

Sistemas Tutoriales Inteligentes

Un análisis crítico

Autor: Constanza Raquel Huapaya

Directora: Prof. Lic. Laura Lanzarini

TRABAJO FINAL INTEGRADOR PARA OPTAR POR EL TÍTULO DE
ESPECIALISTA EN TECNOLOGÍA INFORMÁTICA
APLICADA A EDUCACIÓN

FACULTAD DE INFORMÁTICA

UNIVERSIDAD NACIONAL DE LA PLATA

9 de febrero de 2009

Objetivo

El presente Trabajo Final tiene por objetivo describir los Sistemas Tutoriales Inteligentes (STIs) y realizar un análisis crítico de sus fortalezas y debilidades.

Para alcanzar tal fin, se describirá la arquitectura tradicional de los STIs, se delinearán teorías de aprendizaje e instruccionales que provean el marco conceptual a estos sistemas y se presentarán herramientas de diseño. Asimismo, se analizarán las características principales de estos sistemas, focalizando el problema de su adaptación a los estudiantes.

Prefacio

Si bien no existe unanimidad sobre el significado del concepto Sistema Tutorial Inteligente (STI), la comunidad del área coincide en que son sistemas instruccionales, adaptativos a los estudiantes, y que procuran emular los probados beneficios de la enseñanza uno-a-uno. Los STIs tienen como objetivo imitar a los tutores humanos en su habilidad para determinar en cada caso *qué* enseñar, *cuándo* enseñar y *cómo* enseñar, en lo posible, de un modo autónomo.

Los Sistemas Tutoriales Inteligentes se originaron en el momento que la Inteligencia Artificial (IA) estaba trabajando en el trascendente objetivo de imitar la inteligencia natural mediante la creación de máquinas que ‘pensaran’ como los humanos. Los investigadores se toparon con problemas intratables cuando intentaron alcanzar esta meta. La emulación de la cognición humana con computadoras no alcanzaba el éxito deseado porque se partía de un principio equivocado: asumir que las personas piensan como una computadora. La crisis resultante provocó la revisión de los objetivos de la IA, y como consecuencia, el progreso del área en temas como los Sistemas Expertos. Estos sistemas fueron productivos porque se concentraron en sistemas que fueran útiles en vez de crear ‘máquinas pensantes’. Sin embargo, este interés no se concentró en los STIs.

En forma simultánea, la psicología educativa estaba experimentando un cambio de paradigma desde el conductismo hacia el constructivismo y aprendizaje situado socialmente. Esta revolución impulsó a muchos educadores a cuestionarse la práctica educativa de la época. La tecnología STI, mucha de la cual estaba fundamentada en el conductismo, perdió vigencia.

Los STIs parecían condenados a figurar como un mero comentario en la historia tanto de la ciencia de la computación como de la psicología educativa. Sin embargo, la perspectiva de una rápida expansión del poder de las computadoras no sólo en el procesamiento de la información sino también en la comunicación del conocimiento se presentaba como demasiado atractiva para abandonar la investigación en STIs. La combinación del trabajo de Wenger unida a la perspectiva sugerida por Self permitió el desarrollo de teorías interdisciplinarias necesarias sobre las cuales podía re-crearse una nueva generación de

STIs. El entendimiento sobre ambientes de comunicación resultó igualmente provechoso para los investigadores en STIs. El objetivo perseguido era concretar teorías y tecnologías necesarias para hacer posible el sueño de sistemas inteligentes de comunicación de conocimiento.

Actualmente, la investigación en STIs busca diseñar sistemas de aprendizaje fundamentados en principios propios y en nuevas teorías de aprendizaje así como su implementación con herramientas actuales. La evolución del área ha ido desde diseños generales hacia aplicaciones específicas con nuevas tecnologías (especialmente agentes) y nuevos estilos de sistemas (ambientes y sistemas colaborativos). Este área de investigación se reconoce como Inteligencia Artificial en Educación (IAED).

Los Sistemas Tutoriales Inteligentes han sido parte importante de la IAED durante los últimos 20 años. Estos sistemas aplican los principales tópicos de la Inteligencia Artificial como representación del conocimiento, razonamiento, explicación, Aprendizaje Automático (Machine Learning), lenguaje natural y planificación (planning) en su desarrollo. Además, los STIs conforman un interesante campo de experimentación para formalizar teorías cognitivas y su operacionalización.

Desde el punto de vista de la ingeniería del conocimiento, los STIs pertenecen a la categoría de sistemas basados en conocimiento complejo. En consecuencia comparten todas las características de tales sistemas, es decir, su rendimiento recae principalmente en la calidad, cantidad y aptitud de las diferentes clases de conocimiento puestos en el sistema.

La Instrucción Asistida por Computadora (CAI, Computer-Assisted Instruction) es un antecesor de los STIs. CAI es un sistema de instrucción para ser entregada vía computadora y posee dos características que le son propias: el estudiante es central al sistema y la computadora es sólo un vehículo para la instrucción, no un método de enseñanza. En cambio, el objetivo de un STI es dar a la computadora la capacidad de involucrarse más firmemente en el proceso de enseñanza a través de técnicas de Inteligencia Artificial. El centro es todavía el estudiante, pero un STI transforma a la computadora en un instructor dinámico, en vez de un vehículo estático de información.

Para alcanzar este nuevo y superior objetivo, un STI usa tres tipos de conocimiento: conocimiento del *dominio* (constituído por los tópicos de una materia en particular, como por ejemplo cálculos de porcentajes) conocimiento de *estrategias y métodos de enseñanza* (como por ejemplo el método de guía particular o coaching), y conocimiento sobre el *estudiante* (preferencias y estilos personales). Estos tres tipos de conocimiento proveen al sistema la capacidad de asistir al aprendiz. En particular se enfatiza un acercamiento a una enseñanza individualizada como la lograda por un buen maestro humano

quien posee la habilidad única de adaptar una lección a un estudiante. Un STI integra los tres tipos de conocimiento con información sobre la interactividad estudiante/sistema. Esta información es el conocimiento más relevante que se almacena y procesa en un modelo que representa al estudiante.

Un análisis completo de los STIs es un objetivo difícil de alcanzar. En la literatura se encuentran varias clasificaciones según diferentes puntos de vista, como por ejemplo:

- El nivel de inteligencia global (considerando la inteligencia de sus componentes)
- El algoritmo subyacente que motoriza el comportamiento del STI. Por ejemplo, una categoría importante de esta clasificación son los tutores modelTracing (registra el progreso del estudiante y almacena los caminos que siguió el estudiante para alcanzar la solución).
- Independencia (o no) del dominio. Desde afuera de un STI, puede decirse que este será más inteligente si es independiente del dominio a enseñar (en otras palabras, el modelo de enseñanza podría ser re-usado con distintos dominios). Pero muchos expertos sostienen que el conocimiento pedagógico es fundamentalmente dependiente del dominio.
- El grado de libertad que posee el estudiante: existe un continuo donde en uno de sus extremos se encuentra el aprendizaje por descubrimiento y en el otro el aprendizaje totalmente guiado.

Como se aprecia, existen diversos ejes de exploración del área en análisis y, además, varias de estas visiones se solapan entre sí. A fin de presentar en forma ordenada los conceptos y metodologías centrales involucrados en los Sistemas Tutoriales Inteligentes, se ha organizado la exposición en cuatro capítulos. El primero de ellos presenta el área de Inteligencia Artificial en Educación así como sus principales paradigmas. En el capítulo 2 se expande la descripción de los Sistemas Tutoriales Inteligentes mediante un análisis de su arquitectura básica. El capítulo 3 muestra el marco conceptual teórico de estos sistemas describiendo teorías de aprendizaje e instruccionales apropiadas para los STIs. Para apreciar las posibilidades prácticas de estos sistemas, en el capítulo 4 se describen las modernas tecnologías de diseño. Finalmente, en la Conclusión, se analizan los logros alcanzados y los problemas actuales que presentan los STIs. Asimismo, se presenta la posición personal y planteo de las líneas futuras de estudio de la autora de este Trabajo Final.

Índice general

Objetivo	I
Prefacio	II
1. Inteligencia Artificial en Educación	1
1.1. Inteligencia Artificial	1
1.2. Inteligencia Artificial en Educación (IAED)	2
1.3. Paradigmas de IAED	3
1.3.1. ICAI (Intelligent-Computer-Aided-Instruction)	3
1.3.2. ITS (Intelligent Tutoring System) o IIS (Intelligent Instructional System)	4
1.3.3. ILE (Interactive Learning Environment)	4
1.3.4. CSCL (Computer Supported Collaborative Learning)	5
1.3.5. LCS (Learning Companion System)	6
2. Sistemas Tutoriales Inteligentes	8
2.1. Introducción	8
2.2. Arquitectura tradicional	10
2.2.1. Modelo experto (o dominio)	12
2.2.2. Modelo del Estudiante	14
2.2.3. Modelo de enseñanza	23
2.3. Conclusión del capítulo	25
3. Contexto teórico del diseño de STIs	27
3.1. Introducción	27
3.2. Teorías de aprendizaje	28
3.2.1. Teoría de J. Anderson	28
3.2.2. Teoría de J. Self	29
3.2.3. Teoría constructivista de Piaget	31
3.3. Teorías Instruccionales	32
3.3.1. Teoría de la elaboración	33

3.3.2. Taxonomía de B. Bloom	35
3.3.3. La teoría de las condiciones del aprendizaje de Robert Gagné . . .	35
3.4. Arquitectura de la nueva trinidad a la luz del constructivismo	37
3.5. Conclusión del capítulo	41
4. Contexto práctico del diseño de STIs	43
4.1. Tecnologías para el diseño	43
4.1.1. Diseño basado en agentes	44
4.1.2. Diseño basado en técnicas de Aprendizaje Automático	46
4.1.3. Razonamiento basado en casos (RBC)	47
4.1.4. Sistemas Educativos Inteligentes y Adaptativos (SEIA) basados en la Web	50
4.2. Conclusión del capítulo	52
5. Conclusión	55

Capítulo 1

Inteligencia Artificial en Educación

1.1. Inteligencia Artificial

La diferencia clave entre Inteligencia Artificial y otras formas de programación es que IA responde inteligentemente a situaciones que no fueron específicamente anticipadas por el programador [73]. En la programación convencional, el programador prevé la solución de los problemas especificando todos los pasos que lo llevan a resultados exitosos. En la programación IA, el programador provee los medios para que la computadora resuelva los problemas cuando se presenten. Por ejemplo, un programa que traduce entre lenguajes no puede ser escrito anticipando todas las posibles oraciones y su traducción, ni listar todas las palabras y su traducción y combinarlas de manera simple. El programa debería ‘razonar’ acerca del significado de las oraciones, lo cual implica la posesión de conocimiento sobre los dos lenguajes, sobre el contenido de las oraciones y sobre el mundo de tal manera de poder resolver las ambigüedades.

IA es una disciplina teórica y aplicada. Ha brindado soluciones a una amplia gama de problemas que van desde el trabajo con robots autónomos hasta el diagnóstico de enfermedades; todas ellas pueden ser descritas como inteligentes: algunas demandan conocimiento especializado y otras conocimiento de todos los días que usamos sin tener en cuenta cuan complejo es.

IA no es una ciencia que estudia objetos del mundo natural: estudia objetos que los programadores de IA crean [73]. Para que estas creaciones sean entendidas y analizadas, su diseño debe estar basado en principios que soporten algún tipo de análisis riguroso. Por ejemplo, un programa de diagnóstico médico, debe estar basado en una teoría computacional de diagnóstico. Por una ‘teoría computacional’ se entiende una que acepte un análisis matemático convencional y que sea orientada hacia su implementación como un programa de computadora. Es posible verificar resultados prácticos, por ejemplo,

bajo ciertas condiciones, un diagnóstico será posible en un tiempo limitado. También es posible llevar a cabo experimentos con el programa, por ejemplo, medir como trabaja bajo diferentes condiciones. La teoría se desarrolla coordinando estudios matemáticos y empíricos. De esta manera, IA conduce al desarrollo de nuevas teorías, porque si una teoría que ya existe es adecuada, presumiblemente, podría ser programada de alguna manera.

1.2. Inteligencia Artificial en Educación (IAED)

Mientras la IA se desarrolla exitosamente en diversos campos, la educación ha tenido y tiene varios problemas: disminución del respaldo público y gubernamental, presión para proveer más entrenamiento vocacional, demanda para medir el rendimiento de los maestros, dificultad en obtener cuerpos docentes bien formados y el desafío del avance tecnológico. El campo IAED comprende aplicaciones de las técnicas de IA a problemas educativos. Se busca desarrollar sistemas prácticamente útiles y teóricamente bien fundamentados (sobre todo en la IA).

IAED tiene una historia corta y un poco accidentada. Las exploraciones iniciales (en la década de los 70s) fueron marcadas por el entusiasmo optimista original propio de la IA en general. El trabajo temprano hizo contribuciones significativas tanto a la Educación como a la IA. Por los 80s, el éxito de los sistemas expertos buscó capitalizar la investigación en IA a fin desarrollar sistemas IA-ED que fueran prácticamente útiles en vez de teóricamente interesantes. Es notable que pocos pioneros del área expresaran confianza en los desarrollos prácticos. Inevitablemente, el resultado percibido como fallas de los proyectos solamente confirmó que el aprendizaje y la enseñanza son procesos intrínsecamente difíciles. Además, son temas controversiales con argumentos abiertos. Cualquier proyecto particular en IA-ED debe adscribir a un punto de vista si hace algún progreso en la implementación de un sistema útil, el cual será criticado por aquellos con enfoques diferentes.

IAED es interesante debido a esta constante interacción de ideas y es importante debido a la contribución potencial al objetivo central de la sociedad sobre la mejora de la calidad de la enseñanza.

IAED es un campo de investigación derivado e innovador. En primer lugar, los investigadores del área han tomado o adoptado teorías y metodologías de disciplinas asociadas tales como la psicología y ciencias de la computación. Esta interdisciplinariedad da fortaleza al campo de IAED, pero también trae problemas que se reflejan en perspectivas conflictivas sobre diversos aspectos del área. Por otro lado, IAED es innovadora porque está contribuyendo con nuevas ideas a las disciplinas asociadas y generando sus propias

cuestiones las cuales provocan un marcado avance en el área.

Las tres áreas de mayor interés en IAED son las siguientes:

- Inteligencia Artificial (IA)
- Diseño instruccional (DI)
- Ciencia del aprendizaje (CA)

El diseño instruccional (DI) estudia teorías y métodos sobre estrategias instruccionales así como el proceso de desarrollo e implementación de dichas estrategias, esto es, investiga especificaciones detalladas para el diseño, desarrollo, implementación, evaluación y mantenimiento de situaciones que faciliten el aprendizaje de tópicos de determinadas materias en diversos niveles de complejidad.

Algunas tecnologías de IA y DI han sido introducidas exitosamente en la comunidad IAED produciendo buenos resultados en la construcción de varios Sistemas Instruccionales Inteligentes SII (IIS Intelligent Instructional System). Sin embargo, la interacción entre las comunidades IAED - IA, e IAED - CDI no ha sido muy activa.

Adicionalmente, tecnologías computacionales innovadoras como hipermedia, realidad virtual e internet han afectado significativamente a la comunidad de IAED, en general. Sin embargo, todavía es necesario hallar las direcciones más promisorias hacia donde focalizar el esfuerzo a fin de mejorar el progreso de IAED.

1.3. Paradigmas de IAED

Los esfuerzos de la Inteligencia Artificial aplicada a Educación se concentran en el desarrollo de sistemas que se adapten a las diferencias individuales de los estudiantes para lo cual requieren de tecnologías computacionales avanzadas. A continuación se exponen brevemente las características principales de los paradigmas de IAED a fin de situar los STIs en esta taxonomía.

1.3.1. ICAI (Intelligent-Computer-Aided-Instruction)

CAI (Computer-Assisted Instruction o Instrucción Asistida por Computadora) es un paradigma que no distingue entre la representación separada de los conceptos a enseñar de cómo enseñarlos. CAI parece inteligente porque se adapta a los errores cometidos por el estudiante, pero esto es sólo el resultado del diseño que trata de anticiparse a todas las posibles equivocaciones; el sistema no puede proveer una explicación a tales errores. CAI evolucionó hacia los STIs.

A diferencia de CAI, los sistemas ‘inteligentes’ se diseñan para representar tanto los conceptos a ser enseñados como así también la manera en que un estudiante puede aprenderlos. Un ICAI se construye con componentes separados: el proceso de enseñanza está separado del dominio y ambos de las estrategias tutoriales. El nivel de detalle con el cual se definan los elementos que componen este tipo de sistema (granularidad) determinará la manera en que se llevará a cabo la observación del progreso del estudiante.

