

Sistemas Tutores Inteligentes: El submódulo generador de contenidos

Zulma Cataldi, Fernando A. Salgueiro, Fernando J. Lage

LIEMA - Laboratorio de Informática Educativa y Medios Audiovisuales. Facultad de Ingeniería. UBA.
Facultad Regional Buenos Aires. Universidad Tecnológica Nacional
liema@fi.uba.ar, fsalgueiro@fi.uba.ar, flage@fi.uba.ar

Abstract

In this communication is presented the analysis of the generating contents submodule in the model of ITS (Intelligent Tutoring System). Once selected the pedagogical protocol more adapted by the analyzing profile submodule is necessary to generate the contents of the pedagogical session to being displayed the user of the system. The function of the generator of contents is to handle to the interaction between the system and the user.

Keywords: Intelligent Tutoring Systems, lesson generator.

Resumen

En este artículo se presenta el análisis realizado del submódulo generador de los contenidos dentro del modelo planteado de STI (Sistema Tutor Inteligente). Una vez seleccionado el protocolo pedagógico más adecuado por el submódulo analizador de perfil, se deben generar los contenidos de la sesión pedagógica a ser presentados al usuario del sistema. La función del generador de contenidos es manejar la interacción entre el sistema y el usuario.

Palabras Clave: Sistemas Tutores Inteligentes, generador de contenidos.

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 [1,2]. Su propósito es presentar un comportamiento similar al de un tutor humano, que se adapte a las necesidades del estudiante, identificando la forma en que el mismo resuelve un problema para poder brindarle ayuda cuando cometa errores. Las interacciones entre el alumno y el docente, están enmarcadas en las teorías de aprendizaje y de enseñanza aplicables a dicha interacción, lo que brinda un marco teórico al problema

citado, donde no solo es importante el conocimiento que debe ser facilitado por el docente hacia el alumno, sino que cobra importancia la forma en la que este conocimiento es presentado, ya que se pretende mejorar el proceso de adquisición y construcción de conocimiento.

Se busca la incorporación de los métodos y técnicas de enseñanza más eficaces que permitan adaptar el modo de enseñanza a las necesidades del alumno a fin de mejorar su rendimiento a través de cada clase o “sesión pedagógica”. Cada alumno podrá elegir entonces la técnica de enseñanza que mejor se adapte a su estilo de aprendizaje.

Wenger [3] define la tarea de la enseñanza en términos de una “comunicación del conocimiento” que involucra a un sistema tutor y a un estudiante, donde el objetivo básico es que el estudiante pueda adquirir conocimiento en algún dominio. Se busca un sistema de tutorizado que exhiba un comportamiento similar al de un tutor humano, es decir, que en lugar de ser un modelo rígido se adapte al comportamiento del estudiante. Éste, debería ser capaz de identificar la forma en cual el estudiante está resolviendo el problema [4] y brindarle ayuda cuando cometa errores [5]. A su vez debería proveerle el conocimiento que requiera para poder solucionar el problema con explicaciones en el momento preciso, y adicionalmente podría aprender de la interacción con el estudiante.

2. Los módulos principales del STI

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” [3]. Wolf [6] 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 [7] 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".

