

El problema del modelado del estudiante en Sistemas Tutores Inteligentes

Zulma Cataldi y Fernando Lage

LIEMA. Laboratorio de Informática Educativa y Medios Audiovisuales. Facultad de Ingeniería.
Universidad de Buenos Aires. Paseo Colón 850 4º Piso. C1063ACV. Ciudad Autónoma de Bs. As. Argentina.
liema@fi.uba.ar, flage@fi.uba.ar

Resumen

El presente subproyecto forma parte del proyecto “Metodología de diseño y evaluación de Sistemas tutores inteligentes” que se desarrolla actualmente en Facultad de Ingeniería, donde para la producción de los STI convergen las distintas corrientes y vertientes epistemológicas que quedan evidenciadas en la práctica docente. Se busca una nueva opción que sea más versátil respecto de cómo el alumno mantiene, organiza y adquiere los nuevos conocimientos. Las diferentes formas de pensar de los alumnos, constituyen los estilos de aprendizaje, a través de los cuales se puede llegar a analizar de qué modo en que estos conocimientos se almacenan, se relacionan y se utilizan con los adquiridos anteriormente. Por ello, es necesario caracterizar a los estudiantes, ya que esto permitiría soluciones más individualizadas de acuerdo a cada estilo. Esto conlleva a rediseñar los componentes de cada módulo del sistema tutor. Así, un sistema tutor ó bien un asesor inteligente con diagnóstico del estilo del estudiante puede ser una opción válida.

Palabras clave: estilos de aprendizaje, modelo del estudiante, sistemas tutores inteligentes

1. Introducción.

Los sistemas tutores inteligentes (STI) comenzaron a desarrollarse en los años 80 y fueron diseñados con la idea de impartir conocimiento con base en alguna forma de inteligencia para guiar al estudiante en el proceso de aprendizaje (Urretavizcaya, 2001; Sancho, 2002). Su propósito es exhibir un comportamiento similar al de un tutor humano, que se adapte al comportamiento del

estudiante, identificando la forma en que el mismo resuelve un problema y brindarle ayuda cuando cometa errores. Un tutor inteligente, por lo tanto: “es un sistema de software que utiliza técnicas de inteligencia artificial (IA) para representar el conocimiento e interactúa con los estudiantes para enseñárselo” (VanLehn, 1988). Wolf (1984) define los STI como: “sistemas que modelan la enseñanza, el aprendizaje, la comunicación y el dominio del conocimiento del especialista y el entendimiento del estudiante sobre ese dominio”. Giraffa (1997) los delimita como: “un sistema que incorpora técnicas de IA (Inteligencia Artificial) a fin de crear un ambiente que considere los diversos estilos cognitivos de los alumnos que utilizan el programa”.

Entre los STI desarrollados siguiendo las ideas de Carbonell (1970) y con base en paradigmas de programación convencional (no a través del paradigma de agentes inteligentes) se pueden destacar: *Scholar* (Carbonell, 1970), *Why* (Stevens et al., 1977), *Sophie* (Brown et al., 1989), *Guidon* (Clancey et al., 1991), *West* (Burton et al., 1981), *Buggy* (Brown y Burton, 1978), *Debuggy* (Brown et al., 1989) *Steamer* (Stevens et al., 1977), *Meno* (Wolf, 1984), *Proust* (Johnson et al., 1984), *Sierra* (VanLehn, 1988).

Luego, surgen: *Andes* (Gertner et al., 1998; Gertner y Van Lehn, et al., 2000) en el *Pittsburgh Science of Learning Center's LearnLab*, que su consorcio con miembros de *Carnegie Mellon University*, *University of Pittsburgh* y *Carnegie Learning*. *Metutor* es un tutor de medios-fines del *Department of Computer Science, U.S. Naval Postgraduate School, Monterey*. (Galvin, 1994; Rowe, 1998). *ITSpoke* es un Proyecto que usa un sistema de diálogos basado en textos y medios

finés (Litman y Silliman, 2005). Se desarrolla en la *University of Pittsburg, Department of Computer Science & Learning Research and Development Center Pittsburg*.