1.3.2. ITS (Intelligent Tutoring System) o IIS (Intelligent Instructional System)

La asignación de un significado preciso al término STI (o ITS en inglés) no es una tarea fácil. El análisis de los STIs existentes nos lleva a estudiar gran variedad de programas ‘inteligentes’ con ambiciones educativas.

Un STI tradicional posee cuatro componentes: el modelo del dominio, el Modelo del Estudiante, el modelo pedagógico y el ambiente de aprendizaje. Los proyectos STIs varían según el nivel de inteligencia de sus componentes. Por ejemplo, si un sistema tiene un modelo del dominio con un alto nivel de inteligencia, entonces es posible construir soluciones a problemas complejos, pero tal vez contenga métodos de enseñanza simples, desmejorando el resultado final. O, contrariamente, un STI que posea estrategias tutoriales ‘más inteligentes’ puede contener una representación menos sofisticada del dominio a enseñar.

Este paradigma será desarrollado en el capítulo 2 de esta tesis.

1.3.3. ILE (Interactive Learning Environment)

Uno de los métodos de aprendizaje más estudiado es el basado en preguntas (inquiry-based), aunque también se lo describe de variadas formas: centrado-en-el-estudiante, constructorista [65] [66], constructivista [26], y basado en descubrimiento [7, 13]. Los sistemas que implementan este tipo de aprendizaje son estructuralmente y conceptualmente mucho más diversos que los STI previos. Por falta de un mejor nombre, se los conoce como ILEs (Interactive Learning Environments). Estos sistemas comparten varios principios que contrastan con la visión implícita de los STIs. Los principios que sostienen a los ILEs, presentados someramente, incluyen:

- *Construcción, no instrucción*: los estudiantes aprenden más efectivamente mediante la construcción de su propio conocimiento, no solo a través de la lectura, ni con instrucción-y-práctica (drill-and-practice) organizada.
- *Control del estudiante, no control del tutor*: el tutor se ve como un guía; el estudiante tiene un papel significativo.

- *La individualización es determinada por el estudiante no por el tutor:* los ILEs concuerdan con los STIs tanto en la realimentación individualizada como en la información clave para el aprendizaje. Sin embargo, ambos difieren en el origen de la información personalizada.

Mientras el tutor es responsable de la personalización en la realimentación en los STIs, en los ILEs los estudiantes generalmente reciben realimentación única y la información como una función de su interacción con el ambiente construido en el sistema. Aunque, los mecanismos precisos difieren sustancialmente entre ILEs, la individualización está, al menos, bajo control parcial de los estudiantes. La realimentación más rica es generada por la interacción del estudiante con el ambiente de aprendizaje, no con el tutor. La información más rica debe darse como función de las elecciones del estudiante y acciones en el ambiente de aprendizaje, en vez del discurso generado por el tutor.

Los principios que sostienen al método de enseñanza/aprendizaje basado en preguntas guía el diseño para sistemas de aprendizaje que difieren sustancialmente de un STI. Dicho muy ampliamente, la inteligencia de un ILE está distribuida en las herramientas que lo componen, en lugar de estar centralizada. Estas herramientas computacionales pueden incluir video interactivo u otra representación gráfica, y permiten a los estudiantes investigar y aprender tópicos libres de control externo. Mientras esta libertad se deriva desde argumentos fundamentados sobre como ocurre el aprendizaje efectivo, también tiene beneficios prácticos. Los ILEs no son intensivos en conocimiento como los STIs. Se benefician de alguna representación explícita de tópicos que investiga el aprendiz, pero no son omniscientes, ‘no saben todas las respuestas correctas’. Además, como muchos ILEs no buscan enseñar, son libres de obligaciones de trabajar con un Modelo del Estudiante y tomar decisiones pedagógicas más complejas. Por otro lado, los ILEs tienen sus propios desafíos. Si a un estudiante se le da una ‘herramienta’ que magnifica su habilidad para descubrir ideas interesantes, ¿qué lo previene de usar este poder para equivocarse en un vasto mar de temas sin interés? ¿Cómo sabrán que clase de conocimiento construir? ¿Cómo juzgarán tales construcciones?

1.3.4. CSCL (Computer Supported Collaborative Learning)

El aprendizaje colaborativo se refiere a estudiantes trabajando en grupos. Estos ambientes demostraron ser provechosos tanto cognitivamente como socialmente [84]. En esta situación el centro de las interacciones no se encuentra entre maestro y estudiante, sino entre estudiantes. En este Trabajo Final se entiende por aprendizaje colaborativo al que se da entre estudiantes trabajando juntos con la ayuda de un STI vía una red de computadoras.

Un aspecto importante de los ambientes colaborativos es la situación que se presen-

ta en los grupos ya que no todos los estudiantes poseen las mismas habilidades. Esto crea dos problemas. El primero de ellos es la valoración de la actividad del alumno, tema que afecta la actualización del Modelo del Estudiante. ¿Debe darse la aprobación a respuestas correctas al primer estudiante que alcanzó o a todos los integrantes del grupo? El segundo problema se presenta en la toma de decisiones pedagógicas sobre cómo avanza el grupo en el material instruccional. ¿Un sólo alumno debe marcar la marcha? Y si es así, ¿cuál? La investigación en aprendizaje colaborativo analiza las interacciones entre estudiantes durante el trabajo entre pares a fin de analizar e identificar las ventajas cognitivas de la actividad conjunta [28] [29].

El beneficio dado por el aprendizaje colaborativo se manifiesta en el proceso de articulación y co-construcción de ideas que ocurren cuando los pares trabajan cercanamente. Los participantes en una situación de resolución de problemas tienen que explicitar sus ideas (afirmaciones, hipótesis, negaciones, etc.) a los otros colaboradores, los desacuerdos conducen a negociaciones y justificaciones, ayudando a los estudiantes a converger hacia el objetivo común del entendimiento compartido.

La computadora provee la oportunidad de soportar y mejorar este enfoque de varias maneras. Por ejemplo, ofreciendo espacios de problemas basados en computadora para conjuntamente crear y explotar estructuras de conocimiento común y referencias compartidas. Además, las redes hacen posible abrir la estructura colaborativa a comunidades distribuidas proveyendo acceso remoto a estos espacios como también comunicación mediante computadora para el soporte del intercambio interpersonal y debate.

Los sistemas basados en este paradigma se los denomina Computer Supported Collaborative Learning (CSCL) [51] y poseen los siguientes principios:

- 1. Construcción conjunta de la solución de un problema
- 2. Coordinación de los miembros de un grupo para planificar las tareas
- 3. Semi-estructuración de los mecanismos de la interacción
- 4. Enfoque tanto al proceso de aprendizaje como al resultado del aprendizaje

Actualmente existe un número creciente de ambientes de aprendizaje colaborativo para grupos virtuales abiertos y cerrados para un rango de tareas de aprendizaje y se han publicado experiencias de su uso en escuelas y universidades [8] [52]

1.3.5. LCS (Learning Companion System)

Self, en 1988, sugirió que la computadora podría simular un co-aprendiz en vez de un tutor, para cooperar en lugar de enseñar. Actualmente un co-aprendiz es un agente que juega el papel de un compañero. En [18] los autores proponen los Sistemas de Compañero de Aprendizaje (Learning Companion Systems). Estos sistemas presentan una noción

más general de sistemas basados en agentes múltiples. Sugieren que la computadora puede simular dos agentes co-existentes: un maestro y un compañero de aprendizaje (co-aprendiz), los cuales tendrán varias interacciones con el estudiante real. La interacción entre las personas nos permite intercambiar ideas, entender las ideas de otros, evaluar nuevas ideas, ver problemas desde perspectivas diferentes y competir. En estos sistemas, por ejemplo, el compañero y el estudiante pueden colaborar con el tutor, competir entre ellos, o el compañero puede enseñar al estudiante.

El agente tutor enseña tanto al estudiante verdadero como al estudiante simulado. En este sistema los dos estudiantes aprenden el mismo material e interactúan uno con el otro.

El rol de este STI es proveer ejemplos, guía y comentarios sobre las soluciones sugeridas tanto por el estudiante real como por el compañero virtual. Durante el proceso de resolución de problemas el rendimiento del compañero virtual debe poseer el ritmo de los estudiantes reales.

La construcción del compañero se puede hacer de dos maneras diferentes: por simulación o por Aprendizaje Automático. En la simulación las habilidades se codifican directamente como parte del compañero. En Aprendizaje Automático, el compañero no posee conocimiento previo del dominio. Tiene que aprender con los mismos ejemplos que el estudiante real.

Integration-Kid fue el primer LCS [19] en el dominio del aprendizaje de la integración indefinida (tema del análisis matemático).

Capítulo 2

Sistemas Tutoriales Inteligentes

2.1. Introducción

El uso de las computadoras en educación data de mucho tiempo y uno de los primeros proyectos fue una consecuencia de la teoría del comportamiento de estímulo-respuesta: el aprendizaje no es independiente del estímulo recibido por el estudiante y la enseñanza es concebida como la organización de estímulos a fin de inducir en el estudiante un comportamiento deseable [78]. Los sistemas más primitivos estaban programados según esta concepción, esto es, los estudiantes eran expuestos a la misma secuencia fija de material de enseñanza independientemente de su desempeño.

Sistemas más sofisticados intentaron evitar esta linealidad introduciendo algunas capacidades adicionales, como por ejemplo, anticipar todas las posibles respuestas al estudiante tratando de elegir la correcta. Los problemas más importantes de ese momento eran:

1. la ausencia de individualización (todos los estudiantes reciben la misma secuencia de unidades)
2. los estudiantes eran considerados elementos pasivos en el proceso de aprendizaje (sólo responden preguntas propuestas por el sistema)

Muchos sistemas fueron desarrollados desde que se comenzó a investigar sobre STIs y sentaron las bases conceptuales de los sistemas desarrollados actualmente. Puede fijarse al año 1990 como el momento en que se produjeron importantes cambios. Varios STIs ‘ancianos’ (pre-1990) fueron perfilando los fundamentos de los nuevos sistemas, donde principalmente se busca la enseñanza adaptativa.

Entre los años 1993 y 1995 Shute realizó un estudio para intentar buscar un consenso sobre el significado de inteligencia en STI entre los investigadores más reconocidos en el campo. Para ello les pidió que resumieran en dos o tres líneas sus ideas acerca de

qué significaba la 'I' en el acrónimo STI. Las respuestas fueron variadas e interesantes (se pueden encontrar en [76]), y la conclusión casi unánime fue, en palabras de Shute:

Como vemos en esta muestra no aleatoria de respuestas sobre lo que es la inteligencia en un STI, 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 'I' de STI, nuestra postura es que los dos componentes, (diagnóstico y asistencia), trabajando conjuntamente, constituyen la inteligencia de un STI.

El objetivo de un STI es dar a la computadora la capacidad de involucrarse en el proceso de enseñanza usando técnicas de Inteligencia Artificial. El centro es el estudiante, donde la computadora se transforma en un instructor dinámico en vez de un vehículo estático de información. El conocimiento que posea un STI le permite proveer asistencia a un aprendiz de forma parecida a un instructor humano. Un buen instructor tiene la habilidad de adaptar una lección a un estudiante individual a través de un proceso de enseñanza; por lo tanto, el objetivo de un STI debe ser alcanzar este nivel de instrucción individualizada.

Para alcanzar este objetivo, un STI tiene que reflejar el progreso de un estudiante durante una sesión de tutoría. El conocimiento de este progreso es representado usando un Modelo del Estudiante. Este modelado puede ser definido como el proceso de recolección de información relevante (habilidades de aprendizaje, sus fortalezas y debilidades, etc.) a fin de identificar y representar el estado del conocimiento de un estudiante. Este modelo debe poseer la habilidad de proveer al estudiante real con la ayuda individualizada que este requiera.

Como la enseñanza es un proceso dinámico, un STI debe actualizar el Modelo del Estudiante a medida que aprenda acerca de un estudiante individual. Tal mantenimiento puede alcanzarse representando al estudiante de manera tal que sea fácilmente entendido y modificado por el STI.

En 1970 se desarrolló el primer STI en el Instituto Tecnológico de Massachussets (MIT): SCHOLAR[17]. Este sistema estuvo basado en una aplicación conocida como ELIZA diseñada para el estudio de la comunicación entre persona-máquina. SCHOLAR se diseñó para enseñar geografía del Continente Americano y supuso un salto cualitativo sobre los sistemas educativos anteriores. Previamente a 1970, el proceso de enseñanza se

basaba en series de preguntas y respuestas pre-diseñadas que hacían el sistema rígido, mientras que en SCHOLAR el formato de representación de la información permitía al sistema anticipar las respuestas de la persona que aprende y adaptarse a sus necesidades.

2.2. Arquitectura tradicional

Los sistemas de aprendizaje asistido por computadora tradicionales (como CAIs) fueron desarrollados para entregar instrucción en un área particular, para luego evaluar al aprendiz que tomó la lección programada. Las respuestas a preguntas (usualmente mediante elección múltiple) se usan para dirigir el curso.

El material instruccional del curso se presenta al estudiante y luego se hacen varias preguntas. Si el usuario responde correctamente se pasa a la siguiente fase. Las respuestas a las preguntas son resumidas al final del curso para medir el rendimiento del estudiante. Si las respuestas son incorrectas, se le presenta el material instruccional nuevamente, de manera ligeramente diferente. Si luego el estudiante vuelve a responder equivocadamente entonces se presenta la instrucción reparadora. Tales sistemas pueden parecer inteligentes cuando se adaptan a las falencias del estudiante. Sin embargo, esta apariencia es el resultado del esfuerzo del diseñador tratando de anticipar todos los posibles errores que puede cometer el estudiante. Esto se construye en la etapa de diseño y se codifica en estructuras ramificadas.

Pero, si el diseñador no anticipa interpretaciones incorrectas del material instruccional por parte del estudiante, el sistema nunca podrá dar una explicación que ayude a resolver el malentendido. Estos sistemas son incapaces de generar dinámicamente una respuesta a una situación particular como lo haría un tutor humano.

Reconociendo las deficiencias de los CAIs tradicionales, los Sistemas Tutoriales Inteligentes fueron, subsecuentemente, desarrollados como un intento de adaptar la velocidad y nivel de presentación como lo requiere un estudiante real. La arquitectura de los primeros STIs contenían los siguientes componentes:

- Experto en el tema (o tarea) a enseñar
- Modelo de aquello que el estudiante ha entendido (o posiblemente malentendido)
- Experto pedagógico (como enseñar)

En [14] se identifican estas áreas como componentes de un STI y las denomina módulo experto, módulo de instrucción y currículo, y módulo de diagnóstico del estudiante. En 1988 Anderson [3] identificó, de manera similar, áreas como la representación del

conocimiento y metodologías de enseñanza. Estos tres componentes de un STI son referidos a menudo como la ‘Trinidad tradicional’ de un Sistema Tutorial Inteligente y serán presentados con la denominación más usual:

- **Modelo experto (o dominio):** es el conocimiento del dominio(o tópicos del tema que será enseñado) incrustado en el sistema y representa el conocimiento del experto en el tema y las características de la resolución de problemas asociados
- **Modelo del estudiante:** su función es capturar el entendimiento del aprendizaje sobre el dominio
- **Modelo tutorial:** contiene las estrategias tutoriales e instrucciones indispensables. Estas estrategias deben ajustarse a las necesidades del estudiante sin la intervención del tutor humano. El propósito principal de este módulo es reducir la diferencia del conocimiento entre el experto y el estudiante al mínimo (o a ninguna)

La arquitectura básica (ver figura 2.1) de un STI consta de los tres modelos vistos mas un módulo del entorno donde se encuentra la interfaz de comunicación , el estudiante y el profesor.