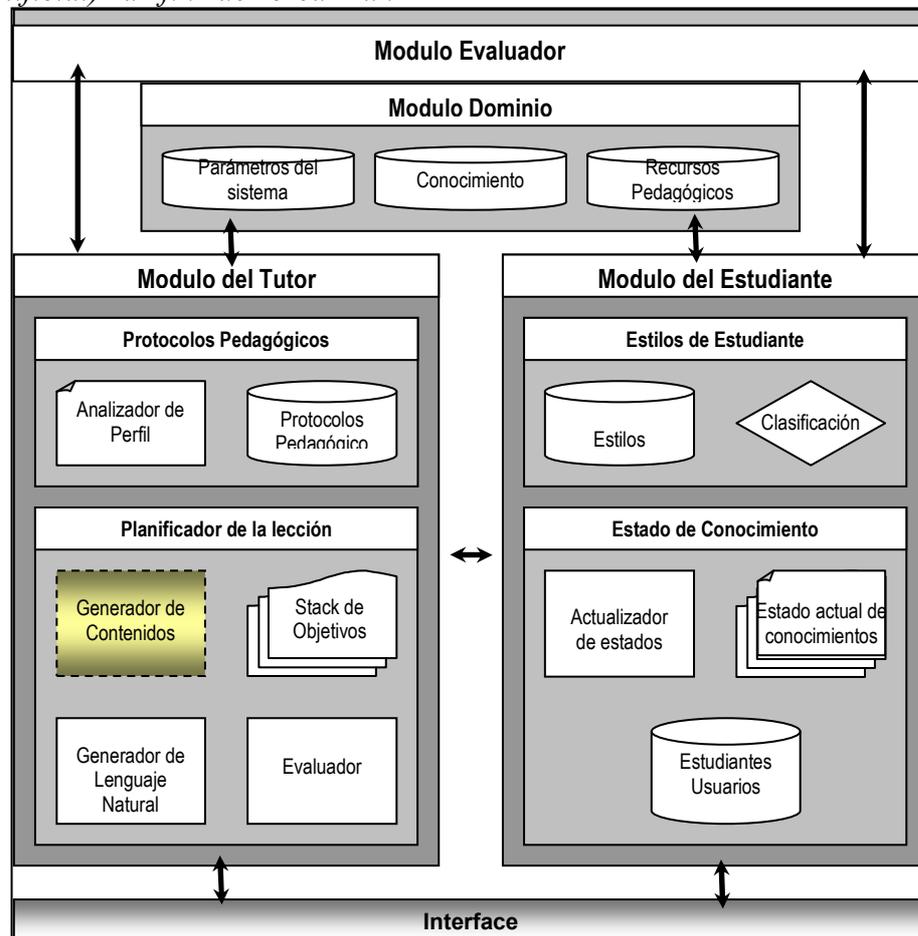


Figura 1: Modelo detallado de un Sistema Tutor Inteligente

Estos sistemas se basan en una arquitectura compuesta por tres grandes módulos: el módulo del tutor, el módulo del alumno y el módulo del dominio a los que podría agregarse un cuarto módulo denominado el módulo de evaluación y el de ecosistema [8] para interactuar con los pares que se conecten al STI.

1. *Módulo del Alumno:* Este módulo debe representar el estado inicial del alumno y sus características individuales, entre ellas una de las más importantes es el conocimiento individual instantáneo sobre el dominio que Guardia Robles [9] define como: "El modelo del estudiante, que refleja cuánto conoce el estudiante sobre el dominio, así como las experiencias cognitivas y de aprendizaje que

ha llevado, del cual puede obtenerse un diagnóstico."

2. *Módulo del Tutor:* Este módulo posee el conocimiento sobre las estrategias y tácticas de enseñanza para poder seleccionarlas en función de las características del alumno, que están almacenadas en el módulo del alumno [10]. Pero debe ir más allá de la experiencia en el dominio, ya que debe ofrecer a cada estudiante un método de enseñanza de acuerdo con sus necesidades.

3. *Módulo del Dominio:* Este módulo posee el conocimiento de la asignatura formado por las reglas de producción, estereotipos, etc. El módulo tutor obtiene de aquí el conocimiento que debe enseñar definido como: "El modelo experto o del dominio, el

cual versa sobre la materia o curso que se impartirá” [9].

4. *Módulo de Evaluación:* Se encarga de realizar una evaluación general del sistema y generar estadísticas acerca de los avances de los estudiantes; pudiendo efectuar el diagnóstico evolutivo luego de cada uno de los estados considerados, de este modo podría también predecir el comportamiento en los eventos futuros. La evaluación de los estudiantes debe ser constante y durante el proceso, con instancias de autoevaluación. También se deberán generar informes a utilizar para evaluar al sistema como método apto de enseñanza.

5. *Módulo de Interface:* Es la interface de interacción entre el STI y el alumno real, la que se encarga de presentar el material del

dominio y cualquier otro elemento didáctico de la manera correcta. “La interface, que permite a los usuarios interactuar con el sistema. Se distinguen tres tipos específicos de usuarios: el estudiante, el instructor y el desarrollador del sistema” [9]. Para su diseño se deben seguir los criterios ergonómicos tales como los basados en el estándar ISO 9241 para *Human Computer Interface* (HCI) u otros similares.

Los primeros tres módulos que se pueden ver en la Figura 2 conforman la arquitectura clásica (propuesta por Carbonell [11]) y funcional de los STI. Esta postura presentó notables avances en el modelado de ambientes educativos, ya que separó el dominio de la forma en la que éste es utilizado [10].

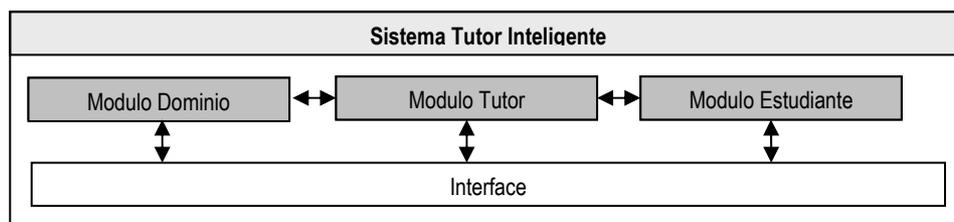


Figura 2: Estructura clásica de un Sistema Tutor Inteligente propuesta por Carbonell [11].