El STI *CircSim*, fue desarrollado en conjunto por el *Departamento de Ciencias de la Computación del Illinois Institute of Technology* y el *Departamento de Fisiología del Rush College of Medicine*. Este tutor es el más avanzado actualmente en su tipo, y se lo utiliza en el *Rush College of Medicine* para complementar las clases teóricas sobre problemas cardiovasculares (Kim, 1989; Kim, 2000; Cho, 2000; Hume *et al.* 1992, 1996; Shah, 1997, Evens *et al.*, 2001). *AGT (Advanced Geometry Tutor)*, es un proyecto que tiene como objetivo, construir un STI para el uso en clases de geometría avanzada (Matsuda y Van Lehn, 2005) en la *University of Pittsburg* a través de la *National Science Foundation through* y el *Center for Interdisciplinary Research on Constructive Learning Environments* de la *University of Pittsburg* y *Carnegie Mellon University*. *AutoTutor*: es un STI basado en la web por un grupo interdisciplinario de la *Office of Naval Research and the National Science Foundation*, (DiPaolo *et al.*, 2002; Graesser *et al.*, 2005a,b, 2006; Chipman *et al.*, 2005).

El *Computer Tutoring Group (ICTG)*, que trabaja en el *University's Computer Science and Software Engineering Department*, en la *University of Canterbury* ha desarrollado una serie de STI: *Aspire* (Mitrovic *et al.*; 2006), *Sql-Tutor* es un sistema de enseñanza basado en el conocimiento que enseña *Sql* a los estudiantes (Mitrovic, Martin y Mayo, 2002; Mitrovic, 2003), *Kermit*; *EER-Tutor*, *ERM-Tutor*: (Milik, Marshall, y Mitrovic, 2006) y *Normit*.

2. Fundamentación.

Los STI permiten la emulación de un tutor humano para determinar *qué enseñar, cómo enseñar y a quién enseñar* a través de un *módulo del dominio*: que define el dominio del conocimiento, un *módulo del estudiante*: que es capaz de definir el conocimiento del

estudiante en cada punto durante la sesión de trabajo, un *módulo del tutor*: que genera las interacciones de aprendizaje basadas en las discrepancias entre el especialista y el estudiante y finalmente *la interface*¹ con el usuario: que permite la interacción del estudiante con un STI de una manera eficiente (conocimiento sobre *cómo presentar* los contenidos). A través de la interacción entre los módulos básicos, los STI son capaces de juzgar lo que sabe el estudiante y cómo va en su progreso, por lo que la enseñanza, puede ser ajustada según las necesidades del estudiante, sin la presencia de un tutor humano.

El problema que da marco a la investigación se centra entonces en que los STI, en general no proveen de un modo de aprendizaje adaptable (Waern, 2001) de acuerdo a los conocimientos previos y a la capacidad de evolución de cada estudiante (Millán *et al.*, 2000) y a las concepciones epistemológicas que subyacen en las prácticas de enseñanza. Por otra parte, cada estudiante debería poder elegir las características del procedimiento aplicado por el tutor de acuerdo a sus preferencias (Khuwaja, 1994), entre los diferentes métodos que éste utilice: instruccional, orientador, socrático u otros (Shim, 1991; Perkins, 1995; Casas, 1999), y si lo deseara debería poder cambiarlo de acuerdo a sus propios requerimientos.

En el contexto de los sistemas inteligentes se encuentran las redes neuronales (RN), que son interconexiones masivas en paralelo de elementos simples y que responden a una cierta jerarquía intentando interactuar con los objetos reales tal como lo haría un sistema neuronal psicológico (Kohonen, 1988, 1998, 2001). Las redes neuronales poseen la característica de asimilar conocimiento en base a las experiencias mediante la generalización de casos, que las convierte en una herramienta interesante en el desarrollo de los modelados

¹ Se siguen los principios del diseño, implementación y evaluación de Sistemas Computacionales Interactivos para su utilización por seres humanos (HCI: Human Computer Interaction), es decir que estudian y tratan de poner en práctica procesos orientados a la construcción de interfaces lo más usables posible, es decir con alto grado de facilidad en el uso del sistema interactivo. (Estándar ISO 92401 de requisitos ergonómicos para el trabajo de oficina con terminales visuales).