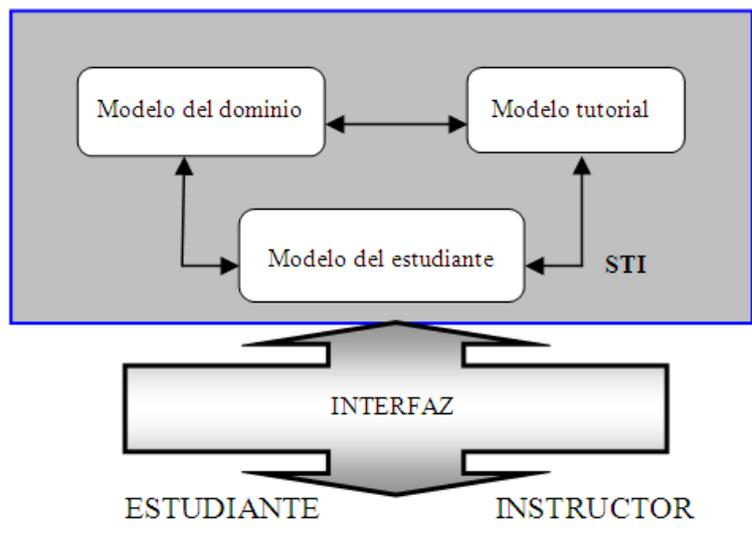


Figura 2.1: Arquitectura básica de un STI

La arquitectura básica de un STI consiste en un módulo experto, un módulo del alumno y un módulo instructor, que operan de forma interactiva y se comunican a través de un módulo central que se suele denominar motor y es el que ejecuta el plan instruccional. El módulo experto contiene el conocimiento acerca de la materia que se pretende enseñar, el módulo del alumno guarda toda la información relativa al mismo que se genera durante la interacción con el sistema, y el módulo instructor controla los

planes y decisiones pedagógicas. Finalmente, el módulo entorno gestiona la interacción de los otros componentes del sistema y controla la interfaz hombre/máquina.

2.2.1. Modelo experto (o dominio)

Los primeros STIs incorporaron un sistema experto (ver figura 2.2) para representar el dominio y varios de ellos fueron contruidos sobre sistemas expertos pre-existentes. Un sistema experto busca dar soluciones como un experto humano a problemas en un dominio específico. Un sistema experto trata con información imprecisa e incompleta y debe explicar sus decisiones y poseer razonamiento similar al experto humano. A continuación se presenta la arquitectura simplificada de un sistema experto:

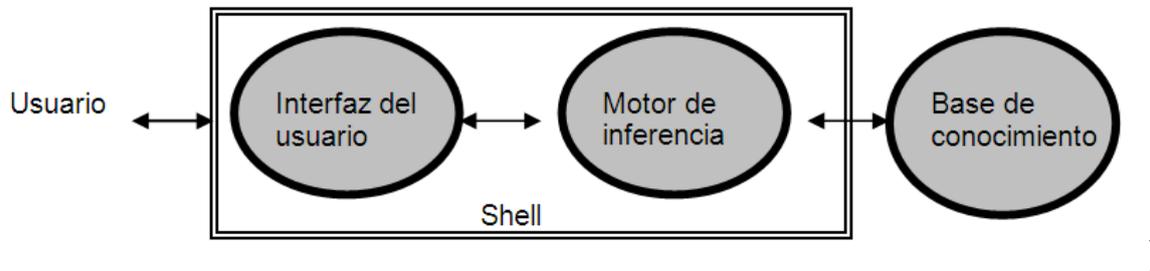


Figura 2.2: Una arquitectura simple de un sistema experto

Un Sistema Experto posee tres módulos:

- la interfaz con el usuario permite establecer comunicación entre usuario y sistema
- el motor de inferencia es un intérprete para la base de conocimiento produciendo resultados y explicaciones para los problemas a resolver
- el corazón del sistema experto es la base de conocimiento la cual contiene la información para resolver problemas de una aplicación particular

Existen varias estrategias para representar el conocimiento en las bases de conocimiento como las reglas if-then y reglas if-then con medidas de incertidumbre.

Las reglas if-then, llamadas también reglas de producción, han sido las más ampliamente usadas para representar el conocimiento del dominio. Una regla de producción típica tiene la forma:

IF condición(es) THEN conclusión(es)

Hay dos estrategias para el razonamiento con estas reglas, ejecutado por el motor de inferencia:

- la entrada o datos del problema se hace corresponder con las condiciones de alguna regla, si esto ocurre, su conclusión se transforma en entrada a otra regla; si una regla no lo hace, busca reglas subsecuentes a fin de hallar el objetivo
- se parte de la regla cuya conclusión se ajusta al objetivo; la condición correspondiente se transforma en objetivos intermedios los cuales deben ser satisfechos y se accede a otras reglas a fin de alcanzar las condiciones conocidas del problema

En la práctica, por ejemplo MYCIN [75], se aplicaron ambas estrategias.

También, existe la posibilidad de asociar a cada regla un nivel de confianza dando lugar a las reglas if-then con incertidumbre. Por ejemplo, en el dominio de las Ecuaciones Diferenciales Ordinarias, el método numérico de Runge Kutta puede formar parte de la siguiente regla de producción:

```
IF Tema=EcuacDiferenciales THEN Presentar=MetodoRungeKutta
con un 95% de certeza
```

Un ejemplo clásico de este tipo de sistemas podemos verlo en LISP TUTOR [2]. Es un STI desarrollado para enseñar los principios básicos de la programación en LISP. El modelo experto fue creado como una serie de reglas de producción para crear programas LISP y un Modelo del Estudiante que se construyó como un subconjunto de estas reglas de producción correctas junto con reglas incorrectas.

LISP TUTOR se basa en el principio ‘aprender haciendo (learning by doing)’ donde el estudiante descubre las producciones mientras trabaja con el problema. El tutor actúa como un guía pero nunca establece que producciones aprender. Anderson y su equipo usaron GRAPES (Goal Restricted Production System Architecture) para representar el conocimiento de LISP TUTOR con aproximadamente 325 reglas de producción. El sistema también usa 425 reglas para la representación de las equivocaciones que pueden cometer los programadores novicios.

Los mapas conceptuales son otro modo de representar el dominio. De acuerdo a la teoría de Ausubel [7], el pensamiento humano involucra no sólo el entendimiento de conceptos sino también las relaciones entre éstos. El aprendizaje significativo se refiere a la incorporación de nuevas ideas o conceptos a los ya adquiridos en forma no arbitraria [60], el estudiante relaciona nuevos conceptos más específicos con otros más generales conocidos anteriormente. A partir de esta teoría se ha creado una metodología denominada mapas

conceptuales. Estos mapas representan relaciones significativas entre conceptos en forma de proposiciones y gráficamente son una red de nodos (conceptos) y arcos dirigidos (relaciones). Los mapas conceptuales pueden ser utilizados como un modelo semántico para estructurar el material instruccional [46]. Son útiles para capturar los puntos claves en el desarrollo del material de enseñanza, ayudar al profesor en el análisis de los contenidos del curso para lograr una mejor estructura del conocimiento y de las habilidades, a determinar una secuencia instruccional de conceptos elementales, a mejorar el proceso de evaluación y la construcción de pruebas más apropiados [58]. Murray, Reigeluth y Zabala [59, 71, 87] coinciden en clasificar los conocimientos curriculares en hechos, conceptos, principios y procedimientos. Sin embargo no hay consenso sobre el tipo de vínculos entre conocimientos pues éstos dependen de la forma en que cada sistema los aplique.

Dentro de un mapa conceptual, un nodo es definido con un nombre que relacionará el tema con su material de enseñanza. Cada nodo tiene tres características asociadas: tipo, importancia y dificultad. El tipo permite clasificar al contenido en hechos, conceptos o procedimientos. La importancia establece una jerarquía en cuanto al interés que el sistema debe otorgar al tema en el momento de la instrucción. Los niveles de dificultad que presenta el aprendizaje del tópico son usados por las meta-estrategias tutoriales en el plan instruccional. Las relaciones (arcos entre nodos) tienen un significado específico, una dirección, y pueden unir uno o más nodos.

Adhiriendo a la postura de Murray, se ha desarrollado una representación de dominio usando un mapa conceptual [43, 6], sobre la temática de Métodos Numéricos para resolver Ecuaciones Diferenciales Ordinarias. A cada nodo se le asoció objetivos instruccionales específicos. En la figura 2.3 se aprecia la estructura del dominio modelado en un mapa conceptual.

2.2.2. Modelo del Estudiante

Una característica distintiva de los sistemas de aprendizaje basados en Inteligencia Artificial en computadora fue la de focalizar el modelado y adaptación de los estudiantes como individuos [82]. AIED fue la primera en reconocer la necesidad de investigar un área ahora conocida como modelado del usuario, conocido así y usado ampliamente desde 1980. Un Modelo del Estudiante permite a un sistema atender los intereses de un estudiante: tópicos conocidos, equivocaciones cometidas, deseos, etc. La adaptación se puede realizar en varios niveles:

- presentación del material y ayudas
- dificultad de los problemas propuestos
- selección de la estrategia instructora más adecuada según sus capacidades

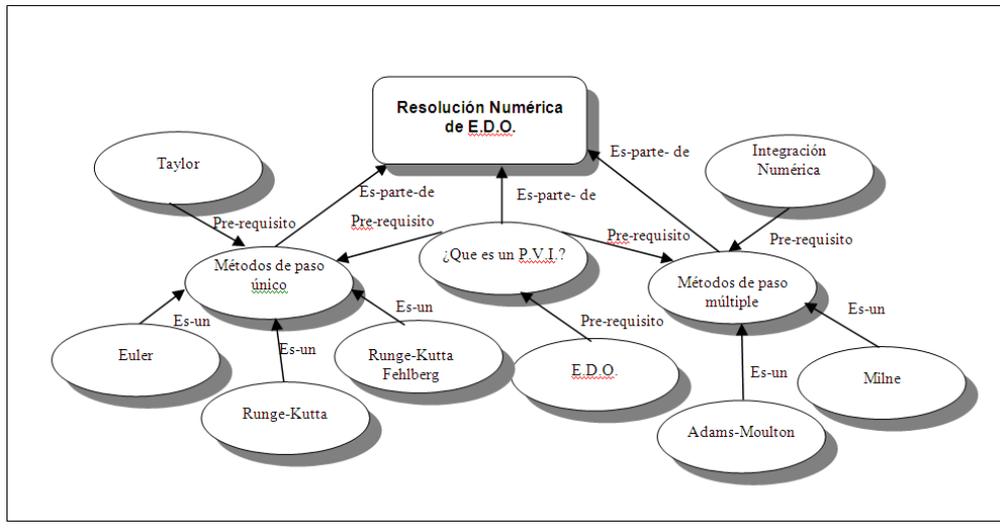


Figura 2.3: Ejemplo de estructura de dominio para Ecuaciones Diferenciales Ordinarias

- habilidades y estilos de aprendizaje preferidos

La importancia de adaptar la enseñanza a cada alumno es analizada en estudios como el que aparece en [10], que avalan la instrucción individualizada como la forma más efectiva de aprendizaje. En este trabajo, Bloom concluyó que cuando se usan los métodos convencionales de enseñanza (un profesor para treinta alumnos, con exámenes periódicos) las calificaciones obtenidas por los alumnos tienen una distribución normal, con media entre 50 y 60 % pero con una desviación típica grande. Si el profesor adapta sus lecciones para intentar evitar los errores que sus alumnos cometen en los exámenes, las medias se mueven hasta un 84 % y la desviación típica disminuye considerablemente. El cambio más dramático ocurre cuando los alumnos reciben enseñanza individualizada. La media llega a ser del 98 %, con una desviación típica que es la mitad de la desviación típica de los alumnos que recibieron una enseñanza convencional. Estos resultados constituyen un argumento más a favor del uso de los Sistemas Tutoriales Inteligentes, ya que demuestran los buenos resultados de la enseñanza individualizada.

En su exploración inicial, la investigación estableció una distinción entre modelos del estudiante descriptivo (o declarativo) o predictivo (o procedimental). Anteriormente el Modelo del Estudiante fue visto como una estructura de datos en la cual se representaban características de un estudiante y su rendimiento en la resolución de problemas. Sin embargo, debía ser visto como un programa describiendo los pasos que el estudiante sigue cuando resuelve problemas, y en este caso el modelo puede ser usado no solamente descriptivamente sino predictivamente a fin de adelantar cómo un estudiante podría re-

resolver un problema en el futuro (asumiendo que el modelo fuera exacto).

A fines de los 70s, los investigadores del área eran optimistas: se desarrollaron varios sistemas que demostraron su naturaleza distintiva; se estableció un programa concertado de investigación en el área, basado en la experimentación.

En 1977 J. Self consideró los posibles beneficios de los modelos de estudiante predictivos. Básicamente, él propuso optimizar el proceso de aprendizaje/enseñanza a través del análisis en vez de la experimentación. En la figura 2.4 se explica el concepto:

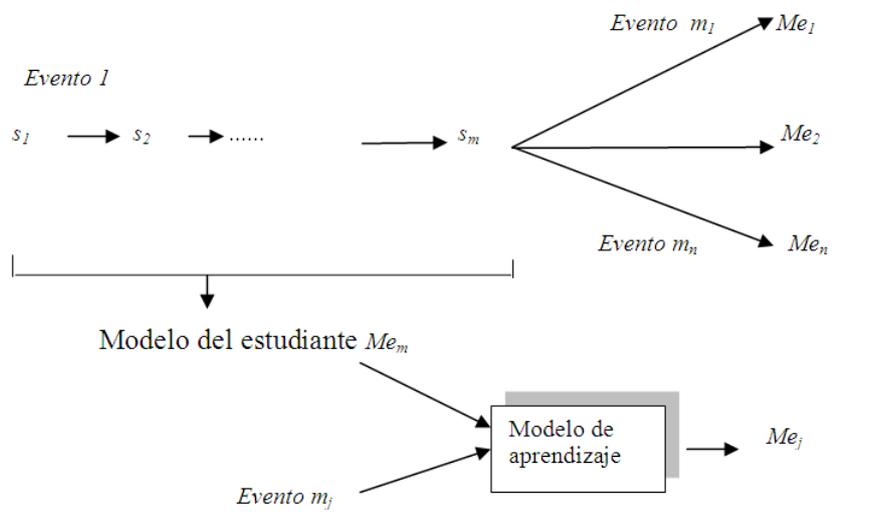


Figura 2.4: Modelado del estudiante a través de la adaptación

Cuando un estudiante real usa un sistema de aprendizaje basado en computadora pasa a través de una serie de situaciones s_1, s_2, \dots, s_m provocados por una secuencia de eventos.

En principio, se puede analizar la secuencia de situaciones y eventos que ya han ocurrido para determinar un nuevo modelo de estudiante M_e para una determinada situación s_m . En la situación s_m hay un conjunto de posibles eventos subsiguientes. Cuál de ellos se elige? Se podría tomar cada posible evento e_{mj} por vez, usando un modelo de aprendizaje, para determinar el modelo del estudiante M_{ej} como si tal evento hubiera ocurrido. Luego, se podría considerar cada Modelo del Estudiante M_{ej} resultante para encontrar cuál de ellos es el mejor. Posteriormente, el sistema podría adaptarse al evento que conduce a ese Modelo del Estudiante.

A principios de la década de los 80s, era muy arduo encontrar representaciones adecuadas para modelos del estudiante, dominio y estrategias tutoriales. Esta situación condujo a problemas prácticos importantes en la implementación de los STIs.

En 1985 los desarrolladores buscaban STIs que funcionaran como ‘colaboradores computacionales’ y que se comprometieran con el estudiante en el proceso de aprendizaje a través de un descubrimiento compartido. Por lo tanto, la base más desarrollada de la arquitectura tripartita fue el Modelo del Estudiante usándose técnicas de Aprendizaje Automático.

Otros autores definían al Modelo del Estudiante como una representación de ‘algunas características y actitudes de aprendices, que serán útiles para alcanzar una interacción adecuada e individualizada establecida entre el ambiente computacional y los estudiantes’ [42].

Actualmente se considera que un Modelo del Estudiante debe contener información tal como una descripción individualizada del conocimiento del estudiante, habilidades de aprendizaje, fortalezas y debilidades personales. Un STI usa el modelo para determinar si un estudiante ha cometido un error o ha adquirido un malentendido. Luego de determinar esta información, el STI puede provocar realimentación tutorial adecuada que ayude al estudiante a entender su error (el estudiante requiere realimentación detallada a fin de localizar el error y extraer inferencias generales a partir de dicho error).

