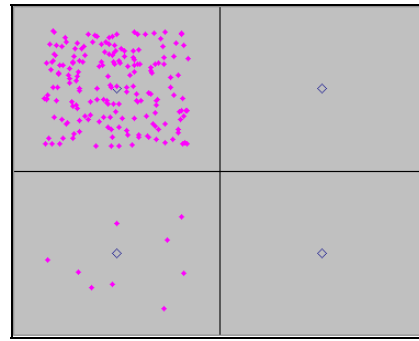


Figura 4.b

Figura 4: Resultado de la clusterización con distinta cantidad de neuronas. **4.a:** Muestra de 9 clusters con (3 neuronas). **4.b:** Muestra de 4 clusters (2 neuronas).

Si bien existen varios algoritmos que realizan estas funciones, uno de los mas completos es el C4.5 de Quinlan [Quinlan, 1993], que es una extensión del algoritmo ID3 (Induction Decision Trees) propuesto también por Quinlan [Quinlan, 1987]. Su objetivo es generar un árbol de decisión y luego las reglas de inferencia que caracterizan a dicho árbol. El C4.5 tomará como entradas los datos de los estudiantes ya categorizados y se obtendrán como salida reglas (ver las reglas en el apartado de resultados) del tipo:

Si Precondición₁, Precondición₂, ..., Precondición_n **Entonces** Cluster_k.

Se utiliza el C4.5 en lugar del ID3 ya que el primero permite trabajar con valores continuos para los atributos (como la edad de los estudiantes), separando los resultados posibles en dos ramas: una para aquellos $A_i \leq \theta$ y otra para $A_i > \theta$, (donde A_i es un atributo y θ es un valor limite para el cual todos los elementos de la regla menores a θ se abren en un subárbol de reglas y los mayores en otro). El algoritmo genera un árbol de decisión a partir de los datos mediante particiones realizadas recursivamente. El árbol se construye mediante la estrategia de *depth-first* (profundidad primero) considerando todas las divisiones posibles del conjunto de datos. Se selecciona, luego, la división del conjunto de datos con ls mayor ganancia de información. En el caso particular de atributos discretos, se considerará una división por cada uno de sus "n" resultados.

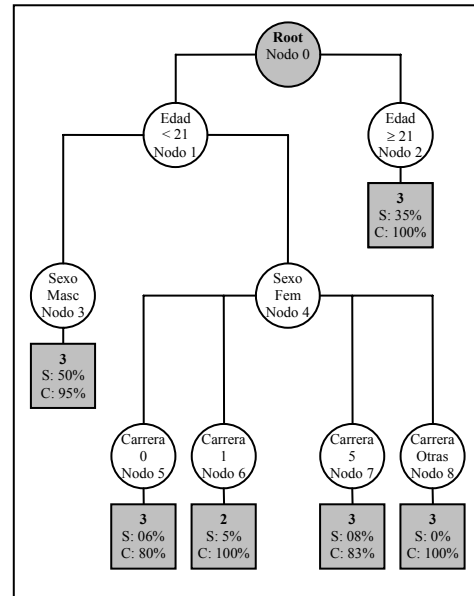


Figura 5: Árbol de decisión generado por el algoritmo C4.5 a partir de la clasificación obtenida mediante la aplicación de redes S.O.M., donde S representa el soporte y C es la confianza de la regla.

C4.5 construye reglas que abarcan todos los datos, cubriendo todos los ejemplos positivos y ninguno de los negativos. En caso de existir un ejemplo negativo a una de las reglas, se continúa avanzando en profundidad en el árbol de atributos. Además, el algoritmo puede aceptar entradas incompletas o inconsistentes y las soluciona mediante pre o post procesamiento, donde también se reduce la

cantidad de reglas generadas mediante *pruning*⁷ (truncamiento o poda), el cual no afecta la precisión de las reglas obtenidas.

Una vez finalizado el proceso de inducción por el algoritmo C4.5, el resultado será un conjunto de reglas que describen a cada uno de los clusters. De esta manera se pueden identificar las características de cada una de las familias de elementos encontradas. En la Figura 5 se puede ver como modo de ejemplo, un resultado gráfico posible como salida del algoritmo de inducción C4.5. Para clasificar a un individuo dentro de los grupos establecidos se debe partir del nodo denominado “*root*” y se debe evaluar cada uno de los atributos hasta llegar a un nodo terminal que contiene la clasificación para dicha observación.