de la presente investigación (Haykin, 1999; Nilsson, 2001)

3. El problema del modelado del estudiante.

Se busca proponer un ambiente de enseñanza y aprendizaje interactivo, que se pueda adaptar a las características del estudiante. Este sistema está pensado con componentes que puedan ser reutilizables (Salgueiro *et al.*, 2005 a,b), por lo que se busca una modularización que sea *independiente del dominio*. Cuando el estudiante ingresa al sistema por primera vez, éste debe tomar sus características a fin de obtener no solo el nivel de conocimientos sino también el modelo de estudiante, así este modelo permitirá seleccionar la estrategia didáctica más adecuada para cada estudiante.

La principal característica que hace al sistema tutor *“inteligente”* es su capacidad para adaptarse a los estudiantes, por este motivo, el problema del modelado de alumno en la metodología, diseño y desarrollo es un tema crítico, ya que es el núcleo del sistema. Shute (1995) realizó un estudio buscando entre los investigadores en el tema un significado de lo que era *“inteligente”* en un tutor que se puede sintetizar en: *“Casi todo el mundo coincide en que el elemento más crítico es el diagnóstico cognitivo (o modelado del alumno). La siguiente característica más citada es la adaptación en la asistencia. Y aunque algunos sostienen que la asistencia forma parte de la “T” de STI, nuestra postura es que las dos componentes, (diagnóstico y asistencia), trabajando conjuntamente constituyen la inteligencia de un STI”*.

El modelo del estudiante entonces, se debería describir en términos de los estilos de aprendizaje a fin de detectar las fortalezas y las debilidades de los mismos (Felder *et al.*, 1988, 1990), a fin de que el sistema lo pueda asistir en su recorrido, con el tutorizado que más se adapte a sus necesidades. Por ese motivo, el STI deberá disponer la representación del estado actual del conocimiento del alumno en todo momento. Millán (2000) señala la

importancia de la adaptación de la enseñanza a cada estudiante a partir del estudio de Bloom (1984) que avala la instrucción individualizada como la forma más efectiva de aprendizaje². *“Estos resultados constituyen un argumento más a favor del uso de los Sistemas Tutores Inteligentes, ya que demuestran los buenos resultados de la enseñanza individualizada”* Millán (2000).

De este modo una vez identificadas las debilidades del alumno, se deben generar los problemas a fin de poder presentar a cada estudiante los conjuntos de ejercicios y problemas específicos según sus necesidades. Una vez llegado a este punto, se debe saber si el estudiante está en condiciones de continuar con el siguiente tema del currículo de acuerdo a sus conocimientos actuales. Algunos autores como Millán (2000) indican que. *“El modelo del alumno es la componente del STI que representa el estado actual del conocimiento del alumno, y el proceso que manipula esta estructura se llama diagnóstico. Ambas componentes deben diseñarse juntas, y este problema de diseño es el que se conoce como el problema del modelado del alumno”*, siendo los elementos fundamentales:

- *La selección de la estructura* que se usará para representar el modelo del alumno, información que se puede almacenar: en un vector, en una red semántica, en una red bayesiana, en forma de afirmaciones, etc.
- *La inicialización del modelo del alumno*. La estructura elegida para representar el conocimiento del alumno debe inicializarse al comienzo de la interacción con el sistema, usando la información disponible acerca del alumno, a fin de poder clasificarlo en clases o tipos previamente definidos, realizar pruebas psicológicas previas, etc.

² Bloom concluyó que cuando se usan los métodos de enseñanza convencionales (relación profesor/alumno entre veinticinco 25 y treinta, con exámenes periódicos) las calificaciones obtenidas por los alumnos poseen una distribución normal, con media entre 50 y 60% pero con una desviación típica grande. Cuando el profesor adapta sus clases orientadas a evitar los errores que sus alumnos cometen en los exámenes, las medias se mueven hasta un 84% y la desviación típica disminuye considerablemente, siendo el cambio más grande cuando los alumnos reciben enseñanza individualizada donde la media llega a un 98%, con una desviación típica que es la mitad de la desviación típica de los alumnos que recibieron una enseñanza convencional.