2.2.2.1. Modelo conceptual del estudiante

Cuando se desea crear un modelo para representar y guiar a un estudiante, las cuestiones fundamentales a considerar son:

- **Selección de la estructura** que se usará para representar el modelo del alumno. Esta información puede almacenarse de muchas formas distintas: en un vector, en una red semántica, en una red bayesiana, en forma de afirmaciones, etc.
- **Inicialización del modelo del alumno.** La estructura elegida para representar el conocimiento del alumno debe inicializarse cuando se inicia la interacción con el sistema. Para ello existen varias opciones: utilizar la información disponible acerca del alumno, pedirle que se asigne en uno de los estereotipos de alumnos previamente definidos, realizar pruebas previas, etc.
- **Diagnóstico.** Una vez que el modelo del alumno se ha inicializado comienza la interacción con el sistema. El procedimiento de diagnóstico elegido actualizará el modelo del alumno tras sus interacciones con el sistema, utilizando dos fuentes de información principalmente:
 1. el modelo del alumno actual

2. su comportamiento en el proceso interactivo de enseñanza, el cual puede medirse usando distintas variables y que es preciso definir previamente (soluciones a problemas, respuestas a preguntas, tiempo empleado en lectura de pantallas, etc.).

También, el modelo debe tomar en consideración el dominio a ser enseñado y el proceso de aprendizaje del estudiante. Como la enseñanza es un proceso dinámico, un STI tiene que actualizar este modelo a medida que recibe información del estudiante real. A fin de mantenerlo, este debe ser representado de manera tal que sea entendible y fácilmente modificable por el sistema. Esta representación depende también del dominio instruccional que será enseñado.

En resumen, el modelo del alumno puede ser utilizado para los siguientes propósitos:

1. Determinar si el alumno está preparado para continuar con el siguiente tema del currículum, y elegir este tema
2. Generar explicaciones que el alumno pueda entender (nivel de detalle adecuado a sus conocimientos actuales)
3. Ofrecer consejos y ayudas sin que el estudiante lo solicite. En este sentido, es importante que el tutor no interrumpa a los alumnos con demasiada frecuencia, y que les permita aprender de sus errores
4. Generar problemas de nivel adecuado. La generación dinámica de problemas es un área que se apoya fuertemente en el modelo del alumno. Una vez identificados los puntos débiles del alumno, se genera un problema que el módulo experto resuelve paralelamente para ser capaz de diagnosticar la solución del alumno. Así, a cada alumno que interactúe con el sistema, se le presentará una colección diferente de problemas, adecuada a su nivel de conocimiento
5. Seleccionar la estrategia tutorial más apropiada dado el nivel de conocimiento actual

2.2.2.2. Tipos de Modelo del Estudiante

Los tipos del Modelo del Estudiante más comunes son el Modelo de superposición, Modelo diferencial, Modelo de perturbación y Modelo basado en traza. A continuación se presenta una breve explicación de cada uno de ellos y se ejemplifica su uso en STIs reales.

Modelo de superposición También se lo conoce como ‘overlay model’. En este enfoque se considera que el conocimiento del alumno es un subconjunto propio del conocimiento del experto. Este enfoque supone que todas las diferencias entre el comportamiento del alumno y el del experto se explican como una falta de conocimiento del

alumno. El modelo funciona bien cuando el principal objetivo del sistema instructor es transmitir el conocimiento experto al alumno. El mayor problema del modelo de superposición es que no considera el conocimiento del alumno que el experto no posee, y por tanto son incapaces de reaccionar ante su aparición. Esta carencia motivó la aparición de otros modelos.

Como se ve en la figura 2.5, el modelo asume que el conocimiento del estudiante es un subconjunto del conocimiento del experto y el objetivo del tutor es aumentar este subconjunto:

Este modelo asume que el estudiante no aprenderá nada que el experto no sepa. Es-

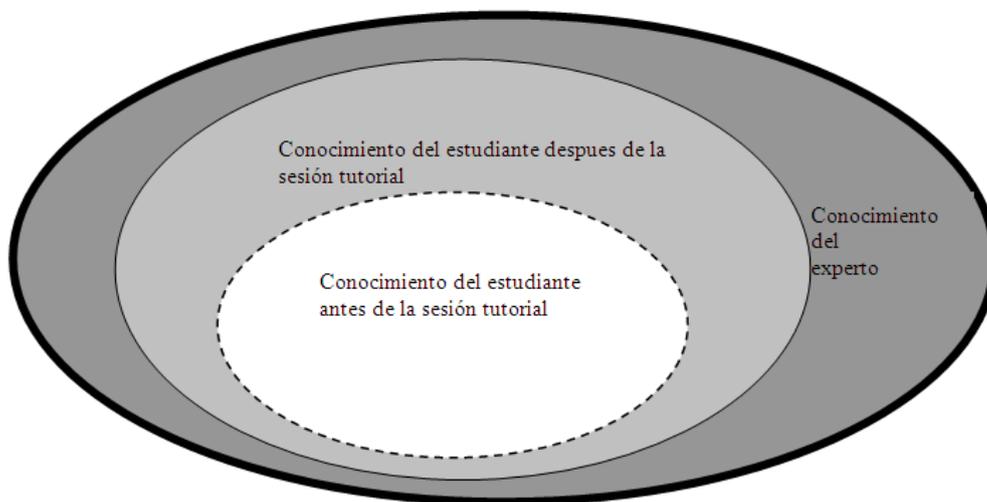


Figura 2.5: Representación del modelo overlay mostrando los efectos del proceso tutorial

pecíficamente no reparará errores o equivocaciones que el estudiante tenga o adquiera durante su aprendizaje. Tampoco hay un mecanismo para diferenciar entre conocimiento ya adquirido del conocimiento no expuesto todavía.

GUIDON [22] usa un modelo de superposición para determinar el diagnóstico de infecciones bacteriales. La base de conocimiento de MYCIN se usó como fundamento para este sistema tutorial.

En GUIDON un estudiante juega el rol de MYCIN haciendo preguntas y arribando a conclusiones. El STI controla las acciones del aprendiz guiándose por las respuestas dadas por MYCIN y de esta manera mantiene el Modelo del Estudiante. Sólo interviene si las acciones del estudiante no son óptimas o cuando este solicita ayuda. La estrategia

tutorial de GUIDON posee alrededor de 200 de reglas las cuales están codificadas independientemente del dominio de conocimiento de MYCIN.

GUIDON tiene problemas de implementación como estrategias tutoriales ineficientes y una muy limitada interacción, sin embargo tuvo un importante papel en el desarrollo de técnicas de Inteligencia Artificial en el aprendizaje basado en computadora .

Modelo diferencial Se presenta como una modificación del modelo de superposición. El modelo diferencial divide el conocimiento del alumno en dos categorías: conocimiento al que es expuesto el alumno para que sea aprendido y conocimiento que no puede esperarse que el alumno tenga (pertenece al experto). Así, a diferencia del modelo de superposición, el modelo diferencial reconoce y trata de representar explícitamente tanto el conocimiento del alumno como las diferencias alumno/experto. La figura 2.6 ilustra el modelo diferencial.

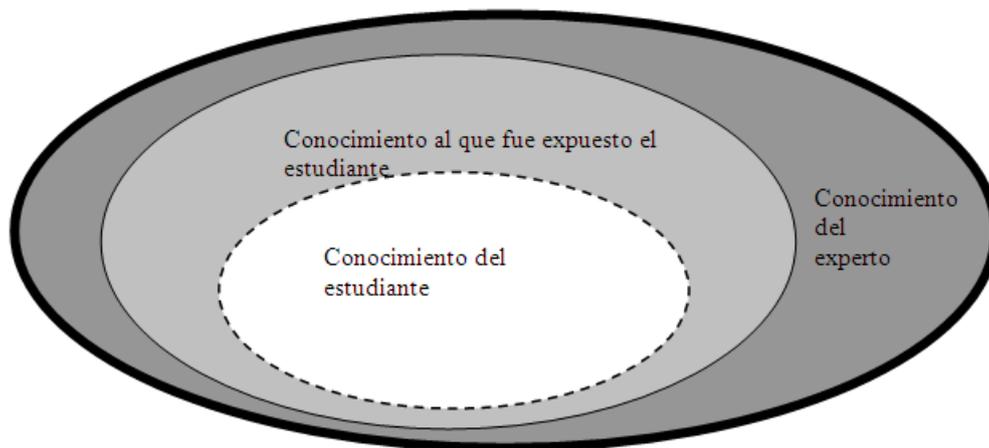


Figura 2.6: Representación del Modelo del Estudiante diferencial

WEST [15] es un sistema tutorial que guía a un estudiante mientras juega un juego diseñado para alentar el uso de destrezas aritméticas básicas. Se presentan tres números generados aleatoriamente, el estudiante genera y evalúa una expresión aritmética cuyo resultado es un número que se mueve en un tablero. El objetivo del juego es alcanzar el fin del tablero antes que el oponente computarizado, el cual juega con las mismas reglas.

WEST emplea un modelo de estudiante diferencial donde las características del juego no introducidas se registran como las diferencias con el estudiante. Se emplea un mecanismo de trazado para medir qué características del juego son empleadas por el estudiante.

Desde un tablero particular el sistema determina todos los posibles movimientos basados en los tres números presentados al estudiante. La solución óptima se compara con el ingreso del estudiante. Basado en esta comparación, un mecanismo de guía determina cuando es apropiado interrumpir al estudiante con alguna advertencia.

Modelo de perturbación Mientras que el modelo de superposición representa el conocimiento del alumno en términos del conocimiento ‘correcto’, el modelo de perturbación lo combina con una representación del conocimiento incorrecto. De este modo, no se considera al alumno como un ‘subconjunto’ del experto, sino que el conocimiento del alumno puede ser potencialmente diferente en calidad y cantidad al del experto. La técnica más frecuente para implementar un modelo de perturbación es representar el conocimiento experto y añadirle los errores que más frecuentemente cometen los alumnos. El modelo del alumno es, entonces, un modelo de superposición sobre este conjunto de conocimiento aumentado con conocimiento correcto e incorrecto. En la literatura aparecen dos tipos de errores: errores de concepto (misconceptions) y fallos o erratas (bugs). La colección de errores que se incluye en un modelo de perturbación se llama biblioteca o catálogo de errores.

Esta biblioteca puede construirse de dos formas diferentes:

- Proceso enumerativo: mediante un análisis empírico se listan todos los posibles errores mediante el análisis del dominio y los errores que comete el estudiante.
- Proceso reconstructivo: se determinan los errores cuando el estudiante aplica inapropiadamente los operadores durante alguna tarea procedimental incluida en la lección.

Aunque la información adicional en un modelo de perturbación proporciona nuevas explicaciones del comportamiento del alumno, introduce también nuevos problemas: el esfuerzo necesario para construir y mantener el modelo del alumno es mucho mayor.

El modelo de perturbación (o modelo buggy) se ve en la figura 2.7. Este modelo permite manipular conocimiento poseído por el estudiante que no está presente en el conocimiento del experto.

El sistema tutor BUGGY [16] [11] presenta a los estudiantes una serie de ejercicios de substracción. El modelo de diagnóstico refleja el conocimiento de una habilidad en términos de sub-habilidades correctamente aprendidas y su representación se basa en redes procedimentales (una red procedimental para una habilidad en particular divide a la habilidad en sub-procedimientos y sub-objetivos). Con la red procedimental se comparan las respuestas generadas por el sistema con las producidas por el estudiante. Si los estudiantes responden incorrectamente la red se cambia para introducir el error en la

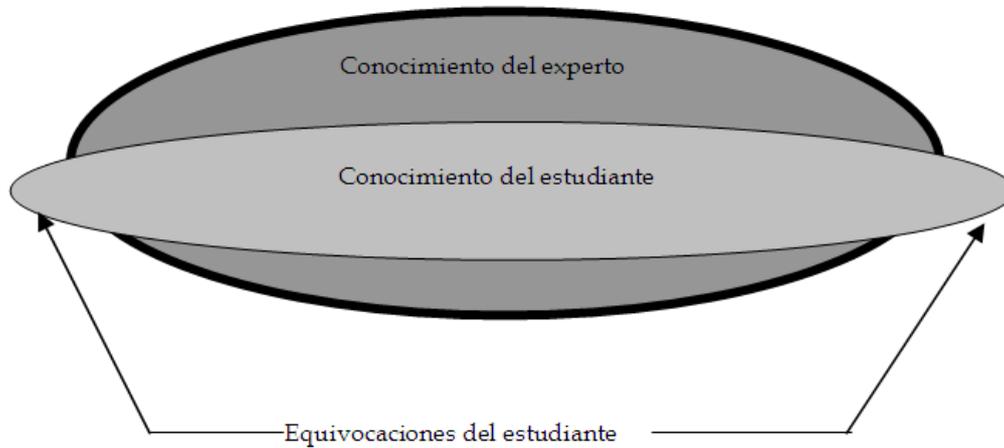


Figura 2.7: Representación del modelo de perturbación o buggy

sub-destreza. Este proceso se repite hasta que las respuestas del estudiante son pronosticadas por la red. Si el estudiante en forma consistente comete los mismos errores en la aplicación de la sub-destreza el error se atribuye al estudiante.

El sistema tutor IDEBUGGY fue desarrollado basándose en BUGGY. El catálogo de errores en las sub-destrezas usadas por IDEBUGGY fue desarrollada mediante la enumeración de errores presentes en la base de datos de las interacciones estudiante/tutor.

Modelos basados en trazas Para los modelos basados en trazas puede distinguirse dos tipos diferentes, *traza del conocimiento* y *traza del modelo* ([4]).

La *traza del conocimiento* consiste en determinar lo que sabe el alumno, incluyendo tanto el conocimiento correcto sobre el dominio como sus errores. Por otro lado, la traza del conocimiento resulta útil para la evaluación del alumno y la toma de decisiones pedagógicas, como que material/problema debe ser propuesto a continuación.

La *traza del modelo* pretende analizar el procedimiento de resolución de problemas que utiliza el alumno. Esta resulta útil en sistemas que intentan dar respuesta a peticiones de ayuda del alumno y ofrecerle pistas e información cuando no sabe como seguir resolviendo el problema. De hecho, para poder ayudar al alumno el sistema necesita ser capaz de analizar y criticar su solución en curso y tener una idea de que línea de razonamiento está siguiendo. Este modelo analiza episodios de la resolución de un problema y mantiene un modelo de la resolución del problema completo el cual es contrastado

con la actividad del estudiante. Dado un estado de la resolución del problema, algunas reglas del modelo de actividades de la resolución de problemas se disparan para predecir el próximo estado. Reglas del experto y de los errores son disparadas con el objetivo de corresponderse con el nuevo estado del estudiante. Se asume que el estudiante usará la regla predicha en el nuevo estado y el Modelo del Estudiante se actualiza con la nueva información. Cualquier desviación entre la solución del estudiante y la del sistema puede ser tomada en cuenta y buscar una acción apropiada.

Asimismo se encuentra el modelo de *trazado del resultado* (Issue tracing). En este modelo se analiza un episodio de la resolución del problema para determinar que habilidades o resultados han sido usados. La calificación de cada habilidad individual es actualizada en el Modelo del Estudiante para indicar que fue usada. Este modelo no intenta modelar el proceso de resolución de problemas y no requiere de modelo de resolución de problemas.

LISP TUTOR [2] emplea un modelo de trazado para proveer al aprendiz con una adecuada realimentación. Se presenta un problema al aprendiz y el tutor controla la entrada del estudiante, carácter a carácter. El tutor genera todos los posibles caracteres siguientes usando tanto las reglas correctas como las de las equivocaciones:

- Si el carácter se predice con la regla correcta, el aprendiz puede continuar
- Si el carácter se predice con una regla de equivocación, se muestran instrucciones para remediarlo
- Si el carácter no se predice, el tutor dice que no entiende y pide al estudiante que trate de nuevo. Después de varios intentos, el tutor explica el próximo paso

Este método tiene la ventaja de un diagnóstico temprano de los errores del aprendiz y la posibilidad de una realimentación inmediata. De esta manera el estudiante nunca se aleja de la solución correcta. Sin embargo, esto puede verse como una restricción innecesaria y contraproducente porque el estudiante nunca puede explorar un comportamiento incorrecto.