Una vez obtenida la menor cantidad posible de reglas por medio de *pruning* (poda) para evitar el *overfitting*⁸ (sobre ajuste), se está en condiciones de pasar a otra etapa del análisis en la que, por medio de un proceso de inferencia, se relacionan las reglas descriptivas de los distintos clusters con los protocolos pedagógicos disponibles. Para llevar a cabo la inferencia se cuenta con datos adicionales del desempeño de los estudiantes muestreados en sendos cursos de Algoritmos y programación I con diferentes protocolos de enseñanza. socrático y magistral.

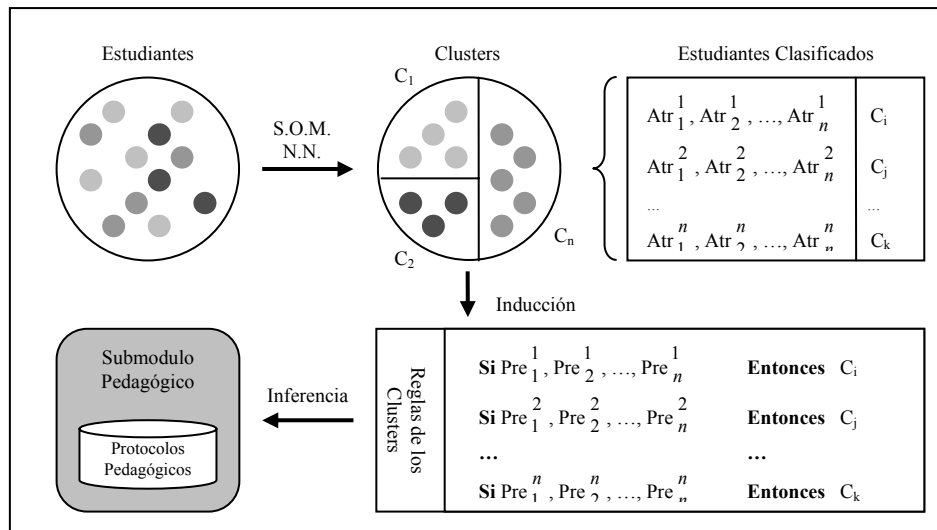


Figura 6: Esquema básico para la inferencia de los distintos patrones para categorizar los estilos pedagógicos de los alumnos

En la Figura 6 se puede ver el esquema que representa al proceso de selección en forma global que comienza con el universo de estudiantes relevados, a estos se les aplica una red S.O.M para identificar los grupos y una vez clasificados los estudiantes en grupos, se procede a aplicar el algoritmo C4.5 para encontrar las reglas que describen a cada uno de los clusters. Luego, se relaciona cada cluster con un protocolo pedagógico real utilizado por un tutor humano, que dicta tutorías a los distintos estudiantes relevados y se realiza la triangulación metodológica.

Las reglas resultantes del proceso global se han seleccionado en función de los datos relevantes de las respuestas a la planilla de estilos de aprendizaje que llenaron los alumnos.

El análisis muestra que se obtienen resultados similares utilizando solo un subconjunto de los atributos de las planillas en lugar de utilizar todos los atributos. Una vez encontrado el protocolo pedagógico de preferencia del alumno para la sesión tutelar para un estudiante usuario determinado,

⁷ En el caso particular del algoritmo C4.5, el mecanismo de *pruning* funciona de la siguiente manera: una vez creado el árbol de decisión sobre un conjunto de instancias de entrenamiento, se realizan ajustes al árbol y se evalúa esta versión modificada contra un conjunto de instancias de validación. El algoritmo solo mantiene los cambios si el árbol modificado o *podado* (pruned) se comporta de manera superior al original. El uso de un conjunto de instancias de validación ayuda a eliminar el *overfitting* en la información por medio de la evaluación de las inconsistencias y las coincidencias de este conjunto contra el conjunto de instancias de entrenamiento. El error se reduce podando los nodos internos del árbol, reetiquetándolos con la nueva clasificación y descartando todos sus nodos descendientes. Esto obviamente limita el *overfitting*, pero es un proceso de costo computacional elevado.

⁸ En el caso en que los datos relevados sean demasiado homogéneos (una varianza muy pequeña) el algoritmo de inducción podría encontrar un modelo demasiado optimista, que satisface demasiado la tendencia de los datos tomados, pero que a la vez es pobre para hacer predicciones para los datos que tiene una varianza grande, los cuales pueden ser los más representativos del dominio.

se debe analizar el caso en que el sistema no cuente en el módulo de dominio el estilo o protocolo de enseñanza seleccionado utilizando ese protocolo en particular. Este punto se analizará a continuación.