– *El diagnóstico.* Una vez inicializado el modelo del alumno comienza la interacción con el sistema. Luego de interactuar con el mismo, el procedimiento de diagnóstico actualizará el modelo utilizando dos fuentes de información a) el modelo del alumno

actual y b) su comportamiento en el proceso interactivo de enseñanza, que se puede medir en función de distintas variables tales como: las soluciones a los problemas, las respuestas a las preguntas, el tiempo empleado en lectura de pantallas, etc.

<i>Modelo de superposición (overlay model).</i>	En este enfoque se considera que el conocimiento del alumno es un subconjunto del conocimiento del experto y trata de explicar las diferencias entre el comportamiento del alumno y el del experto a través de la falta de conocimiento del alumno. Este modelo funciona cuando el objetivo principal del sistema instructor es transmitir el conocimiento experto al alumno, pero no considera que el alumno puede tener algún conocimiento que el experto no tiene.
<i>Modelo diferencial (differential model).</i>	Este modelo es una modificación del modelo de superposición, que divide el conocimiento del alumno en dos categorías: conocimiento que el alumno debería poseer y conocimiento que no se puede esperar que el alumno tenga. A diferencia del modelo de superposición, el modelo diferencial reconoce y trata de representar explícitamente tanto el conocimiento del alumno como las diferencias entre el alumno y el experto. Puede considerarse como un modelo de superposición, pero en lugar de superponer sobre el conocimiento del experto, se hace sobre un subconjunto de éste.
<i>Modelo de perturbación (perturbation model).</i>	El modelo de superposición representa el conocimiento del alumno en términos del conocimiento <i>correcto</i> , pero el modelo de perturbación lo combina con una representación del conocimiento incorrecto, es decir, no se considera al alumno como un <i>subconjunto</i> del experto, sino que el conocimiento del alumno puede ser potencialmente diferente en calidad y cantidad al del experto. Este modelo se implementa representando el conocimiento experto con el agregado de los errores que cometen los alumnos más a menudo. En la literatura se observan dos tipos de errores: de concepto (ó <i>misconceptions</i>) y las fallas más comunes (ó <i>bugs</i>). El conjunto de los errores que se incluye en un modelo de perturbación se denomina <i>biblioteca ó catálogo de errores</i> y se puede construir a través de un análisis empírico ó por enumeración, ó bien generando los errores a partir de un conjunto de errores de concepto subyacentes usando técnicas generativas.
<i>Modelo centrado restricciones.</i>	Este modelo es una modificación del modelo de superposición propuesto por Ohlsson (1994) y fue implementado en el tutor de SQL de Mitrovic (1998) (Mitrovic y Ohlsson, 1999). El dominio del conocimiento se representa a través de una serie de restricciones sobre el estado de los problemas, y el modelo del estudiante es una lista de las restricciones que éste ha violado en el proceso de resolución del problema. Este enfoque las ventajas de su robustez y flexibilidad. Es robusto ya que no depende de la estrategia que haya seguido el estudiante para resolver el problema y permite modelar a estudiantes con patrones de comportamiento inconsistentes, que utilicen estrategias diferentes para problemas diferentes. Es flexible ya que permite reconocer las soluciones innovadoras como correctas.

Tabla 1: Tipos de modelos de estudiante (Millán, 2000).

En este sistema el modelo del estudiante puede ser utilizado con los propósitos de: 1) Determinar si el estudiante está listo para continuar con el siguiente tema, 2) Generar explicaciones en detalles de acuerdo a los

conocimientos actuales del estudiante, 3) Ofrecer ayudas y sugerencias al estudiante, sin interrumpir demasiado pero que les ayude a aprender de sus errores, 4) Generar problemas al nivel requerido en forma dinámica y

ejercicios diferentes adecuados a las necesidades particulares y 5) Seleccionar y presentar la estrategia tutorial más apropiada al nivel de conocimiento actual de las disponibles. En la Tabla 1 se resumen los tipos de modelo del estudiante más utilizados.