2.2.3. Modelo de enseñanza

De acuerdo a Wenger [83], cuando ‘el aprendizaje es visto como transiciones sucesivas entre estados del conocimiento, el propósito de la enseñanza es, por consiguiente, facilitar el pasaje del espacio de estados del conocimiento’. Los STIs tienen que modelar el conocimiento del estudiante y sostener la transición a un nuevo estado de conocimiento. Wenger sostiene que el STI alterna entre el diagnóstico y soporte didáctico. El autor sugiere que la didáctica puede ser organizada siguiendo cuatro principios:

1. *Planes de acción*: es una ‘mini curricula’ usada para guiar al estudiante y provee el contexto para las operaciones de diagnóstico
2. *Contextos estratégicos*: son los contextos en los cuales los planes de acción son implementados. Generalmente, estos son planificados (por el maestro) u oportunista (el aprendiz hace algo o hace preguntas para generar la oportunidad para la tutoría)
3. *Base de decisión*: reglas o guías para asignar los recursos del sistema en el contexto de restricciones. En este tema los diseñadores han tomado decisiones divididas. Por ejemplo, SOPHIE [12] tiene un dominio experto en reparaciones electrónicas pero no tiene la capacidad de explicación; puede demostrar gran variedad de procedimientos, pero no puede explicar que hace en el contexto del ‘entendimiento’ del estudiante. Un intento para lograr explicaciones generales lo encontramos en MENO [32] el cual sacrifica el dominio experto.
4. *Nivel del Modelo del Estudiante*: selección del nivel en el cual toma lugar la enseñanza. En cualquier momento, el estudiante puede encontrarse en niveles de comportamiento, conceptual o metacognitivo, o en niveles múltiples simultáneamente. Un maestro experto humano puede entender y responder apropiadamente a estos niveles. Algunos sistemas se centran en el comportamiento o intentan determinar el nivel en cual el estudiante necesita asistencia y ajusta el plan de acción adecuadamente.

Como puede inferirse, la construcción de sistemas de enseñanza sofisticados necesita de la representación del conocimiento del maestro. Además, de resolver problemas en el dominio y entender al estudiante, los STIs tienen que enseñar a un estudiante como resolver problemas. Estos sistemas son más complejos que un sistema experto porque enseñar un tópico, a menudo, es más difícil que ‘conocer’ el mismo tópico.

Para alcanzar los objetivos enumerados, el módulo pedagógico usa información del Modelo del Estudiante para determinar los aspectos del dominio que deberían ser presentados al aprendiz. Esta información puede ser, por ejemplo, nuevo material, una revisión de tópicos previos, o realimentación sobre un tópico. Un aspecto de gran importancia es la selección de meta-estrategias para enseñar el dominio. Por ejemplo, el sistema puede decidir el uso del método Socrático o seleccionar un tópico y presentar un ejemplo de un problema con ese tópico. Una vez que se seleccionó una meta-estrategia, debe decidir aspectos de bajo nivel como, por ejemplo, el uso del ejemplo exacto.

2.2.3.1. Decisiones de bajo nivel

El tutor decide el contenido del material que será presentado a un estudiante particular. Esto involucra decisiones sobre el tópico, el problema y la realimentación.

Selección del tópico: para seleccionar el tópico el tutor examina el Modelo del Estudiante. Por ejemplo, si la meta-estrategia indica que la revisión es correcta, el tutor selecciona un tópico que el estudiante ya aprendió. Por otro lado, si se tiene que presentar nueva información, el tutor elige un tópico que el estudiante no conoce.

Generación del problema: una vez que el tópico fue seleccionado, se debe generar un problema para que el estudiante lo resuelva. La granularidad (nivel de detalle) del problema queda determinada por el dominio. Por ejemplo, en SHERLOCK, se pregunta al estudiante sobre el diagnóstico de fallas en la estación usada para reparar un F-15, mientras en MFD, se le da al estudiante un problema simple de matemática, tal como sumar dos fracciones. Es importante que la dificultad del problema sea apropiada al nivel de habilidad del estudiante la cual puede ser determinada a partir del Modelo del Estudiante.

Realimentación: muchos tutores trabajan muy bien mientras los estudiantes hagan todo bien. Pero, los problemas aparecen cuando estos tienen dificultades y necesitan ayuda del tutor. En esta situación, el tutor debe determinar que tipo de realimentación proporcionará. Decidir la cantidad de ayuda que se tiene que dar al estudiante también es un análisis bastante complejo. Debe tenerse en cuenta que dar poca realimentación puede llevar a la frustración mientras que mucha realimentación puede interferir con el aprendizaje. Una vez que el sistema decide cuanta realimentación es necesaria, debe determinar el contenido del consejo. La realimentación debería contener suficiente información para que el estudiante pueda proceder al siguiente paso en la resolución del problema. Además, el consejo dado debe ser apropiado a su nivel de habilidad. Algunos sistemas usan el Modelo del Estudiante para seleccionar que pista es la adecuada a la habilidad del aprendiz.

2.3. Conclusión del capítulo

La ventaja más importante del software educativo basado en IA (en especial los tutores inteligentes) es su habilidad para aprender a partir de las acciones de sus usuarios (los estudiantes y los docentes) y adaptarse a sus rasgos particulares. Para alcanzar tal objetivo, estos sistemas buscan mejorar su rendimiento continuamente, aunque los resultados alcanzados no son uniformes en sus distintas áreas de estudio. Se debe tener en cuenta que la naturaleza de los STIs, si bien proviene de diversas fuentes, es idiosincrática.

Considerando la descripción anterior, la investigación sobre STIs avanza en varias áreas, de las cuales las más importantes son:

- Modelado del estudiante: se han presentado varios modelos a fin interpretar el

progreso del estudiante, incluyendo el diagnóstico de los errores, el seguimiento del estudiante a lo largo del tiempo, representación de errores. Es quizás, el área de investigación más activa actualmente y la que cuenta con importante logros. El objetivo perseguido es modificar la trinidad tradicional a fin de alcanzar la instrucción adaptativa.

- **Modelado cognitivo:** la investigación en esta área involucra la creación de representaciones simbólicas plausibles de razonamiento necesario para resolver problemas en un dominio determinado. Los modelos resultantes se usan para proponer acciones a los estudiantes, generar realimentación y proveer bases para el Modelo del Estudiante. El área ha asimilado varias tecnologías a fin de mejorar su objetivo, como técnicas de Aprendizaje Automático.
- **Intercambios con la ciencia de aprendizaje:** en el módulo de enseñanza de la trinidad tradicional no se han tratado principios de teorías de aprendizaje, como si se hará en la nueva trinidad. Se verá que la investigación en STIs se superpone con diversas teorías de aprendizaje e instrucción.
- **Selección de las meta-estrategias:** La selección de estrategias de alto nivel en STIs no ha recibido la misma atención que las decisiones de bajo nivel, además de señalar que hay poca investigación sobre el tema. La investigación en el área ha identificado muchas estrategias de tutoría potenciales para su uso en STI. Como ejemplo se puede nombrar la enseñanza en espiral o el método Socrático.

La implementación de las meta-estrategias continúa siendo un tema de estudio e investigación. Muchos STIs no identifican explícitamente las estrategias que usan para enseñar e implementan implícitamente alguna de ellas. Un mejor método es usar el Modelo del Estudiante para seleccionar una estrategia apropiada. Idealmente el modelo puede seguir la pista de las estrategias instruccionales que son más efectivas para enseñar.

Sin embargo, como muchos sistemas no tienen estrategias múltiples, los modelos de los estudiantes no han sido diseñados para seleccionar tal información. En consecuencia, la representación explícita de estrategias múltiples y el conocimiento del control para seleccionarlas está fuera de la capacidad de muchos sistemas existentes. Otro obstáculo es la dificultad en la representación del conocimiento para estrategias tutoriales, como por ejemplo, el método Socrático requiere ‘sentido común’ muchas veces no representable en una base de conocimiento (problema común en otras áreas de IA). Cualquier estrategia global debe estar fundamentada en principios psicológicos y educacionales con referencia a las necesidades identificadas en el modelo del estudiante.

Capítulo 3

Contexto teórico del diseño de STIs

3.1. Introducción

En la década de los 80s los STIs salieron del laboratorio de desarrollo a las aulas. Inicialmente, las reacciones no fueron favorables. En la literatura del área no hay referencias a fundamentos educacionales, sólo a bases computacionales. Los sistemas tutoriales carecían de bases sólidas teóricas nacidas de modelos de aprendizaje. Los desarrolladores analizaron muy poco dichos modelos a fin de tener marcos conceptuales apropiados.

Paralelamente, el constructivismo se transformó en el tema dominante en la psicología educacional. El enfoque de los constructivistas buscaba una perspectiva más holística que el conductismo o el paradigma del procesamiento de la información. Los constructivistas sostienen que la cognición no puede ser reducida a interacciones entre cierta cantidad de 'cajas negras'. En 1987, Wenger [83] revisó los objetivos de los diseñadores de STIs. Él consideró que los STIs ('comunicación del conocimiento' como él los denomina) deberían focalizarse en los aspectos cognitivos y el aprendizaje en estos sistemas, además de fortalecer la contribución de la IA. Wenger señaló lo que debería ser la base del desarrollo del área como una combinación del trabajo de investigadores desde la IA, ciencia cognitiva y educación:

... considere el ejemplo de los libros: superan a las personas en la precisión y permanencia de su memoria, y en la fiabilidad de su paciencia. Por esta razón, son invaluable para la humanidad. Ahora, imaginemos libros activos que puedan interactuar con el lector para comunicarle el conocimiento en su nivel apropiado, selectivamente destacando las interconexiones y ramificaciones de los ítems, recordar información relevante, investigar el entendimiento, explicar áreas difíciles más profundamente, saltar sobre material aparentemente conocido ... sistemas de comunicación de conocimiento inteligentes son más que un atractivo sueño.

Wenger hizo un llamado para pasar a ‘una forma orientada cognitivamente de la ingeniería de software’ donde la cognición sea central, en lugar de concentrarse en los modelos computacionales. Él señaló que en los 80s había dos visiones opuestas sobre los STIs: el punto de vista tradicional como dispositivos de entrega instruccional y el emergente enfoque de computadoras para el aprendizaje exploratorio. Su visión de los STIs es concebirlos como herramientas inteligentes de comunicación del conocimiento.

3.2. Teorías de aprendizaje

En el aula, el profesor enseña usando estrategias instruccionales que le resultan efectivas. Para alcanzar tal objetivo profesional, el docente usa su concepción de la enseñanza y el aprendizaje. Algunas estrategias ven al estudiante como una vasija vacía que debe ser llenada bajo la dirección del maestro; otras estrategias buscan que el estudiante sea un activo participante en el aprendizaje a través de la resolución de problemas; otras estrategias conciben a los alumnos como organismos sociales que aprenden a través del diálogo e interacción con los otros.

Las estrategias esbozadas en el párrafo anterior poseen fundamentos en varias teorías de aprendizaje, por ejemplo, el modelo conductista de Dick and Carey, el modelo constructivista de Willis, la teoría de la Elaboración de Reigeluth, la teoría ACT de J. Anderson, la teoría de la Transacción de Merrill, la jerarquía del aprendizaje de Gagné y otras.

Todas ellas ilustran la abundancia de los marcos teóricos que pueden servir para asistir al diseñador en su toma de decisión pedagógica. Considerando que las teorías evolucionan en el tiempo como consecuencia de críticas y revisiones, es muy probable que las visiones opuestas señaladas por Wenger, terminen mezcladas en un enfoque conciliador que extraiga lo mejor de cada una.

A continuación se enuncian someramente tres teorías de aprendizaje (Anderson, Self y Piaget) y cuatro teorías instruccionales (Merril, Reigeluth, Bloom y Gagné) consideradas apropiadas para el desarrollo de STIs.

3.2.1. Teoría de J. Anderson

John Anderson ha trabajado en ciencia cognitiva y desarrolló en 1983 la teoría cognitiva ACT* (Adaptive Control of Thought) [1]. Esta teoría es uno de los más tempranos intentos de una teoría de la cognición humana. Combina conocimiento declarativo en la forma de redes semánticas con conocimiento procedimental en la forma de reglas de producción. En ACT* , el aprendizaje se alcanza mediante la formación de nuevos procedimientos a través de la combinación de reglas de producción existentes. Los principios

más importantes de la teoría son:

1. Las funciones cognitivas pueden ser representadas como un conjunto de reglas de producción. El uso de una producción depende del estado del sistema y de los objetivos.
2. El conocimiento es aprendido declarativamente a través de instrucciones. El aprendiz debe ejecutar el proceso de compilación del conocimiento. Las producciones deben ser entendidas apropiadamente e integradas en su conocimiento existente para luego ser llamadas y usadas.

En el cuadro 3.1 se resumen los principios de la teoría ACT* y los principios asociados de diseño para STI [25].

Anderson y sus colegas crearon ACT* como una teoría cognitiva, sosteniendo que es lo suficientemente rigurosa para probar sus principios mediante la implementación en software. Dos ejemplos muy conocidos basados en ACT* son los sistemas Geometry Tutor [48] y LISPITS (LISP Intelligent Tutoring System).

En 2006, en el Centro de Cognitive Tutor Authoring Tools de la Universidad de Carnegie Mellon, a partir del desarrollo original Geometry Tutor, el grupo de J. Anderson ha desarrollado varios STIs, siendo el último BRIDGE TO ALGEBRA cuya la finalidad es enseñar tópicos de Álgebra a alumnos de las escuelas secundarias.¹

3.2.2. Teoría de J. Self

La teoría propuesta por John Self [74] reconoce la necesidad de formalizar una teoría de aprendizaje, quizás como un conjunto de sentencias describiendo como el estado cognitivo de un estudiante cambia como consecuencia de acciones instruccionales u otros eventos. En general podríamos definir un conjunto de axiomas de la forma:

$$\text{Estado}(\text{es}, \text{s1}, \text{t}) \ \& \ \text{Evento}(\text{e}, \text{x}) \ \rightarrow \text{Estado}(\text{es}, \text{s2}, \text{e}(\text{x}, \text{t}))$$

Este axioma tiene el siguiente significado: si el estudiante **es** está en el estado cognitivo **s1** en situación **t** y **e** es un evento con parámetro **x**, entonces, el estudiante se encontrará en estado **s2** en la situación alcanzada luego del evento **e**, esto es, **e(x, t)**.

Por ejemplo, una teoría basada en dificultades (impasse-based) de aprendizaje desde ejemplos podría tener el siguiente axioma (simplificado), entre otros:

¹Se puede consultar en: <http://ctat.pact.cs.cmu.edu/index.php?id=timeline>

Hipótesis de ACT*	Principios de tutoría
El comportamiento en la resolución de problemas está dirigido por objetivos	Comunicar la estructura de objetivos subyacentes en las tareas de resolución de problemas
Los conocimientos declarativos y procedimentales están separados. Las unidades de conocimiento procedimental son reglas IF-THEN llamadas producciones	Representar el conocimiento del estudiante como un conjunto de producciones
El cumplimiento inicial de una tarea se hace mediante la aplicación de procedimientos débiles (en general) a estructuras de conocimiento declarativo	Proveer instrucción en el contexto de resolución de problemas; desarrollar el conocimiento del estudiante a través de aproximaciones sucesivas hacia la habilidad buscada
Las producciones para tareas específicas resultan de aplicar producciones débiles al conocimiento declarativo. En estas producciones para tareas específicas subyace un rendimiento más eficiente	Dar realimentación inmediata en los refuerzos
Como resultado de práctica adicional las producciones pueden ser encadenadas en producciones en gran escala	Modificar el tamaño del paso de la instrucción a medida que progresa el aprendizaje
El estudiante mantiene el estado actual del problema en una memoria de trabajo de capacidad limitada	Minimizar la carga de memoria de trabajo

Cuadro 3.1: Principios de ACT* y sus implicaciones para STI

Estado(es, vacio, t) & Evento(presente-positivo, [p, f, a]) ->
 Estado(es, Cree(es, f->a), presente-positivo([p, f, a], t))

Con este axioma se quiere expresar: si el estudiante **es** no sabe el tema **t** y le presentamos un ejemplo positivo **p** el cual tiene las características **f** y donde la acción **a** fue usada para resolver el problema, entonces el estudiante creerá que cualquier problema con características **f** puede ser resuelto aplicando la acción **a**.

3.2.3. Teoría constructivista de Piaget

En la concepción de Piaget la educación tiene por finalidad favorecer el crecimiento intelectual, afectivo y social del niño, teniendo en cuenta que ese crecimiento es el resultado de procesos evolutivos naturales. La acción educativa, por tanto, ha de estructurarse de manera que favorezca los procesos constructivos personales, mediante los cuales opera el crecimiento. Las actividades de descubrimiento deben ser por tanto, prioritarias. Una de las características básicas del modelo pedagógico piagetiano son las interacciones sociales horizontales.