Existen algunas visiones de los Sistemas Tutores Inteligentes (STI) que no son clásicas, como la propuesta de arquitectura general en Clancey [12] que se puede resumir de la siguiente manera:

- *Son dos sistemas expertos:* Uno encargado de diagnosticar el estado actual del alumno y otro encargado de decidir la mejor forma de retroalimentarlo en el momento actual, cumpliendo el rol de asesor.
- *Poseen un sistema experto de diagnóstico:* Con un algoritmo de decisión sobre la mejor estrategia de retroalimentación, basado en una representación del conocimiento sobre el estado cognitivo del alumno.

Se piensa que un sistema que pueda emular al tutor humano y que además provea al estudiante de cierta flexibilidad para la elección del tipo de tutorizado más adecuado, puede ser una solución factible para el problema de la falta de tutores. En la literatura

analizada se han encontrado dos posturas para la implementación de los conocimientos: una se basa en la estructura sintáctica de lo producido por los tutores humanos [13] y la otra se centra en las metas pedagógicas que deben cumplir a fin de que el alumno pueda comprender el tema [5, 14]. Analizando el problema y utilizando ambas teorías en forma conjunta se podrían obtener una serie de pasos que pueden resumir la forma de impartir los conocimientos [15]:

- *El tutor debe mantener una jerarquía de metas:* que debe cumplir mientras imparte los conocimientos al alumno quien producirá un resultado que el tutor no puede predecir de antemano.
- *El tutor debe poder explicar un mismo concepto de diferentes maneras:* de este modo, si el alumno no entiende un concepto el tutor puede continuar efectuando otro acercamiento al mismo tema, explicando el concepto gradualmente, utilizando un

método iterativo para profundizar en el concepto cada vez más (paso a paso) o para descartar este acercamiento al tema e intentarlo de otra manera.

En este contexto, surgen las posibilidades de aplicabilidad de los sistemas inteligentes a la resolución de problemas de modelado de este tipo. En el campo de los Sistemas Inteligentes se encuentran las Redes Neuronales (RN), que son interconexiones masivas en paralelo de elementos simples, que responden a una cierta jerarquía intentando interactuar con los objetos reales tal como lo haría un sistema neuronal psicológico [16]. Las redes neuronales poseen una característica que las hace muy interesantes, dado que pueden asimilar conocimiento en base a las experiencias mediante la generalización de casos.

3. El submódulo de Generador de Contenidos.

Una vez seleccionado el protocolo pedagógico más adecuado por el submódulo Analizador de Perfil [17], se deben generar los contenidos de

la sesión pedagógica a ser presentados al usuario del sistema. La función del Generador de Contenidos (GC) es manejar la interacción entre el sistema y el usuario para cada lección. En la Figura 1 se puede ver un modelo general de un STI donde se observa el sub-módulo Generador de Contenidos dentro del Planificador de la Lección (ver Figura 1).

Luego de la selección del protocolo pedagógico, el paso siguiente es encontrar cuáles son los conceptos a explicar en la lección. Esto se realiza comparando el mapa de conocimientos (que se obtiene del módulo del dominio) y los conocimientos que el sistema supone que el estudiante posee (que se obtiene del módulo del estudiante).

Los conocimientos del estudiante se deben “superponer” con el mapa de conocimiento que se obtiene del módulo de dominio para determinar los conocimientos faltantes. Se agregarán a la pila (*stack*) de objetivos de la lección, de acuerdo a una función que destaque la importancia parcial de cada uno de los temas o al tiempo disponible.