4. Descripción de la solución

De acuerdo a la estructura clásica de los sistemas tutores inteligentes, se deben redefinir los componentes básicos y las interfaces que posee el módulo del estudiante para llevar a cabo la tarea de representar el estado de conocimiento del estudiante real en forma efectiva. Si bien se busca la separación de los tres módulos fundamentales de los sistemas tutores inteligentes, las implementaciones posteriores han demostrado que el dominio no se puede separar completamente de los módulos del tutor y del estudiante. Como primer paso se analizó la interacción de los módulos para un dominio hipotético general, y luego se buscó realizar las modificaciones para un dado dominio (para ello, se tomó el caso particular de la enseñanza del lenguaje de Programación en Pascal).

4.1. La interacción de los módulos fundamentales.

La Figura 1 muestra los tres módulos fundamentales y la interacción de los mismos en un dominio cualquiera. Este enfoque desliga completamente al módulo del dominio de cualquier tipo de solapamiento con los demás módulos. Las interacciones que se suceden se pueden resumir de la siguiente manera:

a. Determinación del estilo del estudiante:

Se entrega al estudiante una planilla con preguntas para categorizarlo dentro de los estilos de estudiantes disponibles en el sistema. El estudiante entrega la planilla completa y el modelo del estudiante lo categoriza dentro de alguno de los estilos disponibles. esta acción se sucede una sola vez.

b. Generación del estado de conocimientos:

Basándose en el modelo del dominio, el modelo del estudiante se encarga de generar el estado de conocimientos del estudiante; el cual se actualizará a medida que el usuario se vaya capacitando tanto en acción tutelar con el sistema como en tutorías externas en clase. En este último caso, el sistema deberá pasar el horario de una clase con un tutor humano, y la lista de estudiantes presentes generando el nuevo estado de conocimientos del estudiante.

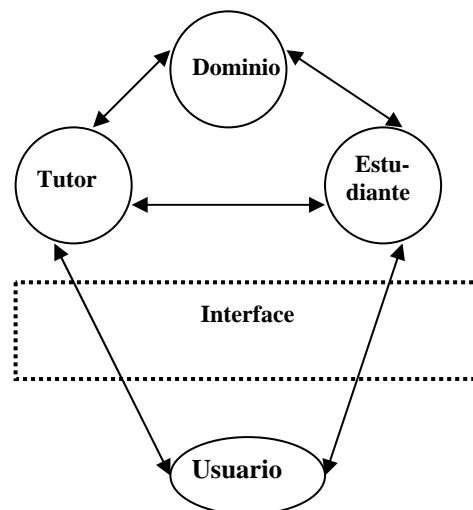


Figura 1: Interacción de los Módulos de un Sistema Tutor inteligente.

c. Determinación de estilo pedagógico:

El modelo del estudiante envía el estilo en el cual cuadra el usuario; con ello el módulo tutor selecciona el estilo pedagógico mas adecuado a las características del estudiante para impartir las lecciones.

d. Planificación de la lección:

En base al estilo pedagógico seleccionado se gestiona al módulo del estudiante el estado de conocimientos del estudiante. En base al estilo y al estado de conocimientos se planifica una lección y se gestiona sobre el módulo del dominio los temas que se requieren en la lección en el orden específico en los que fueron planificados por el módulo tutor.

e. **Sesión pedagógica:** El módulo tutor imparte los conocimientos planificados en la lección de acuerdo al estilo pedagógico seleccionado. A esto se le dará la forma del lenguaje natural para facilitar la interacción con el usuario del sistema y se lo presentara a través de la interfaz, la cual ubicara los elementos pedagógicos utilizados para esa lección en particular. Dada una sesión pedagógica, el módulo del tutor tendrá objetivos generales a cumplir (como puede ser la resolución de un problema en particular, alcanzar cierto grado de conocimiento sobre uno de los temas de la curricula, etc.), pero pueden surgir a lo largo de la interacción con el usuario objetivos secundarios, necesarios o no, para alcanzar el objetivo principal de la sesión pedagógica. Será responsabilidad del módulo tutor guiar al usuario hacia todos los objetivos, manejándolos en forma correcta y no dar por terminada la sesión hasta que todas las metas hayan sido satisfechos.

f. **Evaluación de sesión pedagógica:** Los resultados de la sesión son compilados por el módulo tutor e interpretados. Se parte de preguntas, ejercicios, etc. y se obtiene como resultado el conocimiento (o no) de los temas impartidos. Con esto también se actualiza la pila de objetivos a cumplir para la sesión que se esta llevando a cabo.

g. **Actualización del mapa de conocimientos:** Con los resultados de la evaluación procesados sobre los temas impartidos el módulo del estudiante actualiza el estado de conocimientos del usuario, con esto puede modificar sus predicciones sobre el tipo de aprendizaje del mismo, sus creencias sobre el dominio y puede obtener estadísticas.