Las implicaciones del pensamiento de Piaget en el aprendizaje inciden en la concepción constructivista del aprendizaje. Los principios generales del pensamiento piagetiano sobre el aprendizaje son:

- El principio básico es la primacía del método de descubrimiento.
- El aprendizaje es un proceso constructivo interno.
- La interacción social favorece el aprendizaje.
- El aprendizaje depende del nivel de desarrollo del sujeto.
- Los objetivos pedagógicos deben, además de estar centrados en el estudiante, partir de las actividades del alumno.
- Los contenidos, no se conciben como fines, sino como instrumentos al servicio del desarrollo evolutivo natural.
- El aprendizaje es un proceso de reorganización cognitiva.
- Las experiencias de aprendizaje deben estructurarse de manera que se privilegie la cooperación, la colaboración y el intercambio de puntos de vista en la búsqueda conjunta del conocimiento (aprendizaje interactivo).

La teoría de Piaget establece que cuando los individuos cooperan en el medio, ocurre un conflicto sociocognitivo que crea un desequilibrio, que a su vez estimula el desarrollo cognitivo. En un aula tradicional, el *aprendizaje cooperativo* involucra la ejercitación en grupos pequeños, con estudiantes trabajando en equipos sobre problemas y bajo la guía

del maestro. La tarea del docente es guiar a los grupos, asegurando que los estudiantes realizan la tarea y mostrando la respuesta correcta.

Además, Piaget sostiene que hay cuatro factores que inciden e intervienen en la modificación de las estructuras cognoscitivas: la maduración, la experiencia, el equilibrio y la transmisión social. Todos ellos se pueden propiciar a través de ambientes colaborativos. El *aprendizaje colaborativo* (presentado someramente en el apartado 1.3.4), es otro de los postulados constructivistas que parte de concebir a la educación como proceso de socioconstrucción que permite conocer las diferentes perspectivas para abordar un determinado problema, desarrollar tolerancia en torno a la diversidad y pericia para reelaborar una alternativa conjunta. Los entornos de aprendizaje constructivista se definen como un lugar donde los alumnos deben trabajar juntos, en pequeños grupos, ayudándose unos a otros, usando una variedad de instrumentos y recursos informativos que permitan la búsqueda de los objetivos de aprendizaje y actividades para la solución de problemas. No hay un maestro como autoridad, por lo tanto nadie guía a los grupos o corrige los errores. Las actividades de aprendizaje en estos grupos causan que los estudiantes integren experiencias, conocimiento y creencias, y al mismo tiempo, conocimiento y creencias son formados dentro de cada aprendiz.

3.3. Teorías Instruccionales

La Ciencia InstruccionaI nació en 1966, con la Teoría de la Instrucción de Bruner, y ha crecido marcadamente con el trabajo en Ciencias Cognitivas de Ausubel, Gagné, Merrill para nombrar unos pocos. Se construyó sobre las Ciencias del Aprendizaje (Psicología del Aprendizaje, Sociología del Aprendizaje, Ciencias de Sistemas) y comprende teorías, modelos y metodologías para Instrucción y para Investigación en Instrucción. Como lo ha definido Simon, la Ciencia InstruccionaI es una Ciencia de Diseño. El foco de los estudios en Ciencia InstruccionaI es la interrelación entre cuatro clases de variables: situación instruccionaI, tema, resultados instruccionales y estrategia instruccionaI [70].

El Diseño InstruccionaI es un proceso sistemático y sistémico para aplicar estrategias y técnicas derivadas de las teorías conductista, cognitiva y constructivista para encontrar soluciones al problema instruccionaI. Representa la aplicación sistémica de la teoría y otro conocimiento organizado a la tarea de diseño y desarrollo instruccionaI. Los subprocesos incluyen: análisis (del dominio, objetivos y metas, características del estudiante, contexto y restricciones), diseño (decisiones de alto nivel basadas en estrategias), desarrollo (decisiones de bajo nivel basadas en actividades de aprendizaje y evaluaciones, y material instruccionaI) y gerenciamiento (del diseño y del proceso de entrega)

Hay tres tipos de estrategias en las teorías de Diseño InstruccionaI:

- **Estrategias organizacionales:** son separadas en niveles micro o macro y tratan la forma en que las lecciones son organizadas y secuenciadas.
- **Estrategias de distribución:** están relacionadas con las decisiones que afectan la manera en la cual la información se lleva al estudiante, particularmente la selección del medio instruccional.
- **Estrategias de administración:** involucran las decisiones sobre la ayuda al aprendiz cuando este interactúa con las actividades diseñadas para el aprendizaje.

3.3.0.1. Teoría de Merrill

La teoría de Merrill llamada Component Display (CDT) describe elementos micro de instrucción [55]. Ha sido diseñada para trabajar en conjunción con la teoría de Reigeluth (la cual se presenta en el apartado siguiente). CDT comprende tres partes:

1. Una dimensión de rendimiento-contenido que comprende el nivel deseado del rendimiento del estudiante y tipo de contenido.
2. Cuatro formas de presentación primaria.
3. Un conjunto de prescripciones relativas al nivel de rendimiento y tipo de contenido de las formas de presentación.

La teoría clasifica al aprendizaje en dos dimensiones:

1. **Contenido:** es el contenido real que será aprendido, va desde los hechos (la forma más básica de contenido) a los principios. Los cuatro tipos de contenido son: hechos (nombres, fechas, eventos), conceptos (símbolos, eventos, y objetos que comparten características y son identificados por el mismo nombre), procedimientos (conjunto de pasos ordenados y secuenciados para resolver un problema) y principios (causa-efecto o relaciones, explican o predicen por qué algo ocurre de una determinada manera).
2. **Rendimiento:** confección de la recordación, uso y generalidades. Para recordar, se le requiere al estudiante que busque en su memoria un ítem particular de información; luego lo usa aplicando esa información a un caso específico y con esa información puede derivar una nueva abstracción (conceptos, principios, etc.).

3.3.1. Teoría de la elaboración

Charles Reigeluth fue un estudiante de doctorado de Merrill. Ha usado un enfoque que es consistente con la teoría CDT. El autor cree que la instrucción se representa en capas y cada capa de instrucción elabora las ideas previamente presentadas (mejorando la retención). Las capas permiten ampliar el enfoque de secuenciación que va desde lo simple a lo complejo y desde lo general a lo específico [69]

La instrucción debe estar organizada en un orden creciente de complejidad para lograr un aprendizaje óptimo. Por ejemplo, cuando se enseña una tarea procedimental, la versión más simple de la tarea debe ser presentada primero; luego, las lecciones mostrarán versiones adicionales hasta que la totalidad de la tarea es enseñada. En cada lección, se le debe recordar al estudiante todas las versiones enseñadas presentando un resumen/síntesis. La idea clave de la teoría de la elaboración es que el aprendiz necesita desarrollar un contexto significativo en el cual ideas y habilidades subsecuentes puedan ser asimiladas.

La teoría de la elaboración propone siete componentes de estrategias mayores:

1. Una secuencia elaborativa
2. Secuencias de pre-requisitos de aprendizaje
3. Resumen
4. Síntesis
5. Analogías
6. Estrategias cognitivas
7. Control del aprendiz

El primer componente es el más crítico de esta teoría. La secuencia elaborativa se define como una secuencia de lo simple a lo complejo en la cual la primera lección compendia (en vez de resumir o abstraer) las ideas y habilidades que siguen. El compendio debe ser hecho en base a un tipo de contenido simple (conceptos, procedimientos, principios), aunque dos o más tipos pueden ser elaborados simultáneamente, y debería involucrar el aprendizaje de sólo unas pocas habilidades fundamentales en el nivel de aplicación.

Se afirma que esta teoría da por resultado la formación de estructuras cognitivas más estables y por lo tanto se alcanzan tres aspectos muy importantes, a saber: mejor retención y transferencia, motivación del aprendiz incrementada a través de la creación de contextos de aprendizaje significativos y la provisión de información acerca del contenido que permite el control del aprendiz. La teoría de la elaboración es una extensión del trabajo de Ausubel (organizadores avanzados) y Bruner (curriculum en espiral).

Para el diseño de un STI, un desarrollador puede utilizar los dos principios básicos de esta teoría:

1. La instrucción será más efectiva si sigue una estrategia de elaboración, es decir, uso de compendios (sumarios) conteniendo motivadores, analogías, resúmenes y síntesis.

2. Hay cuatro tipos de relaciones importantes en el diseño de la instrucción: conceptuales, procedimentales, teóricos y pre-requisitos de aprendizaje.

3.3.2. Taxonomía de B. Bloom

En 1956, Benjamín Bloom (líder de un grupo de psicólogos educacionales quienes desarrollaron una clasificación de niveles de comportamiento intelectual en aprendizaje) creó una taxonomía² que incluye tres dominios solapados: cognitivo, psicomotor y afectivo. Considerando que el diagnóstico cognitivo del alumno es un componente importante del Modelo del Estudiante, la taxonomía (ver cuadro 3.2) proporciona una estructura útil para categorizar las pruebas del diagnóstico. Cada capacidad puede establecer un Objetivo Instruccional a alcanzar (conocimiento, comprensión, aplicación, análisis, síntesis y evaluación). La autora de esta tesis ha usado las tres primeras capacidades en el STI que está desarrollando (se presentan más detalles en el capítulo final).

3.3.3. La teoría de las condiciones del aprendizaje de Robert Gagné

Esta teoría propone la existencia de diferentes tipos o niveles de aprendizaje [37]. El aspecto importante de esta clasificación es que cada tipo requiere niveles diferentes de instrucción. Gagné ha identificado cinco grandes categorías de aprendizaje: información verbal, habilidades intelectuales, estrategias cognitivas, habilidades motoras y actitudes. Condiciones internas y externas son necesarias para cada tipo de aprendizaje. Por ejemplo, como estrategia cognitiva para aprender, debe existir una oportunidad para desarrollar nuevas soluciones a problemas; para aprender actitudes, el estudiante debe estar expuesto a modelos creíbles o argumentos persuasivos.

Gagné sugiere que las tareas de aprendizaje para lograr habilidades intelectuales pueden estar organizadas en una jerarquía siguiendo su complejidad: reconocimiento por estímulo, generación de respuesta, seguimiento de un procedimiento, uso de terminologías, discriminaciones, formación de conceptos, aplicación de reglas, y resolución de problemas. La significancia primaria de la jerarquía es para identificar prerrequisitos que puedan facilitar el aprendizaje en cada nivel. Los prerrequisitos son identificados haciendo un análisis de tareas de una situación de aprendizaje. Las jerarquías de aprendizaje proveen una base para la construcción de la secuencia de la instrucción.

Además, la teoría propone nueve eventos instruccionales y su correspondiente proceso cognitivo:

²Adaptado de: Bloom, B.S. (Ed.) (1956) *Taxonomy of educational objectives: The classification of educational goals: Handbook I, cognitive domain*. New York ; Toronto: Longmans, Green.

Capacidad	Habilidad demostrada
Conocimiento	<ul style="list-style-type: none"> ■ Observación y recordación de información ■ Conocimiento de fechas, eventos , lugares ■ Conocimiento de ideas principales <p><i>Preguntas: listar, definir, decir, describir, identificar, mostrar, ,examinar, etc.</i></p>
Comprensión	<ul style="list-style-type: none"> ■ Entendimiento de la información ■ Significado de la comprensión ■ Traducir conocimiento a un nuevo contexto <p><i>Preguntas: resumir, describir, interpretar, asociar, distinguir, estimar, discutir, etc.</i></p>
Aplicación	<ul style="list-style-type: none"> ■ Uso de la información ■ Uso de métodos, conceptos, teorías en nuevas situaciones ■ Resolver problemas usando habilidades adecuadas o conocimiento <p><i>Preguntas: aplicar, demostrar, calcular, completar, mostrar, resolver, examinar, clasificar, experimentar, descubrir, etc.</i></p>
Análisis	<ul style="list-style-type: none"> ■ Ver patrones ■ Organización de partes ■ Identificación de componentes <p><i>Preguntas: analizar, separar, ordenar, explicar, conectar, clasificar, dividir, comparar, seleccionar, explicar, inferir, etc.</i></p>
Síntesis	<ul style="list-style-type: none"> ■ Usar viejas ideas a fin de crear nuevas generalizando a partir de hechos ■ Relacionar conocimiento desde varias áreas ■ Predecir, extraer conclusiones <p><i>Preguntas: combinar, integrar, re-adaptar, substituir, planear, crear, diseñar, inventar, formular, preparar, generalizar, re-escribir, etc.</i></p>
Evaluación	<ul style="list-style-type: none"> ■ Comparar y discriminar entre ideas ■ Hacer elecciones entre argumentos razonados ■ Verificar valores de evidencia <p><i>Preguntas: evaluar, decidir, clasificar, calificar, probar, medir, recomendar, seleccionar, juzgar, explicar, discriminar, comparar, resumir, etc.</i></p>

Cuadro 3.2: Taxonomía de los objetivos educativos de Bloom

1. *Atraer la atención:* presentar un problema o una situación nueva. Se puede usar ‘artefactos de interés’: un cuento, presentar un problema o demostrar algo.
2. *Informar el objetivo al aprendiz:* esto permite al aprendiz organizar su pensamiento y por lo tanto decidir que ver, escuchar o hacer.
3. *Estimular el recuerdo del conocimiento previo:* esto permite al estudiante construir sobre su conocimiento o habilidades anteriores, proveyendo de una estructura que lo ayuda a aprender y recordar.
4. *Presentar el material:* mostrar la información relevante para evitar recargar la memoria. Armonizar la información para ayudar a recordar la información.
5. *Dar guía para aprender:* son instrucciones sobre cómo aprender.
6. *Producir rendimiento:* práctica que permite al aprendiz hacer algo con el comportamiento, habilidades o conocimiento recientemente adquiridos.
7. *Permitir realimentación:* mostrar la corrección de la respuesta del aprendiz y analizar su comportamiento (pruebas, comentarios) .
8. *Evaluar el rendimiento:* prueba para determinar si las lecciones han sido aprendidas. Se puede dar información general sobre su progreso.
9. *Resaltar la retención y transferencia:* informar al estudiante sobre situaciones de problemas similares, proveer práctica adicional, revisar la lección.

3.4. Arquitectura de la nueva trinidad a la luz del constructivismo

Una de las hipótesis básicas del constructivismo es que el conocimiento no puede ser objetivamente definido ni estáticamente representado (como lo son los Modelos de Estudiantes convencionales). Mediante su actuación en ese mundo, los estudiantes asimilan nuevos conceptos a sus previamente construidas estructuras cognitivas o modifican sus estructuras cognitivas para acomodar las interpretaciones según sus nuevas experiencias. En consecuencia, el conocimiento y la acción no pueden estar separados y la actividad y el contexto de una experiencia son parte integral del significado de esa experiencia. Por lo tanto, el foco está sobre el proceso por medio del cual el conocimiento es construido en vez del conocimiento del dominio que debe ser adquirido.

En este contexto constructivista, los especialistas analizan la naturaleza del aprendizaje y conocimiento, a fin de descubrir las propiedades que estos deben poseer para que efectivamente cumplan su objetivo. Esta hipótesis sugiere que los diseñadores deberían desarrollar sistemas que posean tales propiedades.

Las propiedades de un proceso de aprendizaje dependen de las creencias del estudiante y de lo que hace mientras interactúa con el sistema. Por lo tanto, los sistemas deberían poder adaptarse a fin de asegurarse esas propiedades deseables. Para habilitar tal adaptación, un sistema necesita modelar las propiedades de las interacciones entre el sistema y el estudiante.

El propósito es, entonces, modelar el proceso de aprendizaje a lo largo del tiempo, con interacciones entre estructuras cognitivas con el contexto, utilizando la actividad. Se necesita describir los contextos ambientales donde ocurren las interacciones y los eventos que podrían transcurrir en una posible situación, sus precondiciones y sus efectos. Una serie de eventos crean una secuencia de situaciones a las cuales podríamos considerar un curso de interacción.

En un curso de interacción particular, se pasa a través de una secuencia de situaciones s_1, s_2, \dots, s_m por medio de una serie de eventos e_1, e_2, \dots, e_{m-1} (ver figura 3.1).