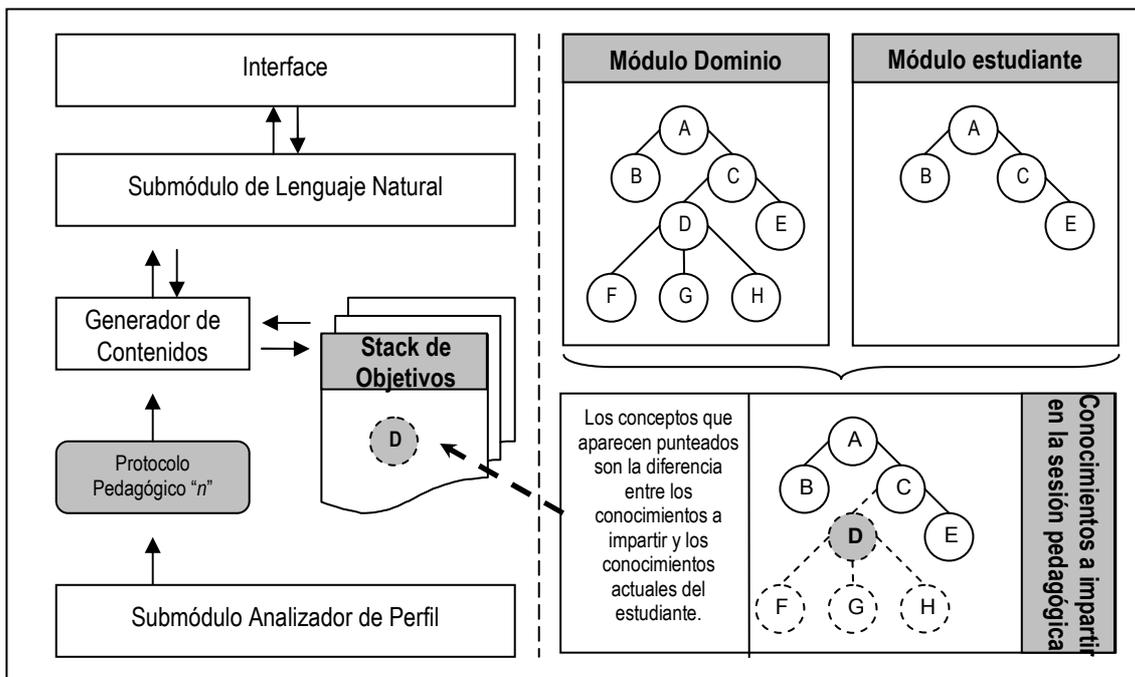


Figura 3: Esquema general de los pasos a seguir luego de la obtención del protocolo pedagógico por el submódulo analizador de perfil y de la creación de los contenidos de la lección.

El *generador de contenidos* se debe encargar de presentar los contenidos en orden y de

procesar las respuestas de los estudiantes, agregando los conceptos que el alumno tuviera

mal clasificados respecto de los que están en el sistema, al *stack* de objetivos de la lección e informando sobre esta discrepancia entre la realidad y lo supuesto por el sistema al módulo del estudiante. Cuando el sistema detecta que el estudiante ha adquirido un nuevo concepto o que se ha corregido un concepto erróneo, el generador de contenidos será el encargado de actualizar el estado de la pila (*stack*) de objetivos y si ésta se encontrase vacía, dará por finalizada la sesión tutelar para la lección.

Una vez que el generador del lenguaje natural, procese las salidas, las enviará directamente a la interface del sistema. Cuando el estudiante responda las preguntas del sistema, es decir, continúe la interacción planteada por el sistema, todos los datos pasarán desde la interface del sistema al módulo de lenguaje natural. La información procesada parte de éste para convertirse en la entrada del generador de la lección. Esta información se evalúa por el mismo y es la que provocará las respuestas pertinentes. El esquema general del funcionamiento del generador de la lección se puede ver en la Figura 3, donde se observa que gestiona los conocimientos del estudiante desde el módulo del estudiante y los conocimientos a explicar desde el módulo del dominio, luego los superpone y detecta cuáles son los conocimientos faltantes en la estructura del estudiante y selecciona uno. Utiliza el protocolo pedagógico resultante del submódulo analizador de perfil y pasa a generar los contenidos requeridos.

Este módulo es el responsable de controlar la interacción entre el tutor y el estudiante, por lo tanto dado un problema en particular debe decidir cómo responder al estudiante [6]. Cada lección se debe planear específicamente y debe ser lo suficientemente flexible y coherente como para respetar la interacción. El ítem de la flexibilidad es central en el planeamiento de la lección, ya que en los primeros tutores, como el tutor *Meno* [6], el STI era capaz de seleccionar los discursos almacenados con anterioridad, pero la falta de control global sobre la lección no permitían la flexibilidad con respecto a las interacciones del alumno. El

grado de la flexibilidad de la interacción está dada por el tamaño de los discursos almacenados.

Por lo tanto, se pueden redefinir las funciones básicas del módulo 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 al estudiante. Los objetivos principales de este módulo se pueden plantear tomando como base las ideas de Woo Woo [18]:

- *Decidir* cuándo presentar los contenidos de la lección, interactuando con los otros módulos del sistema tutor en general y con los submódulos del módulo tutor en particular, como el generador de lenguaje natural, el evaluador, los protocolos pedagógicos, etc.
- *Corregir* los conceptos erróneos del estudiante para un problema dado y generar el feedback del usuario hacia el sistema en todas las respuestas durante la sesión pedagógica.
- *Proveer* los medios para que el estudiante se encauce nuevamente en el camino correcto para solucionar el problema cuando no logra dar con la respuesta correcta. Es aquí donde se insertarán los conceptos de pistas que aporta Hume [5,6]. En este paso es donde esta función difiere levemente con lo planteado por Woo Woo [18], ya que se utilizará el *stack* de objetivos para conocer el estado de la lección.
- La forma en la que el discurso de la sesión se encuentra en el sistema, debe estar en forma explícita, en lugar de almacenado en forma procedural como en el tutor *Meno* [6]. Esto permitirá modificar las estrategias de enseñanza a medida que la lección progresa en el tiempo. Se pueden utilizar las técnicas de “*guiones*” para codificar las acciones posibles en el discurso dentro de cada uno de los temas o unidades didácticas. Así se pueden generar los contenidos de una manera sencilla y lo suficientemente flexible como para obtener un mayor rendimiento de una interacción entre el sistema y el estudiante usuario.