3.3. Modificaciones posteriores a la selección del protocolo pedagógico

Una vez entrenada la red neuronal y generado el árbol, sólo resta ingresar los datos del estudiante y utilizar la red entrenada S.O.M. decodificando luego el clúster de salida por el correspondiente protocolo pedagógico o utilizar los datos almacenados en el módulo del estudiante para ver cuáles de las reglas generadas por el algoritmo de inducción se cumplen y encontrar así el protocolo pedagógico más adecuado. En la Figura 7 se puede ver una representación gráfica de estas dos maneras de selección del protocolo pedagógico.

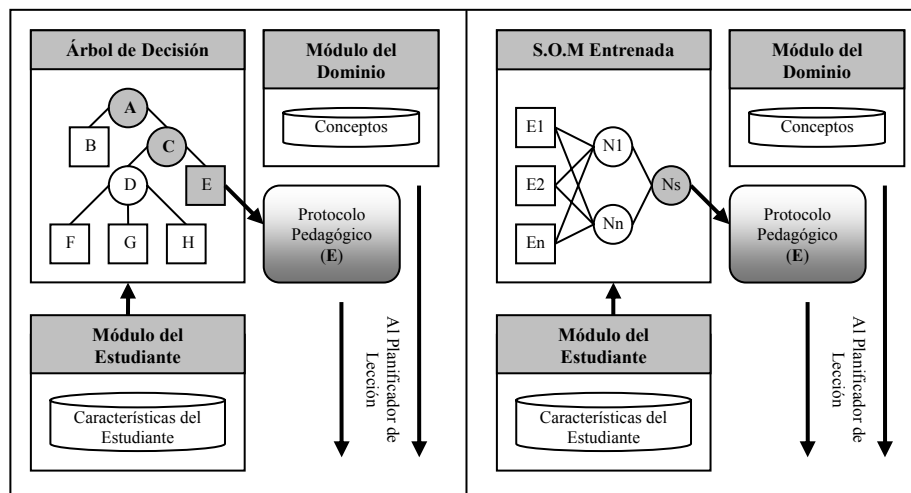


Figura 7.a

Figura 7.b

Figura 7: Se observa la solución en sus dos versiones, una utilizando directamente las reglas resultantes, modificando las salidas para que concuerden con los protocolos pedagógicos (7.a) y otra donde directamente se utiliza la red neuronal artificial S.O.M. ya entrenada para establecer el protocolo pedagógico a utilizar (7.b).

En este esquema de implementación (en sus dos variantes), tanto los protocolos pedagógicos predichos por la red S.O.M., como por las reglas del algoritmo de inducción C4.5 dan una única salida, es decir, un único protocolo pedagógico resultante y no proporcionan información acerca de cual sería una segunda opción adecuada. Es necesario evaluar las opciones alternativas de protocolos pedagógicos ya que el módulo del dominio puede no contener almacenados datos suficientes para completar satisfactoriamente todos los contenidos de sesión pedagógica en todos los protocolos que se encuentran en el sistema.

La idea inicial se puede ampliar para que no solo el submódulo analizador de perfil entregue el mejor protocolo pedagógico, sino que entregue una escala o *ranking* de protocolos en orden descendente con respecto a su afinidad con el estudiante. Luego, lo único que se requiere es recorrer todos los protocolos pedagógicos incluidos en el sistema, en el orden que indica este *ranking* y verificar que todo el material necesario para llevar a cabo la lección utilizando ese protocolo en particular esté disponible en el módulo del dominio.

Analizadas las herramientas disponibles proporcionadas por la Inteligencia Artificial, la opción más viable para la construcción del *ranking*, consiste en la implementación de otra red neuronal, pero esta vez utilizando aprendizaje del tipo supervisado. En este nuevo modelo se conocen todas las entradas (los atributos correspondientes al árbol de decisión resultante del algoritmo de inducción C4.5) y se conocen las salidas (los clusters de las redes S.O.M., reemplazando el número de cluster por el protocolo pedagógico asociado). Se pueden utilizar para solucionar el problema planteado redes neuronales artificiales del tipo Backpropagation [Rumelhart *et al.*, 1986], evaluando la aptitud de

cada uno de los protocolos pedagógicos con respecto a las preferencias del estudiante que se está analizando.