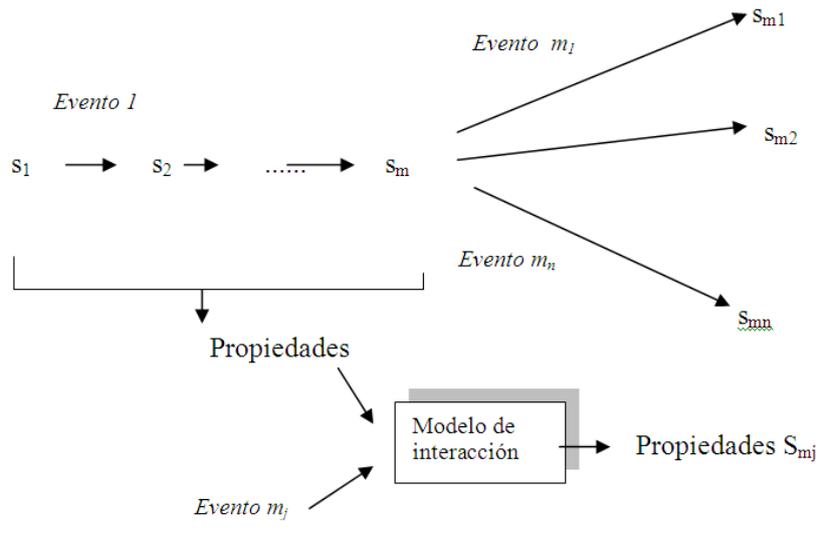


Figura 3.1: Adaptación a través del modelado de interacciones

Cuando se alcanza la situación s_m se dispone de un posible conjunto de eventos $e_{m1}, e_{m2}, \dots, e_{mn}$. Si un evento particular e_{mj} ocurre entonces una nueva situación s_{mj} sería alcanzada. El curso de acción (potencial) $s_1, s_2, \dots, s_m, s_{mj}$ poseería un conjunto de propiedades $P_{mj} = p_1, p_2, \dots, p_r$. Puede decirse que la situación s_m afronta cada una de estas propiedades p_i a través del evento e_{mj} . En general hay un conjunto de propiedades potenciales disponibles para distintos cursos de interacciones.

Existen ciertas similitudes entre las figuras 2.4 y 3.1. El aspecto focalizado aquí es

el evento. Se dice que un curso de interacción es bueno cuando el estudiante tiene más probabilidad de aprender, pero no se intenta representar lo que aprende el estudiante. Esto parece estar más cerca del espíritu del constructivismo en cuanto a que el ‘contenido’ del aprendizaje está íntimamente relacionado con la ‘actividad’ del aprendizaje, y lo que realmente se aprende es una construcción hecha por el estudiante a partir de sus propias actividades. En este caso, no se consideró el conocimiento que se desea alcanzar, simplemente se representaron los cursos de interacción por los que pasa el aprendiz, teniendo como objetivo alcanzar el aprendizaje.

Volviendo a la división tripartita convencional de los STI, vemos que tal arquitectura se modifica a la luz de este nuevo enfoque [74]. En particular, es claro que el intento de desarrollar sistemas inteligentes para soportar aprendizaje no es inherentemente contradictorio a la visión constructivista del aprendizaje. A fin de mejorar la presentación de los conceptos se reconsiderará los tres componentes tradicionales de un STI (figura 3.2):

Modelo del dominio del conocimiento (dominio): un diseñador de STI puede asumir que el conocimiento se describe en términos, hechos, principios, etc. los cuales pueden ser representados simbólicamente y jerárquicamente, y aprendidos de un modo incremental.

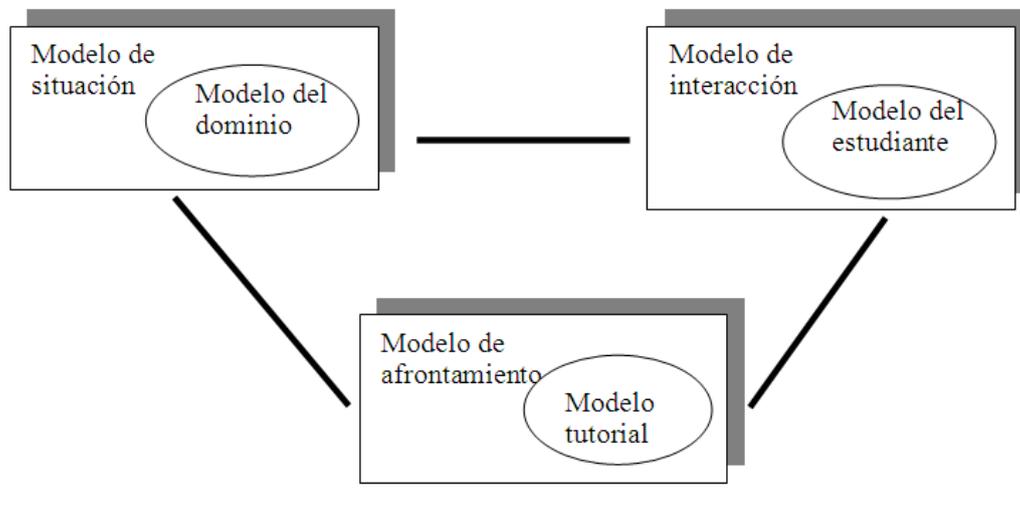


Figura 3.2: Componentes de la nueva trinidad

Por lo tanto, todo el esfuerzo se centra en desarrollar complejas representaciones de tal conocimiento. Los constructivistas enfatizan que los aprendices construyen por sí mismos a través de la interpretación de sus propias experiencias en un contexto de interacciones. En consecuencia, el diseñador de ambientes de aprendizajes orientados

hacia el constructivismo se centra no en la representación del conocimiento sino en la naturaleza de situaciones, contextos e interacciones.

Esta situación conduce a consideraciones del contenido de contextos y a la dinámica del proceso de aprendizaje. Sin embargo, un modelo de situación puede contener representaciones de aspectos del dominio de conocimiento al cual el estudiante puede acceder durante las interacciones. Estas representaciones pueden parecer similares a los modelos de dominio de conocimiento, pero se debe tener en cuenta que su propósito puede ser muy diferente. No son descripciones del conocimiento a alcanzar, sino son descripciones de recursos disponibles en una situación de aprendizaje. Desde esta perspectiva, un modelo del dominio de conocimiento puede ser visto como un subconjunto de una noción más amplia de un modelo de situación.

Modelo del conocimiento del aprendiz: el Modelo del Estudiante de un STI es, típicamente, determinado mediante el análisis de las interacciones del estudiante con el modelo del dominio y sirve para determinar errores y faltas, que luego serán la base de las intervenciones instruccionales. Sin embargo, como argumentan los constructivistas, el proceso constructivista individual del estudiante (conduciendo a construcciones personales, quizás no relacionado a algún conocimiento) es más importante que el producto particular de algún proceso de aprendizaje, por lo tanto el Modelo del Estudiante debería centrarse más en el proceso interactivo, extendido en el tiempo, teniendo en cuenta las acciones del aprendiz, el contexto en el cual ocurre y las estructuras cognitivas. El desarrollo de tal ‘modelo del proceso de interacción’ permite considerar los tipos de regularidades de secuencias de interacción que acercan las propiedades que benefician o dificultan el aprendizaje.

Las estructuras cognitivas pueden formar parte de las descripciones del proceso interactivo en el tiempo, y la significancia de eventos interactivos pueden depender de cogniciones individuales. Sin embargo, los aspectos de las estructuras cognitivas a ser consideradas son de una naturaleza diferente y asumen roles diferentes a los de un STI estándar, porque estos son tomados en relación al contexto y actividad que constituyen interacciones de aprendizaje. Así, la noción de un modelo de proceso de interacciones es, de alguna manera, un superconjunto del Modelo del Estudiante tradicional.

El modelo de conocimiento de enseñanza: los diseñadores consideran que sus sistemas pueden, más o menos determinísticamente, encontrar planes instruccionales mediante la interpretación de su Modelo de Estudiante con respecto a la estructura del currículum basada en su modelo del dominio del conocimiento. Los constructivistas podrían argumentar que el proceso de aprendizaje es demasiado impredecible para ajustarse dócilmente al análisis por medio de estructuras pre-especificadas y que las secuencias de aprendizaje emergen desde interacciones entre el aprendiz y el ambiente.

Por lo tanto, el rol pedagógico del sistema no es determinar eventos instruccionales sino proveer espacios útiles para la interacción de los aprendices basados en algún modelo de afrontamiento (affordance) de situaciones potenciales. El ‘modelo de afrontamiento’ se centra en las propiedades particulares de cursos de interacciones enfrentadas. Esto es porque, de acuerdo a la visión constructivista, el proceso de aprendizaje es más importante que el producto.

Sin embargo, a partir de una visión objetivista del conocimiento, puede desarrollarse un modelo de afrontamiento en términos de los ítems de conocimiento los cuales pueden ser aprendidos con eventos particulares (como por ejemplo, un evento de STI tal como la presentación de una acción para remediar un error puede ser considerada como un afrontamiento al aprendizaje de un ítem de conocimiento que debe ser corregido). Por lo tanto puede verse al modelo de afrontamiento como un modelo más amplio que el modelo de enseñanza.

3.5. Conclusión del capítulo

El entendimiento de las teorías de aprendizaje conduce a los diseñadores instruccionales a desarrollar mejores ambientes educativos. Por ejemplo, la teoría de J. Anderson ha sido el fundamento de gran variedad de tutores inteligentes desde 1993. El diseñador dispone, además de las teorías, de nuevas herramientas para desarrollar aplicaciones educativas más inteligentes según los requerimientos de las soluciones que se necesitan actualmente, como se verá en el capítulo 4.

La presentación de marcos conceptuales se hizo a través de teorías de aprendizaje e instruccionales a fin de mostrar la variedad de bases teóricas sobre las que se puede construir mejores STIs. Se ha presentado un modelo de arquitectura de un ambiente de aprendizaje basado en computadora que incluye la arquitectura de STI estándar como un subconjunto (figura 3.2). Puede pensarse, en consecuencia, que su desarrollo será más difícil de alcanzar que un STI tradicional. Sin embargo, el contenido de estos componentes puede ser más simple de implementar que aquellos que esforzadamente se realizan en un STI tradicional. Por ejemplo, un modelo de proceso de interacción del estudiante basado en propiedades de cursos de interacción podría ser más fácil de construir que los modelos cognitivos de alta fidelidad basados en consideraciones detalladas de estructuras de conocimiento como se usa en los STI convencionales.

El diseñador de STIs puede adherir a una o más teorías de aprendizaje que sean apropiadas para su concepción del sistema. Inclusive la teoría conductista, no tratada en este informe por considerarla poco apropiada para los ambientes inteligentes de aprendizaje, puede facilitar el aprendizaje de tareas con un bajo grado de procesamiento, donde se

usan estrategias de estímulo/respuesta y realimentación/refuerzo. La teoría conductista puede usarse para la asociación de pares, discriminación de conceptos o memorización rutinaria.

La teoría de la Elaboración de Reigeluth puede servir para el diseño instruccional introduciendo al estudiante en los conceptos principales de un curso y pasar posteriormente a un estudio más personalizado y significativo en algún contexto particular del tema tratado.

Aquellos diseñadores que adhieren a los tutores cognitivos tratan con tareas de un nivel creciente de procesamiento como clasificación, reglas y procedimientos. Estas tareas están asociadas a estrategias con gran énfasis cognitivo como la organización en esquemas, razonamiento analógico y resolución de problemas algorítmicos. La fortaleza de este enfoque radica en el dominio que adquiere el estudiante para hacer una tarea de manera consistente. Su debilidad se encuentra en el modo que el alumno aprende una tarea, no necesariamente el mejor; tampoco está adaptado a su individualidad ni ajustado al contexto.

La propuesta de la teoría constructivista supone un estudiante con un alto nivel de procesamiento (por ejemplo, resolución de problemas heurísticos o control de estrategias cognitivas). El aprendizaje en el constructivismo trata con estrategias más sofisticadas que el cognitivismo, como el aprendizaje situado o la negociación social. La fortaleza del constructivismo se basa en la libertad que el estudiante tiene para interpretar múltiples realidades y la capacidad que adquiere para tratar con situaciones de la vida real. Pero, esa misma libertad se puede convertir en una debilidad en situaciones donde se necesiten ciertas limitaciones dentro de un contexto específico, como por ejemplo, los métodos de evaluación del conocimiento dentro de una universidad; en la teoría constructivista puede pensarse a cada estudiante proponiendo su propio modo para ser evaluado, sin seguir los procedimientos de la institución educativa.

Como ejemplos de áreas de aplicación, según los principios de aprendizajes apropiados [35], se puede citar los temas propios de una profesión a la luz del enfoque conductista (conocer el 'que'); las estrategias cognitivas son útiles para enseñar la resolución de problemas donde hechos y reglas son aplicadas a situaciones no familiares; y finalmente, las estrategias constructivistas son especialmente apropiadas para tratar con problemas abiertos a través de la reflexión.

Capítulo 4

Contexto práctico del diseño de STIs

4.1. Tecnologías para el diseño

Muchos desarrollos presentan funcionalidades similares a las presentadas en la ‘trinidad STI’, arquitectura básica descrita en el capítulo 2. Pero en los detalles, se han implementado muchas arquitecturas, tal vez tantas como desarrolladores de STIs existan. Muy pocas de ellas han sido re-usadas. Tampoco se observa en la literatura una misma arquitectura aplicada a gran variedad de problemas de la educación. Sin embargo, algunas ‘variantes’ pueden ser desarrolladas para ajustarlas a varias categorías de dominios de STI, como por ejemplo, para enseñar tópicos matemáticos o físicos.

El progreso en la investigación en STIs ha sufrido retrasos debido a la falta de arquitecturas modulares, componentes reusables y bases de conocimiento compartidas. Cada nueva investigación o esfuerzo de desarrollo parte de cero, muchas veces sin fundamentos teóricos. Según la literatura, las metodologías actuales para el desarrollo de STIs sufren de varios problemas, a saber:

- Cada aplicación es desarrollada independientemente una de otra
- El conocimiento experto de enseñanza es difícil de codificar en aplicaciones individuales y además es fuertemente dependiente del dominio
- Existe muy poco re-uso de los componentes de un STI, como el Modelo del Estudiante
- Existe la necesidad de un lenguaje estándar para representar el conocimiento y un conjunto de herramientas para manipular ese conocimiento

A fin de exponer algunos de los esfuerzos hechos por la comunidad de STI con el objetivo de afrontar los problemas delineados, se presentarán algunas de las tecnologías usadas en su desarrollo: agentes, Aprendizaje Automático, razonamiento basado en casos y sistemas basados en la web.

4.1.1. Diseño basado en agentes

No existe acuerdo unánime sobre el concepto de agente inteligente de software. Los agentes inteligentes han sido caracterizados de diversas maneras. Sus habilidades varían significativamente, dependiendo de sus roles, capacidades y ambientes.

Los agentes inteligentes involucran cuatro conceptos fundamentales que definen su comportamiento [85, 47]: autonomía, sensibilidad o reactividad, pro-actividad (en el sentido de estar siempre preparado para ayudar) y habilidad social. Hay una fuerte corriente para caracterizar a los agentes según la adaptación, pro-actividad (en el sentido de estar en movimiento) e intencionalidad [27, 56, 67]. Hay también varias taxonomías creadas para agentes. De acuerdo a [61] hay ocho categorías de agentes - colaborativos, interfaz, móviles, de información, reactivos, híbridos, heterogéneos e inteligentes. Franklin y Graesser [36] identifican una taxonomía basada en la de Nwana. Los agentes inteligentes han sido asociados con gran variedad de funciones como por ejemplo asistentes personales, administradores de información, buscadores de información, agentes de planificación, representantes del usuario, etc.

Desde el punto de vista de arquitecturas un agente actúa como mediador entre las tareas específicas que ejecuta el usuario y la aplicación de un software específico. Los agentes se descomponen en módulos funcionales que se refieren a la percepción, modelado, planificación, coordinación, y ejecución de planes o tareas. El aspecto más importante es que los agentes pueden proveer la habilidad de ejecución y coordinación, independientemente y autónomamente, de las subtareas que conducen a completar la tarea del usuario.

La naturaleza dinámica y distribuida tanto de datos como de aplicaciones requiere que el software no sólo deba responder a la requisitoria por información sino anticiparse, adaptarse y activamente buscar formas de ayuda al usuario. En este contexto, el usuario espera asistencia dedicada desde la aplicación que usa. Para responder a estos requerimientos los agentes de software tendrán un rol importante en la interacción humano-computadora y en la coordinación de los procesos internos del sistema.

Desde el punto de vista funcional el agente puede verse en diferentes roles. Las estructuras conceptuales juegan un papel importante en la implementación del agente. Pueden ser aplicadas para la representación del conocimiento y sus mecanismos de manipulación, navegación y definición de problemas.

Los agentes, en el desarrollo de sistemas para educación, se presentan como soportes eficientes para los requerimientos de los nuevos ambientes de aprendizaje como adaptación, personalización, soporte para la movilidad del usuario, efectividad, etc. Además de la habilidad para procesar la autonomía y cooperación entre ellos, los agentes tienen las capacidades para manejar la seguridad, tanto en línea como fuera de línea.