- El sistema debe responder las preguntas del usuario con la respuesta más apropiada respecto de las opciones posibles considerando que solo se están evaluando las interacciones que son iniciadas por el sistema y no por el estudiante.

Pero, a un nivel mucho más general, el módulo generador de contenidos debe cumplir con la “táctica” para lograr los objetivos de la lección. Se puede decir que para un Sistema Tutor Inteligente (STI) los objetivos de la lección del módulo tutor en conjunto, se pueden dividir centrados en:

- *Estrategia o Metodología*: Es la forma en que se cumplirán las definiciones de los objetivos globales para la lección. Éstos determinan la utilidad final de la sesión pedagógica, y dan una media del tiempo que insumirá impartirlos. También la selección del protocolo pedagógico más adecuado es

parte de las estrategias a definir por el sistema para impartir una lección en particular con el estudiante.

- *Táctica o uso recursos*: Son parte de la implementación de la estrategia definida por el módulo tutor. Son las acciones a llevar a cabo en la interacción entre el sistema y el usuario, estas se pueden ir modificando a medida que se reciben las respuestas del estudiante usuario. Todas las tácticas deben actuar de manera de respetar la estrategia establecida.

Ambos conceptos pueden se pueden integrar dentro del módulo tutor en la Figura 4, donde la táctica es llevada a cabo por el submódulo generador de la lección. Con los datos de las características estudiante se obtiene una estrategia proveniente del módulo del tutor que permita cumplir los objetivos de una lección.

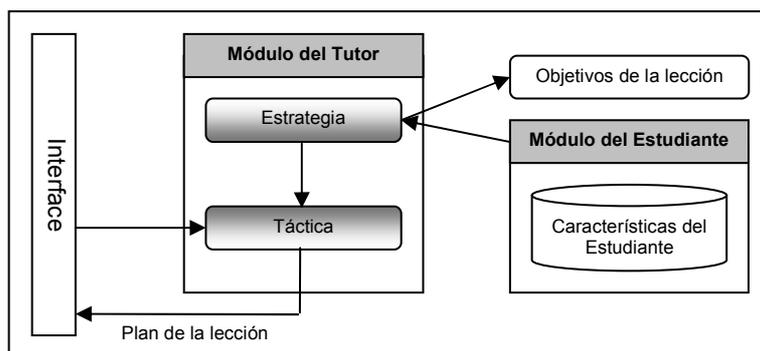


Figura 4: Esquema general del establecimiento de la estrategia (determinación de objetivos) de la lección y la táctica implementada para llevar dicha estrategia a cabo (plan de la lección).

A continuación se detalla la forma de establecer como cumplir con los objetivos de la lección pedagógica (estrategia) y luego se detallarán que tácticas implementan estas estrategias.

4. Los objetivos de la lección.

La determinación de los objetivos de la lección es una tarea esencial que deben llevar a cabo los Sistemas Tutores Inteligentes, ya que se deben generar objetivos globales, coherentes y consistentes para que éstos sean impartidos los estudiantes [18], dado que cada uno de los conceptos almacenados en el módulo del

dominio para cada asignatura estarán relacionados entre sí. La secuencia en la que éstos se presenten afectará de manera decisiva el rendimiento del sistema en su conjunto y también lo hará a las necesidades del estudiante. La determinación de los objetivos dejará planteada la estrategia del sistema para una sesión pedagógica en particular.

Woo Woo [18] plantea la determinación de objetivos utilizando como base una “*tabla de predicción*” que almacena los resultados en el módulo del estudiante. Estas ideas son implementadas en el tutor *CircSim* [5,19,20,21;22,23] y pueden generar un planeamiento tanto de tipo dinámico como de

tipo jerárquico. *CircSim* es un Tutor Inteligente aplicado al dominio de la medicina, más precisamente a las afecciones del sistema circulatorio que está diseñado en forma modular, pero posee algunas funcionalidades que están superpuestas. 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 desconoce, sino que también requiere de metainformación sobre los conceptos a impartir. La metainformación es información adicional acerca del conocimiento y en el caso del módulo del dominio se utiliza para determinar las dependencias entre los distintos conceptos, el tiempo que puede tardarse en impartirlos y la importancia de los mismos, es decir si son un concepto aislado, si son subtemas o si son temas principales o capítulos. Con la metainformación se puede armar el mapa de conocimientos del alumno como el conjunto básico de conocimientos necesarios para finalizar con el programa de la asignatura.