No se podría haber utilizado una red Backpropagation desde el principio, es decir, directamente reemplazando la red S.O.M. de Kohonen, ya que se requería en primer lugar una división de datos en familias y eso sólo se podía lograr con una red con aprendizaje no supervisado.

Una vez finalizado el proceso de la Figura 6 recién se estará en condiciones de entrenar una red para que produzca los resultados deseados, utilizando ahora aprendizaje del tipo supervisado ya que se conocen las salidas a obtener.

En la figura 8 se puede ver el esquema similar al de la figura 7, pero en lugar de utilizar un árbol o la red S.O.M. entrenada, se utiliza la red Backpropagation.

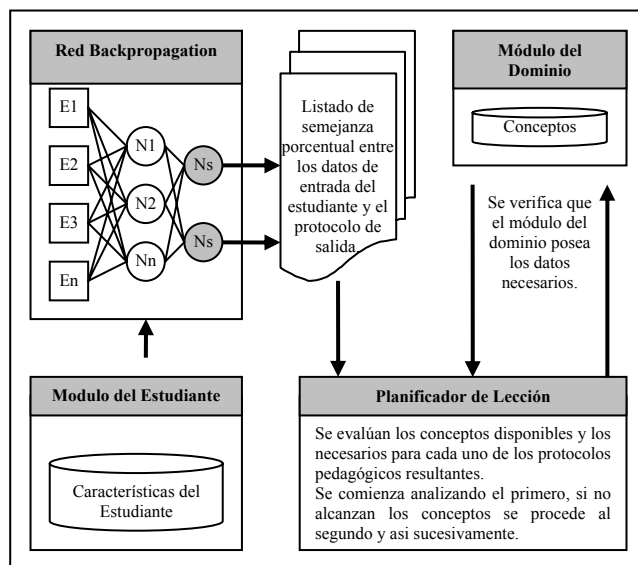


Figura 8: Estructura modificada de la solución, una utilizando la red neuronal artificial del tipo Backpropagation.

3.4. Submódulo de Generador de la Lección.

Una vez seleccionado el protocolo pedagógico por el submódulo Analizador de Perfil, se deben generar los contenidos de la sesión a ser presentados al alumno del sistema. La tarea del Generador de la Lección es la de encargarse de la interacción entre el sistema y el alumno.

Luego de la selección del protocolo pedagógico, se debe encontrar el concepto a explicar en la lección. Esto se realiza comparando el mapa de conocimientos que se obtiene del módulo del dominio con los conocimientos que el sistema supone que el estudiante posee que se obtiene del módulo del estudiante.

Se deben "superponer" los conocimientos del estudiante con el mapa de conocimiento que se obtiene del módulo de dominio para encontrar los conocimientos faltantes, y de acuerdo al tiempo disponible o una función que jerarquice la importancia parcial de cada uno de los temas que se agregaran al *stack* de objetivos de la lección. Esto se puede ver en la Figura 10.

El Generador de contenidos debe encargarse ahora de presentar los contenidos en el orden correcto y procesar las respuestas de los estudiantes, agregando los temas que se detectan faltantes en el mapa de conocimientos del alumno en el *stack* de objetivos de la lección e informando sobre esta discrepancia entre la realidad y lo supuesto por el sistema el módulo del estudiante.

Cuando el sistema detecta que el estudiante ha adquirido un nuevo concepto, o que se ha corregido un concepto erróneo, es el generador de contenidos el encargado de actualizar el estado del *stack* de objetivos y si este se encontrara vacío, dará por finalizada la sesión tutelar.

Este módulo es el responsable de controlar la interacción entre el tutor y el estudiante, por lo tanto debe decidir cómo responder ante el estudiante dado un problema en particular [Woolf, 1984; Winkels *et al.*, 1988]. La lección debe ser planeada especialmente para obtener los resultados esperados del alumno y a su vez debe ser lo suficientemente flexible y coherente como para respetar la interacción. Por lo tanto se pueden redefinir las funciones básicas del modulo generador de contenidos tales como: decidir la forma de presentar la información al estudiante, el orden en que esta información se presentará y cómo se responderá a las interacciones del estudiante. Los objetivos principales de este modulo se pueden plantear según Woo Woo [Woo Woo, 1991] como: decidir cuando presentar los contenidos de la lección, corregir los conceptos erróneos del estudiante para un problema dado y generar el feedback en todas las respuestas del usuario hacia el sistema,