A partir de las interacciones descritas se pueden plantear los componentes básicos del módulo del tutor y del módulo del estudiante.

Se detallan los componentes y la interacción entre ellos.

4.2. Los componentes básicos del módulo del estudiante

En la Figura 2 se observa el módulo del estudiante que contiene dos grandes submódulos cuyas características se resumen a continuación:

– **Módulo de estilos de aprendizaje:** Esta compuesto por una base de datos de estilos de aprendizajes disponibles en el sistema tutor inteligente, y los métodos de selección y características de cada uno de los métodos con respecto a los elementos propios del estudiante. Estos últimos varían con respecto al estado del sistema: si el estado es inicial, es decir que el estudiante recién es cargado en el sistema, se lo debe evaluar para generar una representación estándar del estilo de aprendizaje (la cual después puede resultar incorrecta) y a lo largo de las sesiones tuteladas con el sistema se ira actualizando o modificando la visión del sistema con respecto al estilo de aprendizaje real del estudiante. Esto hace que el sistema sea muy versátil a la hora de categorizar y recategorizar los estilos de los estudiantes.

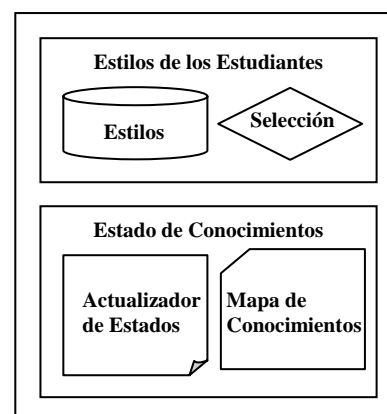


Figura. 2: Componentes básicos del módulo del estudiante.

– **Módulo de estado de conocimientos:** Este contiene el mapa de conocimientos obtenido inicialmente del módulo del dominio que progresivamente el actualizador de

conocimientos irá modificando a través de los resultados obtenidos de las evaluaciones efectuadas por el módulo del tutor, el cuál enviará dichos resultados procesados. Estos datos también se le proporcionaran al módulo tutor para que éste pueda decidir cómo y qué impartir en cada una de las sesiones con el estudiante. La idea de este módulo es contener una representación del estado de conocimientos instantáneo del estudiante. Deberá ser modificado por sesiones tuteladas fuera del sistema, como pueden ser las clases particulares o magistrales con tutores humanos.

Estos dos grandes sub-módulos interactúan entre si y con el resto de los módulos para componer el módulo del estudiante de un sistema tutor inteligente.

5. Conclusiones y líneas de acción.

La redefinición del módulo del estudiante, en submódulos y con interfaces bien definidas busca solucionar algunos problemas generales que afectan a los sistemas tutores inteligentes desde hace años que es el solapamiento de funcionalidades. Esto facilitará los circuitos y se mejorará la interacción entre los módulos, separando completamente al módulo en cuestión del verdadero dominio de aplicación. Con este enfoque no se rompe con la estructura clásica de los sistemas tutores inteligentes, sino que esta nueva visión supone una mayor flexibilidad y adaptación de nuevos dominios a las aplicaciones que la utilicen, como ya se explicó. Actualmente, se está trabajando en el rediseño de los módulos que corresponden al estudiante como se plantea, para integrarlo al módulo tutor del módulo del tutor y al sistema evaluador con predicción y diagnóstico a fin de obtener un sistema tutor inteligente flexible. A esta etapa sobreviene la selección de las tecnologías disponibles más aptas en cada caso. Así, una vez llevada a cabo la implementación e integración de los módulos, el paso siguiente será contrastar el sistema con los datos recabados hasta el momento para generar conclusiones al

respecto, indicando la viabilidad de la implementación y posterior utilización efectiva de un sistema tutor o asesor inteligente como complemento de las clases teóricas en un curso tomado como piloto ó prueba.