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 respecto de la flexibilidad de un Sistema Tutor Inteligente reside en que éste sea capaz de evaluar los conocimientos adquiridos por el estudiante y poder actuar conforme a ello. Es por esto, que se requiere efectuar el contraste de los datos del módulo del dominio con el mapa de conocimientos instantáneo del estudiante, que se obtiene a través de módulo del estudiante, a fin de encontrar cuáles son los conceptos posibles y disponibles para impartir al estudiante. En la Figura 5 se puede ver un esquema del procedimiento a seguir. El alumno selecciona un protocolo pedagógico para la sesión que sumando a los contenidos provenientes del dominio y la metainformación permite que el sistema pueda impartir la lección cumpliendo con los objetivos de misma. Finalizada ésta se debe actualizar el mapa de contenidos del estudiante.

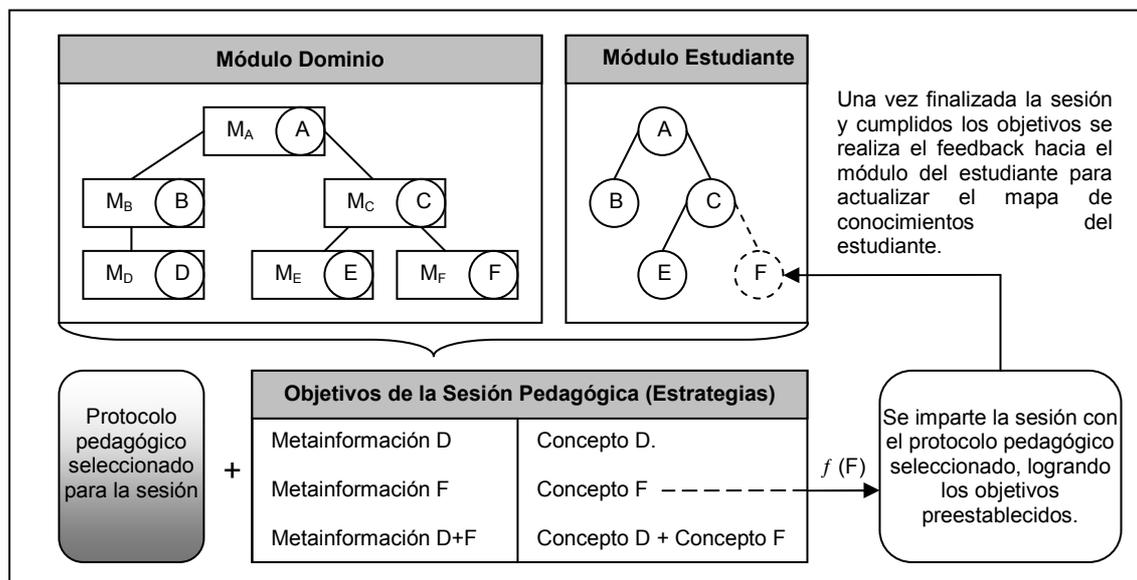


Figura 5: Esquema general para la selección de objetivos para la sesión pedagógica y retroalimentación luego de finalizar dicha sesión.

Una vez determinadas 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 que permita determinar cuál de todas las combinaciones será la seleccionada para la sesión en particular. Se tendrá en cuenta que el número

de opciones puede ser relativamente grande y éstas requieren a veces de un análisis más complejo.

Existen distintas opciones para implementar la función que determinará cual de todas las opciones será la seleccionada para una determinada sesión pedagógica. Por ejemplo, en el tutor *CircSim* [5,19,20,21;22,23] se selecciona el objetivo de la sesión por medio de un conjunto de reglas que toman como base los conocimientos del estudiante [18]. Si bien estas reglas son tomadas de sesiones de educación de conocimiento con tutores humanos fallan en la separación de los módulos, ya que los conocimientos resultantes a enseñar deben provenir del módulo de dominio y no estar escritos en forma (estática) “*Hard-Coded*” dentro del módulo del estudiante. Otra opción es la que utiliza el tutor *Meno* [6] donde se va avanzando linealmente en cada uno de los temas del programa.

Independientemente del método que se utilice para la creación de la función para determinar el objetivo de la lección, los parámetros sobre los que se puede realizar la evaluación están relacionados con los metaconocimientos que almacenan los conceptos en el módulo del dominio. Se pueden citar los siguientes criterios como evaluación de cada una de las opciones:

– *Duración*: Si existe el metaconocimiento, cuánto tiempo llevará en forma aproximada impartir ese concepto en particular así se pueden armar las sesiones pedagógicas de una duración aproximada determinada. El sistema deberá optar entre todas las combinaciones disponibles para lograr sesiones que no sean demasiado extensas (que pueden perjudicar al concentración del estudiante) ni extremadamente cortas (si se explican muy pocos conceptos en una sola sesión esto puede ayudar muy poco al estudiante a lograr un entendimiento global de cada uno de los temas). Otra opción sería preguntarle al estudiante cuánto tiempo está dispuesto a invertir en la sesión y encontrar cuál es la opción que se ajusta mejor a sus requerimientos.