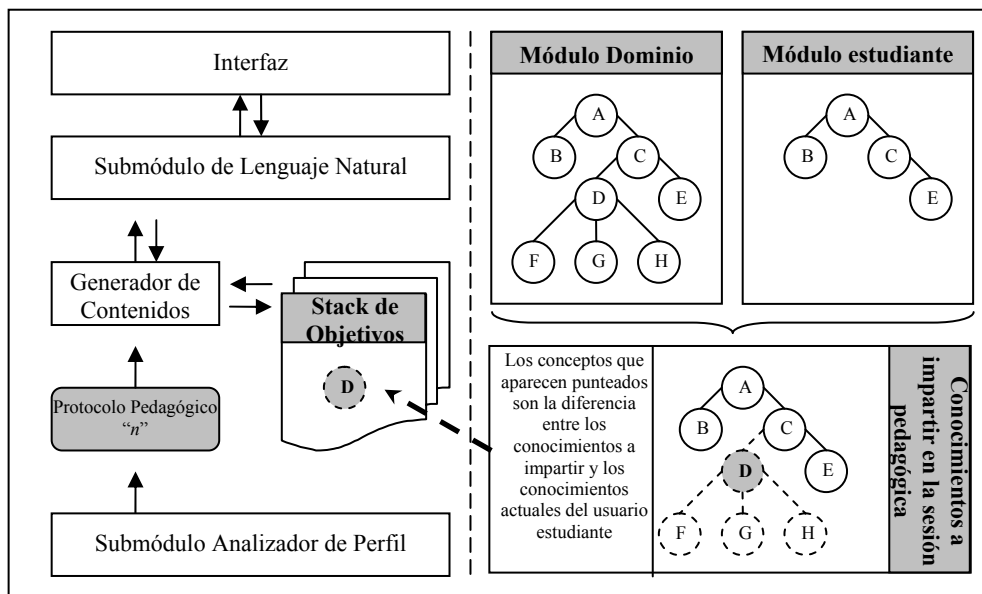


Figura 9: Esquema general de los pasos a seguir luego de la obtención del protocolo pedagógico por el submódulo ar de perfil y de la creación de los contenidos de la lección.

Si el estudiante no logra obtener la respuesta correcta, el tutor debe proveer los medios para que el estudiante se encauce nuevamente en el camino correcto para solucionar el problema. A su vez, el sistema debe responder las preguntas del usuario con la respuesta más apropiada de acuerdo a todas las opciones.

3.5. Determinación de Objetivos de la lección.

La determinación de los objetivos de la lección es una tarea esencial en los STI, ya que debe generar objetivos globales, coherentes y consistentes para que éstos sean impartidos a cada estudiante en particular. [Woo Woo, 1991]. Dado que cada uno de los conceptos almacenados en el módulo del dominio para cada asignatura están relacionados entre sí, la secuencia en la que estos se presenten afectará de manera decisiva el rendimiento del sistema en su conjunto y las necesidades del estudiante. La determinación de los objetivos dejará planteada la estrategia del sistema para una sesión pedagógica en particular.

Por lo tanto, la determinación de los objetivos de la lección no solo requiere el uso de conocimientos almacenados en el módulo del estudiante, para determinar lo que éste conoce y lo que no, sino que también requiere de metainformación sobre los conceptos a impartir. El metaconocimiento *es conocimiento acerca del conocimiento* y se utiliza en el caso del módulo del dominio para determinar las dependencias *jerárquicas* entre los distintos conceptos, el tiempo que puede tardarse en impartirlos y la importancia de los mismos (si es un concepto aislado, si son sub. temas o si son temas principales o capítulos). Se puede armar entonces el mapa de conocimientos del alumno y el mapa de conocimientos faltantes necesarios para finalizar con la curricula de la materia.

Por otra parte no es el objetivo de una lección pedagógica obtener las dependencias del módulo del dominio e impartirlas todas en una única "mega" sesión pedagógica, sino que la idea de la flexibilidad de un STI también reside en que éste sea capaz de evaluar los conocimientos adquiridos por el estudiante y actuar conforme a ellos. Es por esto que se requiere contrastar los datos del módulo del dominio con el mapa de conocimientos instantáneo del estudiante para encontrar los conceptos posibles y disponibles para impartirle al estudiante. Un esquema del procedimiento a seguir puede verse en la Figura 10.