6. Referencias

- Bloom, B. (1984). The 2 sigma problem: The search for methods of group instruction as effective as one-to-one tutoring. *Educational Researcher*, 13, 4-15.
- Brown, E.; Palincsar, B. (1989) *Guided, Cooperative learning and the individual knowledge acquisition*. En Resnick B. (comp.) *Knowing, learning, and instruction*. Hillsdale. N.J.
- Brown, J. y Burton, R. 1978 *Diagnostic Models for Procedural Bugs in Mathematical Skills*, *Cognitive Science*, no. 2, pp. 155-192.
- Burton, R. R.; Brown, J. S. (1981). An investigation of computer coaching for informal learning activities. In: Sleeman, D., Brown, J. (eds.): *Intelligent Tutoring Systems*, Capítulo 4, p. 79-98, London: Academic Press.
- 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. 13 y 14 de mayo. Universidad Nacional de Río Cuarto. Córdoba. RED UNCI.
- Chipman, P., Olney, A., y Graesser, A. C. (2005). *The AutoTutor 3 architecture: A software architecture for an expandable, high-availability ITS*. Proceedings of WEBIST 2005: pp. 466-473. Portugal: Insticc Press.
- Cho, B. (2000). *Dynamic Planning Models to Support Curriculum Planning and Multiple Tutoring Protocols in*