– *Profundidad*: El sistema no debe presentar contenidos de una lección que contengan varias unidades del programa de la asignatura a la vez o que muestren solo algunos conceptos separados. La profundidad de los conceptos a abarcar en el mapa de conocimientos debe ser suficiente para que las sesiones cubran, aproximadamente, una unidad por sesión u otro valor adecuado según parámetros educados de los tutores humanos.

– *Relación con la currícula docente*: La idea de un Sistema Tutor Inteligente, no es reemplazar a un tutor humano, sino de complementarlo. Se puede dar el caso del tutor humano, que recomiende a sus alumnos el uso del tutor solo para refuerzo y que no desee que éstos avancen en el programa hasta que él no explique determinados temas “*claves*” en las clases presenciales. En estos casos se pueden seleccionar los temas no dependientes de las unidades claves que el tutor humano quiere que el sistema no explique sin que antes sean éstos sean introducidos en la clase presencial.

– *Tipo*: Inicialmente el sistema no necesariamente tiene cargada en el módulo de dominio la cantidad necesaria de información para presentar una sesión de acuerdo a todos los protocolos pedagógicos disponibles en el sistema. Se puede dar el caso en que muchas de las opciones de presentación de los conceptos a impartir no estén disponibles en el protocolo pedagógico seleccionado en el sistema. Éste puede ser un criterio para la eliminación de todas esas instancias y la selección de las instancias que utilicen el protocolo seleccionado por el submódulo de protocolo pedagógico que tiene por objetivo presentar el protocolo que provea mejores resultados [24]. En el caso extremo de eliminarse todas las opciones disponibles, se puede volver a realizar el mismo análisis, pero esta vez utilizando el segundo protocolo recomendado por el submódulo de protocolos pedagógicos y así sucesivamente hasta conseguir un conjunto distinto al vacío de donde elegir [17,24].

– *Mixtas y otras*: Existen otras opciones que pueden llegar a ser tan válidas como las

mencionadas. Se pueden utilizar combinaciones de las opciones mencionadas a fin de hallar una función para determinar el objetivo de la lección que sea tan completa y compleja como se requiera en cada uno de los casos particulares de los STI a implementar. Independientemente de las particularizaciones de la función para determinar el objetivo de la lección, se puede plantear un pseudocódigo que explique el funcionamiento y determinación de estos objetivos.

En la Figura 6 se puede ver el 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, para luego encontrar todas las combinaciones posibles de conocimiento, seleccionando por medio de la función F el más apto de ellos.

```

define("Result", "0");

Knowledge = get student knowledge(Student X)
if (Knowledge != Full ) {
    K_list[]=get_domain_dependant_knowledge
(Knowledge);
foreach (K_list as lesson_objetivo) {
K list eval[]= F (lesson objetivo);
Sort (K list eval[], Result);
}
Output(K_list_eval[0])
}

```

Figura 6: Pseudocódigo del objetivo de la sesión.

Este submódulo GC interactúa con varios componentes del STI para ejercer el control entre la interacción entre las preguntas y respuestas entre el sistema y el estudiante. Cuando se procesan las respuestas a una serie de preguntas realizadas por el sistema es el planeador de la lección el que debe decidir qué acción tomará continuación, siguiendo las pautas básicas del protocolo pedagógico seleccionado, por lo tanto el planeador requiere conocimientos en cada punto de la lección para poder comunicarse con el usuario. Esto se logra a través de reglas o “*guiones*”, las cuales se obtienen a través de los expertos aplicando las técnicas de educación de los conocimientos. Este proceso formal se lleva a cabo utilizando la metodología para generar sistemas expertos.

Por otro lado se requiere mantener el estado de la interacción entre el sistema y el estudiante usuario, para ello se mantiene el *stack* de objetivos, donde se almacenan, en la forma de estructura del tipo *LIFO*¹. El generador de contenidos se encargará de conocer el contexto de las respuestas del estudiante con respecto al valor tope del *stack* de objetivos (o a los primeros *n* valores del *stack*).

5. Conclusiones

Se presentó un modelo para el sub-módulo Generador de Contenidos para manejar la interacción entre el sistema y el usuario (Figura 4). Este sub-módulo está dentro del Planificador de la Lección y en relación directa con el *stack* de objetivos, el generador de lenguaje natural y evaluador. Como trabajos posteriores se propone: a) Ampliar los contenidos disponibles para la autoevaluación, b) Escalar el sistema informático para que se pueda realizar un seguimiento del alumno, de esta forma el docente puede tener una *clusterización* de las necesidades cognitivas de su clase. c) Escalar el sistema informático de tal forma que evolucione hacia bases de datos e interfaces capaces de interactuar con el alumno de manera autónoma y d) Integrar los hallazgos en el módulo generador de contenidos en un STI cuya arquitectura se está desarrollando.