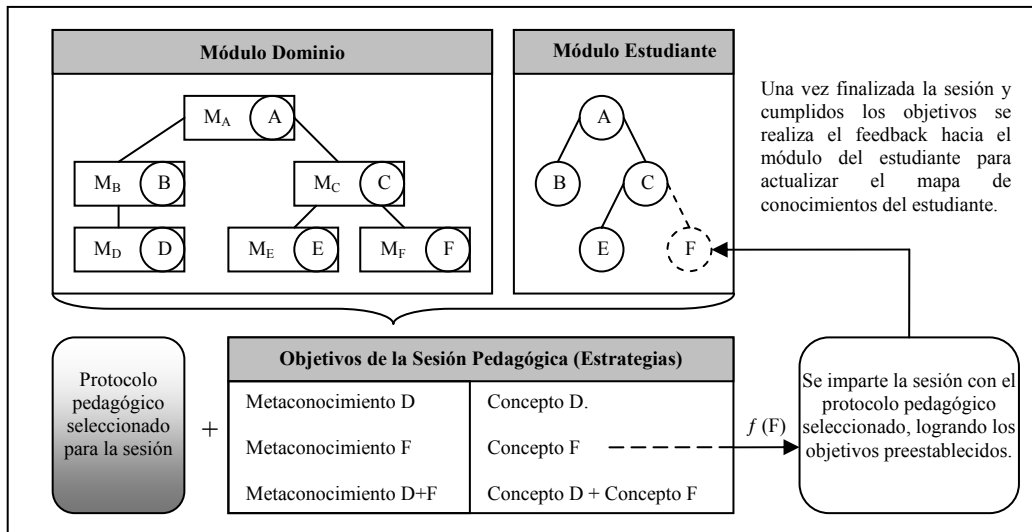


Figura 10: Esquema general de selección de objetivos para la sesión pedagógica con retroalimentación luego de finalización de dicha sesión.

Una vez que se determinan todas las combinaciones válidas entre los conocimientos faltantes en el mapa de conocimientos del estudiante y los conceptos almacenados en el módulo de dominio debe existir una función para determinar cuál de todas las combinaciones será la seleccionada para la sesión en particular. Se debe tener en cuenta que el número de opciones puede ser relativamente grande y éstas requieren un análisis mucho más complejo.

Existen distintas opciones para implementar la función que determinará cuál de todas las opciones será la seleccionada para una determinada sesión pedagógica. Por ejemplo, en el tutor *CircSim* [Kim, 1989; Kim, 2000; Cho, 2000; Hume, 1995; Shah, 1997; Hume *et al.*, 1992] se selecciona el objetivo de la sesión por medio de un conjunto de reglas que toman como base los conocimientos del estudiante [Woo Woo, 1991]. Estas reglas son tomadas de las sesiones de educación de conocimiento con tutores humanos, pero fallan nuevamente en la separación de los módulos, ya que los conocimientos resultantes a enseñar deben provenir del módulo de dominio y no deben estar escritos en forma “*Hard-Coded*”⁹ dentro del módulo del estudiante. Otra opción es, por ejemplo, la que se utiliza en el tutor *Meno* [Woolf, 1984] donde avanza linealmente a través de cada uno de los temas de la curricula.

```

define("Result", "0");
Knowledge = get_student_knowledge(Student_X)
if (Knowledge != Full) {
    K_list[]=get_domain_dependant_knowledge (Knowledge);
    foreach (K_list as lesson_objective) {
        K_list_eval[]= F (lesson_objective);
        Sort (K_list_eval[], Result);
    }
    Output(K_list_eval[0])
}

```

Figura 11. Uso de la función F para determinar el objetivo de la sesión pedagógica. Primero se encuentra la diferencia entre el conocimiento a aprender y el aprendido por el estudiante, luego las combinaciones posibles de conocimiento, seleccionando a través de F el más apto..

Independientemente el método que se utilice para la creación de la función para determinar el objetivo de la lección, los parámetros que se deben evaluar están relacionados con los metaconocimientos que almacenan los conceptos en el módulo del dominio. Los criterios básicos puede basarse en *duración (tiempo que insume la lección: que no sea ni muy extensa ni muy corta)*,

⁹ Los datos están FIJOS en el código fuente y no pueden alterarse salvo recompilando todo

profundidad (cantidad de temas y subtemas a barrer en una lección particular), relación con la curricula (mantiene una relación entre lo dictado por el docente y por el sistema).