- Intelligent Tutoring Systems*. Ph.D. tesis, Illinois Institute of Technology.
- Clancey, W. J. (1991). *Intelligent tutoring systems: A tutorial survey*, en App. Artificial Intelligence: A Sourcebook. McGraw-Hill.
- 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.
- DiPaolo, R.E., Graesser, A.C., Hacker, D.J., White, H.A., y TRG (Tutoring Research Group) (2002). *Hints in human and computer tutoring*. In M. Rabinowitz (Ed.), *The impact of media on technology of instruction*. Mahwah, NJ: Erlbaum.
- Felder R.M.; Silverman L.K. (1988). *Learning Styles and Teaching Styles in Engineering Education*. Engr. Ed., v. 78, 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.
- Galvin, T. (1994) Tesis Doctoral: "*Mebuilder: An Object-Oriented Lesson Authoring System for Procedural Skills (9/94)* donde
- Gertner, A. S; Conati, C.; VanLehn, K. (1998). *Learning Procedural help in Andes: Generating hints using a Bayesian network student model*. Research & Development. American Association for Artificial Intelligence.
- Gertner, A.S. y VanLehn, K. (2000). *Andes: A Coached Problem Solving Environment for Physics*. Lecture Notes In Computer Science; Vol. 1839 Proceedings of the 5th International Conference on Intelligent Tutoring Systems. Pages: 133 - 142
- Giraffa, L.; Nunes, M. A.; Viccari, R.M. (1997) *Multi-Ecological: A Learning Environment using Multi-Agent architecture*. MASTA'97: Multi-Agent System: Theory and Applications. Proceedings.. Coimbra: DE-Universidade de Coimbra.
- Graesser, A.C., Chipman, P., Haynes, B.C. y Olney, A. (2005a). *AutoTutor: An intelligent tutoring system with mixed-initiative dialogue*. *IEEE Transactions in Education*, 48, 612-618.
- Graesser, A.C., Jackson, G.T. y McDaniel, B. (in press2006). *AutoTutor holds conversations with learners that are responsive to their cognitive and emotional states*. *Educational Technology*.
- Graesser, A., Olney, A., Haynes, B. y Chipman, P. (2005b). *AutoTutor: A cognitive system simulates a tutor that facilitates learning through mixed-initiative dialogue*. *Cognitive systems: Human cognitive models in systems design*. Mahwah, NJ: Erlbaum
- Hume G., Michael, J; Rovick, A.; Evens, M. (1996), *Hinting as a tactic in one-on-one tutoring*. *Journal of Learning Sciencies*
- Hume, G.; Evens, M. (1992) *Student modeling and the classification of errors cardiovascular intelligent tutoring system*. Proceedings of the 4th Midwest Artificial Intelligence and Cognitive Science Society Conference, Utica, IL.
- Johnson, W. L. (1986). *Intention-based diagnosis of novice programming errors*. Morgan-Kauffman.
- Khuwaja, R. (1994). *A Model of Tutoring: Facilitating Knowledge Integration Using Multiple Models of the Domain*, Ph. D. Dissertation, Illinois Institute of Technology.
- Kim, J. H. (1989). *CIRCSIM-Tutor: An Intelligent Tutoring System for Circulatory Physiology*. Ph.D. Th., Illinois Inst. of Tech.
- Kim, J. H. (2000) *Natural Language Analysis and Generation for Tutorial Dialogue*. 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.
- Litman D. J. and Silliman. S. 2004. *Itspoke: An Intelligent Tutoring Spoken Dialogue System*. In Proceedings of the Human Language Technology Conference: 4th Meeting of *HLT/NAACL*. (Companion Proceedings), Boston, MA, May.
- Matsuda, N., & VanLehn, K. (2005, to appear). *Advanced Geometry Tutor: An intelligent tutor that teaches proof-writing with construction*. In Proc. of The 12th International Conference on Artificial Intelligence in Education. (acceptance rate: 0.31)
- Milik, N., Marshall, M., Mitrovic, A. (2006) *Teaching Logical Database Design in ERM-Tutor*. M. Ikeda, K. Ashley, and T.-W. Chan (Eds.): ITS 2006, LNCS 4053, pp. 707-709.
- Millán, E. (2000) *Sistema bayesiano para modelado del alumno*. Tesis Doctoral Universidad de Málaga.
- Mitrovic, A. An intelligent SQL tutor on the Web Int. *J. Artificial Intelligence in Education*, vol. 13, no. 2-4, 2003, 173-197.
- Mitrovic, A., Martin, B. y Mayo, M. *Using evaluation to shape ITS design: Results and Experiences with SQL-Tutor*. Int. J. User Modeling and User-Adapted Interaction, vol. 12, no. 2-3, pp. 243-279, 2002.
- Mitrovic, A., Suraweera, P., Martin, B., Zakharov, K., Milik, N., Holland, J. (2006) *Authoring constraint-based tutors in ASPIRE*. M. Ikeda, K. Ashley, and T.-W. Chan (Eds.): ITS 2006, LNCS 4053, pp. 41-50.
- Perkins, D. (1995). *La escuela inteligente*. Gedisa.
- Rowe N. C. and T. Galvin, (1998) An authoring system for intelligent tutors for procedural skills. *IEEE Intelligent Systems*, 13, 3 May/June, 61-69.
- Salgueiro, F. A, Costa, G., Cataldi, Z., García Martínez, R. y Lage, F. J. (2005a). *Sistemas inteligentes para el modelado del tutor*. GCETE'2005, Global Congress on Engineering and Technology Education. marzo 13-15.
- Salgueiro, F; Costa, G., Cataldi, Z., Lage, F., García-Martínez, R. (2005b). *Redefinition of basic modules of an intelligent tutoring system: the tutor module*. WICC 2005. WICC 2005. 13 y 14 de mayo. Universidad Nacional de Río Cuarto. Córdoba. RED UNCI.
- Shah, F. (1997). *Recognizing and Responding to Student Plans in an Intelligent Tutoring System: Circsim-Tutor* Ph.D. Th., Illinois Ins. of Tech.
- Shim, L. (1991) Student Modeling for an ITS: Based on the Analysis of Human Tutoring Sessions. Ph.D., Illinois Inst. of Tech.y.
- Stevens, A.; Collins, A. (1977). *The goal structure of a Socratic tutor*. Proc.s of the National ACM Conference. New York: ACM.
- VanLehn, K (1988). *Student Modelling*. M. Polson. Foundations of Intelligent Tutoring systems. Hillsdale. N.J. Lawrence Erlbaum Associates, 55-78.
- Waern, A. (2001) *What is an intelligent interface?*. Notas de Seminario de introductorio. Centro de Investigaciones en Computación de Suecia. Consultado el 13/03/06, disponible en <http://www.sics.se/annika/publications.html>.
- Wolf, B. (1984). *Context Dependent Planning in a Machine Tutor*. Ph.D. Dissertation, University of Massachusetts, Amherst, Massachusetts.