6. Agradecimientos

Esta comunicación forma parte de los proyectos de investigación: 25/C099 *Modelado del tutor basado en redes neuronales para un Sistema Tutor Inteligente*, de la Facultad Regional Buenos Aires de la Universidad Tecnológica Nacional 2007-2008 convenio FI-UBA y UTN-FRBA.

7. Referencias

- [1]. Urretavizcaya, M (2001) Sistemas inteligentes en el ámbito de la educación. Revista interamericana de Inteligencia Artificial N. 12. Pp. 5-12

¹ LIFO es el acrónimo en inglés de *Last In – First Out*, o en castellano el último en entrar es el primero en salir. Es la metodología básica para la operación de una estructura del tipo *pila*.

- [2]. Sancho, L (2002) Una alternativa para el uso de computadoras en educación. Educación Net. Red global de educación a distancia (DistEdNET). Universidad estatal a Distancia.
- [3]. Wenger, E. (1987). *Artificial intelligence and tutoring systems. Computational and Cognitive Approaches to the Communication of Knowledge*. Los Altos C. A. Morgan and Kaufman.
- [4]. Hume, G. D. (1995). *Using Student Modelling to Determine When and How to Hint in an Intelligent Tutoring System* Ph.D., Illinois Institute of Technology.
- [5]. Hume G., Michael, J.; Rovick, A.; Evens, M. (1996), *Hinting as a tactic in one-on-one tutoring*. Journal of Learning Sciences.
- [6]. Wolf, B. (1984). Context Dependent Planning in a Machine Tutor. Ph.D. Dissertation, University of Massachusetts, Amherst, Massachusetts
- [7]. Giraffa, L. M. M. (1997). Selecao e adoçao de estratguas de ensino em Sistemas Tutores Inteligentes . Porto Alegre: CPGCC/UFRGS.
- [8]. Cataldi, Z. (2004). *Metodología para el diseño y evaluación de sistemas tutores inteligentes*. FI-UNLP.
- [9]. Guardia Robles, B. (1993). *Asesores Inteligentes para apoyar el proceso de enseñanza de lenguajes de programación*. Tesis de grado. Asesor. ITESM: Instituto Tecnológico de Monterrey.
- [10]. Villareal Goulart R. R.; Giraffa M. L. (2001). *Utilizando a tecnologia de agentes na construção de Sistemas Tutores inteligentes em ambiente interativo*. Anais do XII Simpósio Brasileiro de Informática na Educação.
- [11]. 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.
- [12]. Clancey, W. J. (1991). *Intelligent tutoring systems: A tutorial survey*, en Applied Artificial Intelligence: A Sourcebook. McGraw-Hill.
- [13]. Seu, Jai, Ru-Charn Chang, Jun Li, Evens, M.; Michael, J. and Rovick, a. (1991). *Language Differences in Face-to-Face and Keyboard-to-Keyboard tutoring Session*. Proceedings of the Cognitive Science Society.
- [14]. Evens, M. W.; Spitkovsky, J.; Boyle, P.; Michael, J.; Rovick, A. A. (1993). *Synthesizing tutorial Dialogues*. Preceedings of the 15th Annual Conference of the Cognitive Science Society.
- [15]. Freeva, R.; Evens, M. (1996). *Generating and revising multi-turn text plans in STI*. Evens. Lecture Notes in Computer Science. Pp 632-640.
- [16]. Kohonen, T. (1988). *Self-Organizing Maps Springer Series in Information Sciences*. Vol. 30, Springer, Berlin, Heidelberg, NY. P. 236.
- [17]. Salgueiro, F.; Costa, G.; Lage, F.; Cataldi, Z. and García-Martínez, R. 2006. *Selecting pedagogical protocols using SOM*. RESI, Revista Eletrônica de Sistemas de Informação ISSN 1677-3071, Ed.7, Año V, N°1, junio.
- [18]. 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, 1991.
- [19]. Kim, J. H. (1989). *CIRCSIM-Tutor: An Intelligent Tutoring System for Circulatory Physiology*. Ph.D. tesis, Illinois Institute of Technology.
- [20]. Kim, J. H. (2000) *Natural Language Analysis and Generation for Tutorial Dialogue*. Ph.D. tesis, Illinois Institute of Technology.
- [21]. 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.
- [22]. Shah, F. (1997). *Recognizing and Responding to Student Plans in an Intelligent Tutoring System: CIRCSIM-Tutor* Ph.D. tesis, Illinois Institute of Technology.
- [23]. Hume, G.; Evens, M. (1992) *Student modeling and the classification of errors cardiovascular intelligent tutoring system*. Proc. of the 4th Midwest Artificial Intelligence and Cognitive Science Society Conference, Utica, IL.
- [24]. Cataldi, Z.; Salgueiro, F., Lage, F. (2006). *Selecting teaching method in Intelligent Tutoring Systems through learning styles using neuronal networks* WCCSETE'2006: World Congress on Computer Science, Engineering and Technology Education. Brasil. Marzo 19-22.