Independientemente de las particularices de la función para determinar el objetivo de la lección, se puede plantear un pseudocódigo que explique el funcionamiento y determinación de estos objetivos, tal como se muestra en la Figura 11.

4. Resultados

Se procesaron las planillas de 121 estudiantes por medio de la citada red S.O.M. con parámetros de configuración que se muestran en la Tabla 1.

Una vez clasificados los datos se procede a analizar los 3 clusters resultantes y a eliminar uno ya que correspondía a una planilla mal completada (excepción).

Luego, se aplica el algoritmo C4.5, se lo analiza para evitar *overfitting* y se realiza la poda con lo que se obtienen las reglas que describen los clusters que se muestran en la Tabla 2.

Para finalizar, se realiza el cruce de los datos para relacionar los clusters resultantes con

los protocolos utilizados en las clases de los tutores humanos, resultando que el cluster 1 esta relacionado con el protocolo *magistral* y el cluster 2 con el protocolo *socrático*.

Número de Observaciones	121
Número de Variables	47
La cantidad de Neuronas Artificiales ¹⁰	10
La cantidad de Ciclos	1000
Aleatoriedad	Si
Parámetro de Aprendizaje	
Inicial	0.9
Final	0.1
Función de decaimiento	Exp
Parámetro para el vecindario Gaussiano	
Inicial	
Final	99,0%
Función de decaimiento	01,0%
	Exp

Tabla 1: Parámetros utilizados para el entrenamiento de las Redes Neuronales S.O.M.

Reglas	Antecedente	Consecuente
Reglas 0	Clust. ID = Magistral	Clust. ID = Magistral
Reglas 1	IF Edad ≥ 21	THEN Clust. ID = Socratico
Reglas 2	IF Sexo = Masc	THEN Clust. ID = Magistral
Reglas 3	IF Carrera = 5	THEN Clust. ID = Socrático
Reglas 4	ELSE	Clust. ID = Magistral

Tabla 2: Reglas resultantes de recorrer el árbol generado por el Algoritmo C4.5 y reemplazar los clusters obtenidos por S.O.M. por los correspondientes protocolos pedagógicos.

El atributo discriminante más importante resulto ser la edad del estudiante. Para los mayores de 21 años se encuentra que prefieren un protocolo del tipo *socrático*, mientras que para los menores se requieren evaluar más parámetros. Esto puede evidenciar el estado de formación mental a lo largo de los años dentro de una carrera de las denominadas “*duras*” como pueden serlo las ingenierías.

5. Grado de avance

Se ha completando la primera etapa del proyecto centrada en el modelado de las preferencias y necesidades de los estudiantes, para obtener el método de tutorizado según su estilo de aprendizaje.

Los resultados de la selección de los protocolos se validaron utilizando muestras longitudinales de estudiantes, a partir de 2004 y continuando en 2005 (que son las que se presentan). Estos resultados permiten replicar el trabajo durante tres cuatrimestres. De este modo se obtiene un método eficiente para adecuar el sistema al tipo de tutorizado con el que el alumno siente más afinidad.

Esto permite obtener un modelo adaptativo de STI (módulo del tutor) que reacciona ante las necesidades particulares de los alumnos y que da cuenta de las problemáticas surgidas en los alumnos ingresantes al

¹⁰ Si bien el valor hallado es muy elevado con respecto a la cantidad de clusters que se forman, esto solo agrega un retardo temporal en el entrenamiento de la red y ninguna otra complicación. Los valores para el caso presentado obtienen un resultado similar si en lugar de 10 neuronas artificiales (100 clusters como máximo) se utilizan 3 (9 clusters como máximo).

proveerles de una herramienta que si bien no reemplaza al tutor puede para paliarlas ayudando al estudiante.

6. Conclusiones

El resultado presentado es un módulo de tutor inteligente que se ajuste a los estilos de los estudiantes que brinde flexibilidad a un sistema de módulos componentes. La red SOM agrupa a los estudiantes de acuerdo a sus preferencias y les ofrece un protocolo pedagógico (estilo de enseñanza) que se ajusta a sus preferencias o estilo de aprendizaje. Los datos obtenidos concuerdan en un 80% con la selección real de los alumnos.

Entre las *líneas de investigación futuras* queda por definir la incidencia del modo de tutorizado de acuerdo a la teoría de las inteligencias múltiples de Gardner [1995] en el módulo del tutor, a fin de obtener un acercamiento a través de la postura de las ocho inteligencias. Esto podría flexibilizar aun más el estilo de tutorizado lográndose un mejor ajuste a las necesidades del estudiante. Por otra parte, paralelamente se están diseñando los módulos del estudiante centrados en la aproximaciones de la psicología cognitiva y de la educación, y el modulo evaluador conteniendo el historial del estudiante.

5. Referencias

- Carbonell, J. R. (1970). *AI in CAI: An artificial intelligence approach to computer assisted instruction*. IEEE transaction on Man Machine System. Volumen 11 número 4, p. 190-202.
- Cataldi, Z.; Salgueiro, F.; Lage, F. J. y García-Martínez, R. (2005). *Sistemas tutores inteligentes: los estilos del estudiante para selección del tutorizado*. Proceedings of WICC 2005: Workshop de Investigadores en Ciencias de la Computación en CD. 13 y 14 de mayo. Universidad Nacional de Río Cuarto. Córdoba. RED UNCI.
- Costa, G.; Salgueiro, F. A., Cataldi, Z., García Martínez, R. y Lage, F. J. 2005. *Sistemas inteligentes para el modelado del estudiante* Proc. GCETE'2005, Global Congress on Engineering and Technology Education CD. marzo 13-15.
- Felder R.M.; Silverman L.K. (1988). *Learning Styles and Teaching Styles in Engineering Education*. Engr. Education, Volumen 78, número 7, p. 674-681.
- Felder, R. M.; Baker Ward, L. (1990). *How engineering students learn, how engineering professor teach, and what goes wrong in the process*. Proceedings of FIE'90: Frontiers in Education, p. 82-84.
- Figuroa, N.; Lage, F.; Cataldi, Z.; Denazis, J. (2003). *Evaluación de las experiencias para mejoramiento del proceso de aprendizaje en asignatura inicial de la carrera ingeniería informática*. Proceedings of International Conference on Engineering and Computer Education, CD. ICECE 2003, 16 al 19 de Marzo. San Paulo. Brasil
- Gardner, H. (1985) *Las Inteligencias Múltiples*. Paidós, Barcelona.
- Hume G., Michael, J; Rovick, A.; Evens, M. (1996), *Hinting as a tactic in one-on-one tutoring*. Journal of Learning Sciences
- Kim, J. H. (1989). *CIRCSIM-Tutor: An Intelligent Tutoring System for Circulatory Physiology*. Ph.D. tesis, Illinois Institute of Technology
- Kohonen, T. (2001). *Self-Organizing Maps, third edition. Springer Series in Informarion Sciences*. Ed. Springer. Helsinki University of Technology Neural Networks. Espoo, Finlandia. Pitman, London.
- Peña, C. I.; Marzo, J. L.; De la Rosa, J. L.; Fabregat, R. (2002). *Un sistema de tutoría inteligente adaptativo considerando estilos de aprendizaje*. Actas del IV Congreso Iberoamericano de Informática Educativa, IE2002, Vigo (España), Noviembre 20-22, ISBN 848158-227-1.
- Quinlan, J. R. (1987). *Simplifying Decision Trees*. Simplifying Decision Trees. International Journal of Man-Machine Studies 27(3): 221-234.
- Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann.
- Rumelhart, D. E.; Hinton, G. E.; Williams, R. J. (1986). Learning internal representations by back-propagating errors in Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Eds. Cambridge, MA: MIT Press, vol. 1, p. 318-362.
- Salgueiro, F. A, Costa, G., Cataldi, Z., García Martínez, R. y Lage, F. J. 2005. *Sistemas inteligentes para el modelado del tutor*. Proc. GCETE'2005, Global Congress on Engineering and Technology Education. CD. marzo 13-15
- Winkels, R., Breuker, J. and Sandberg, J. 1988. Didactic Discourse in Intelligent Help Systems. Proceedings of Intelligent Tutoring Systems: ITS-88, Montreal, Quebec, Canada, 279-285.
- Wolf, B. (1984). *Context Dependent Planning in a Machine Tutor*. Ph.D. Dissertation, University of Massachusetts, Amherst, Massachusetts
- Woo Woo, C. (1991). *Instructional planning in an Intelligent Tutoring System: Combining global lesson plan with local discourse control*. Degree of Doctor of Philosophy in Computer Science in the Graduate School of the Illinois Institute of Technology. Chicago, Illinois. December.