A continuación se presentan ejemplos del uso de agentes en sistemas tutoriales:

- MATS (Multi-agent tutoring system) [79] es un sistema distribuido que soporta la relación ‘un estudiante-muchos maestros’. Cada agente MATS representa a un tutor capaz de enseñar un tema distinto y todos los agentes son capaces de colaborar entre ellos para resolver las dificultades de aprendizaje que tenga el estudiante. La arquitectura del sistema consta de una arquitectura de agentes y una plataforma multi-agente, lenguaje de intercambio de conocimiento para las tareas de aprendizaje y una ontología general del ambiente de aprendizaje como fundamento para compartir conocimiento. MATS puede ser usado para soportar colaboración de objetos de aprendizaje heterogéneos y esto permitiría la rápida expansión del ambiente hacia el mundo que lo rodea.
- Ambiente de aprendizaje donde un agente soporta al aprendiz actuando en un espacio de aprendizaje multi-dimensional [68]. Debe encontrar ciertas piezas del material de aprendizaje y resolver problemas. El camino individualizado a través del espacio ayuda al aprendiz a construir su propio modelo del tópico en el espacio de aprendizaje.
- La tecnología de agentes pueden proveer ayuda a los estudiantes cuando trabajan en grupo y en línea [53]. Cada estudiante que trabaja en el proyecto tendrá un agente, operando en un segundo plano, observando su progreso, midiéndolo con respecto al plan instruccional y tomando acciones para remediar los errores cuando sea necesario. Estos agentes interactúan con los agentes-estudiantes para asegurar que el proyecto sea completado satisfactoriamente.
- Agentes interfaz pueden actuar como mediadores entre aprendices y el sistema tutor, proveyendo un ambiente colaborativo [80]. El agente será autónomo, dirigido por el objetivo, dinámico y colaborativo. Estos agentes administran los hábitos y debilidades de los estudiantes a fin de adaptar sus acciones didácticas. Para el aprendiz, estos agentes ofrecen una interfaz común flexible.
- El agente ‘facilitador’ ayuda a un usuario en el proceso de aprendizaje [9].
- Los sistemas SLS (Social Learning System)[20] extienden el paradigma LCS (Sistemas de Compañero de Aprendizaje o Learning Companion Systems, ya vistos en el apartado 1.3.5). Los Sistemas de Aprendizaje Social sostienen el trabajo de estudiantes y agentes en una misma computadora o con computadoras conectadas

bajo diferentes protocolos de actividades de aprendizaje. La noción de agente es central a estos modelos.

4.1.2. Diseño basado en técnicas de Aprendizaje Automático

El Aprendizaje Automático (AU) (también conocido como Machine Learning) es una subárea de la Inteligencia Artificial interesada fundamentalmente en el diseño, análisis, implementación y aplicaciones de programas que aprenden de la experiencia. El AU ofrece algunos de los enfoques más efectivos de adquisición automática de conocimiento en disciplinas donde se involucra gran cantidad de datos (bioinformática, neuroinformática, informática ambiental, informática social, informática de los materiales, etc.). Los algoritmos de aprendizaje también pueden ser usados para modelizar aspectos del aprendizaje humano y animal. El Aprendizaje Automático es la auto-modificación que ocurre en un programa. A partir de una misma entrada, ejecuciones diversas del programa arrojan salidas diferentes.

El AU trata con el diseño y desarrollo de técnicas y algoritmos que permiten a las computadoras ‘aprender’ (en el sentido de adquirir conocimiento) a través de la extracción de información de datos automáticamente. Sus campos de estudio involucran el aprendizaje supervisado, aprendizaje no-supervisado, algoritmos de aprendizaje simbólico y algoritmos de aproximación de funciones.

Como ya se trató, los STIs tienen que adquirir conocimiento de varias fuentes (estudiante, teorías de aprendizaje, dominio, etc.). El modelado y representación de ese conocimiento requiere un gran esfuerzo y la búsqueda de solución puede consumir gran parte de los recursos volcados para el desarrollo de STIs. A pesar de este hecho, en la última década, muy pocos resultados de Aprendizaje Automático han sido aplicados al desarrollo de sistemas tutoriales [77]. Varias de sus técnicas podrían ser aplicadas para mejorar y aumentar las bases de conocimiento durante el uso de los STIs. A continuación se presentan y ejemplifican varias de esas técnicas:

- El *aprendizaje rutinario* es memorización [23]. Un sistema de este tipo no necesita entender o interpretar la información dada por el ambiente. Lo único que hace es almacenar nueva información para uso posterior. Elsom-Cook [34] no estuvo lejos de este modelo usando modelos de estudiante de superposición y bibliotecas de errores.
- El *algoritmo de focalización* [86] es considerado una de las técnicas más poderosas para aprender conceptos y busca producir una definición consistente con todos los datos positivos, pero con ninguno de los negativos. Usa búsqueda en el espacio de versiones y en el espacio de conceptos. El espacio de conceptos cubre todas las posi-

bles descripciones de conceptos. El espacio de versiones cubre solamente descripciones de conceptos consistentes con los ejemplos de entrenamiento. El concepto que será aprendido se representa por un conjunto de árboles. Hay un árbol por cada atributo usado para describir el concepto. El método de distinguir miembros de un concepto de los no miembros ha sido usado por varios STIs. La implementación del aprendizaje de un concepto ha sido considerado como básico para construir modelos de estudiante dinámicos y para esquemas que permitan la colaboración entre estudiante y tutor. Por otro lado, estos algoritmos no pueden manipular descripciones disjuntas de un concepto y solamente el conocimiento anterior usado es representado por árboles del espacio de conceptos.

- Los *métodos basados en explicaciones* (MBE) analizan un ejemplo de entrenamiento para producir una generalización válida junto con una justificación de la generalización en términos del conocimiento del sistema [57]. Las entradas a MBE son:
 - Ejemplo de entrenamiento: un ejemplo es un conjunto de hechos el cual describe una instancia del concepto objetivo
 - Teoría del dominio
 - Criterio operativo : especifica la forma en la cual la definición del concepto generalizado puede ser expresado
 - Concepto objetivo: una descripción de alto nivel del concepto que será aprendido

A posteriori de las entradas, MBE crea una generalización del ejemplo de entrenamiento el cual puede describir adecuadamente el concepto objetivo y también describe el criterio operativo. Además, puede justificar el concepto generalizado. La aplicación de MBE a STI se encuentra, por ejemplo, en PROBIT [62] . Este sistema usa MBE en la construcción del Modelo del Estudiante a partir de las respuestas específicas del estudiante. PROBIT diagnostica conceptos y malentendidos en problemas de probabilidades.

4.1.3. Razonamiento basado en casos (RBC)

El Razonamiento Basado en Casos (RBC) es un paradigma para resolución de problemas que posee una capacidad inherente de aprender. El principio básico de RBC es resolver nuevos problemas mediante la adaptación de soluciones que fueron usadas para resolver problemas en el pasado [50]. Un razonador basado en casos usa una memoria basada en casos que almacena descripciones de problemas resueltos previamente y las soluciones aplicadas. El proceso principal de RBC consiste en la búsqueda de casos similares al problema presente y adaptar las soluciones halladas para ajustar la situación actual. Un sistema basado en casos aprende desde sus experiencias que son almacenadas como nuevos casos.

La técnica de RBC ha sido usada en el campo de AIED para propósitos diferentes, sobretodo para aplicaciones en aprendizaje basadas en el principio de enseñar mediante la presentación de ejemplos (casos).

Otros investigadores [31, 49, 64] han aplicado este paradigma en planificación instruccional. El **caso** define una pieza de un plan instruccional usado previamente y contiene: el contexto en el cual fue aplicado, el plan instruccional y los resultados alcanzados. La inclusión de RBC en un STI puede mejorar el enfoque previo. Para iniciar la presentación, se describe a continuación los elementos que se incluyen en un caso (ver figura 4.1):

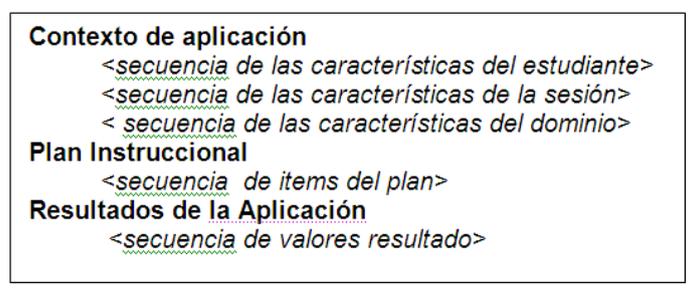


Figura 4.1: Componentes de un caso

- El *Contexto de Aplicación* describe la situación instruccional en la cual el plan fue aplicado y es implementado como una lista de pares (atributo, valor). En general, contiene información extractada desde el Modelo del Estudiante sobre su conocimiento y preferencias. El contexto de aplicación es usado para calcular la similitud entre situaciones instruccionales, así como para estructurar toda la información como índice.
- El *Plan Instruccional* está compuesto de una secuencia variable (o grafo) de ítems del plan instruccional, es decir, una lista de tópicos o una lista de actividades de enseñanza.
- Los *Resultados de la Aplicación* del plan instruccional definen la última componente del caso; contiene los elementos que describen los resultados de la sesión presente, esto es, los cambios detectados en el Modelo del Estudiante.

Para continuar con la presentación de RBC como planificador instruccional, se analiza el **Planificador Instruccional Basado en Casos**(PIBC). El PIBC puede ser integrado en STIs existentes a fin de mejorar su rendimiento. Este planificador podría agregarse

como un nuevo módulo a la arquitectura estándar de los STIs. El PIBC se compone de una memoria para los planes instruccionales y tres módulos funcionales :

- Memoria del Plan Instruccional(MPI): es el repositorio de las experiencias de enseñanza/aprendizaje anteriores al sistema basado en casos. Es responsable del mantenimiento de los casos de una manera organizada a fin de facilitar la recuperación de casos y de almacenar en la memoria.
- Componente de Generación: es responsable de la construcción de planes instruccionales que serán ejecutados a fin de producir sesiones de enseñanza
- Componente de Aprendizaje: es responsable de la actualización de la MPI insertando los casos correspondientes a las sesiones ya desarrolladas. El aprendizaje se hace directamente mediante la incorporación a la MPI de nuevas experiencias del PIBC para usos futuros. Este módulo ejecuta su tarea en dos fases: revisión y retención, las dos últimas fases de RBC.

Durante la fase de revisión este módulo evalúa cada aplicación de un plan instruccional observando los resultados de la sesión de enseñanza.

Durante la fase de retención el componente de Aprendizaje agrega los nuevos casos a la MPI; (se produce una reorganización y actualización de los enlaces como efectos colaterales). La MPI incluye mecanismos robustos para integrar nuevas experiencias en la memoria como también mantener su estructura y propiedades.

- Componente de Evaluación: evalúa la idoneidad de los elementos usados en la generación del plan tomando en cuenta diferentes aspectos como el grado de similitud entre los contextos de aplicación de los casos o los resultados obtenidos.

Para finalizar el tema RBC, se analiza la fase de reuso que combina y adapta los casos recuperados para formar un plan de la sesión apropiado a la situación instruccional presente. La adaptación de casos recuperados a la situación actual es una tarea que necesita conocimiento específico y por lo tanto, se necesitaría un tratamiento diferente para cada STI a fin de proveer al PIBC con un mecanismo efectivo de adaptación. Elorriaga y colegas [33] proponen y definen un enfoque basado en un crítico [41] para crear un mecanismo de adaptación y fue elegido por su flexibilidad. Un crítico (ver figura 4.2) es un fragmento de código de propósitos especiales útil para la adaptación. La estructura del crítico ajusta perfectamente con sistemas basados en reglas. A la izquierda del crítico se definen las condiciones que necesita la adaptación por medio de predicados sobre el plan actual, Modelo del Estudiante y el dominio. El lado derecho contiene el código de adaptación del método.

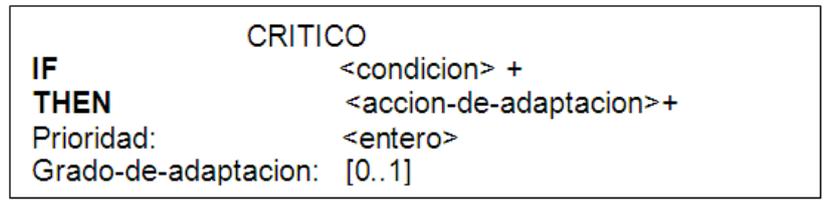


Figura 4.2: Esquema de un crítico

4.1.4. Sistemas Educativos Inteligentes y Adaptativos (SEIA) basados en la Web

La educación basada en la Web se ha transformado en un importante campo de investigación. Los beneficios de tal educación son evidentes: independencia del aula e independencia de la plataforma. El curso instalado y colocado en un lugar puede ser usado por cientos de estudiantes alrededor del mundo con diferentes equipamiento. Durante los últimos cinco años se han desarrollado cientos de cursos y aplicaciones diversas vía Internet. El mayor problema reside en que los cursos son sólo una red de páginas de hipertexto estáticas.

El objetivo desafiante actual es desarrollar aplicaciones educativas que ofrezcan cierta cantidad de inteligencia y adaptación [?]. Estas características son importantes para las aplicaciones de la educación basadas en la Web porque los estudiantes a distancia usualmente trabajan solos (a menudo en su casa). No es fácil disponer de un asistente personalizado e inteligente que nos ayude como si estuviéramos en un salón de clase. La adaptación es importante porque los cursos pueden ser usados por una gran variedad de estudiantes. El curso que sea diseñado para una clase particular de estudiantes no será útil para otros.

Los Sistemas Tutoriales Inteligentes y los Sistemas de Hipermedia Adaptativos dieron lugar a los Sistemas Educativos Inteligentes y Adaptativos (SEIA) basados en la Web. Si bien, muchas de las tecnologías de SEIA-Web fueron heredadas, actualmente la comunidad SEIA ya está produciendo las propias, inspiradas por el contexto de la Web. Como ejemplos de estas tecnologías se encuentran las siguientes:

- Propias del núcleo de los STIs : secuencia del curriculum o planificación instruccional, análisis inteligente de las soluciones del estudiante y soporte para la resolución de problemas interactivos
- Hipermedia adaptativa (aplica formas diferentes de modelos de usuario para adaptar el contenido de los enlaces de las páginas de hipermedia al usuario) : pre-

sentación adaptativa y soporte de navegación adaptativa (SNA).

El objetivo de la presentación adaptativa es personalizar el contenido de una página de hipermedia a los objetivos del usuario, al conocimiento y otra información almacenada en el modelo del usuario (las páginas no son estáticas). Por otro lado, el objetivo de navegación adaptativa es sustentar al estudiante en la orientación y navegación por el hiperespacio mediante el cambio en la apariencia de los enlaces visibles. SNA puede ser considerada como una generalización de la planificación instruccional en el contexto de la hipermedia.

GEKA [63] es un ejemplo donde se unen la robustez y la ‘amigabilidad’ de la multimedia, la sofisticación y complejidad de los Sistemas Tutoriales Inteligentes y la Web. Los autores proponen SITA (Sistema Inteligente de Tutorio Avanzado) como una solución novedosa para la construcción de una nueva generación de sistemas de teleformación orientada al conocimiento procedimental.

SITA se encuentra en la categoría de Sistema Educativo Inteligente y Adaptativo (SEIA) basado en la Web. El sistema enfatiza el desarrollo de las técnicas relacionadas con el modelo del conocimiento, es decir, busca resaltar la adaptabilidad e interoperabilidad del sistema en base a una nueva forma de representación del conocimiento en lugar de fundamentarse en nuevas teorías relacionadas con el Modelo del Alumno, ampliamente desarrolladas en la línea de investigación de Brusilovsky.

SITA está formado por un conjunto de módulos que conforman el sistema inteligente de enseñanza instalados en un sistema de teleformación comercial elegido como plataforma (ver figura 4.3):

- Módulo del Dominio: representación del conocimiento de un dominio determinado mediante un modelo de reglas de producción y material de apoyo en forma de texto, dibujos, videos, etc.
- Módulo del Tutor: representación del conocimiento de profesores y tutores acerca de como impartir determinados contenidos.
- Módulo del Alumno: nivel de conocimiento del alumno; almacena la situación de aprendizaje de cada alumno en cada proceso de enseñanza.
- El sistema de teleformación: se encarga de la gestión y administración de alumnos y cursos, alguno de los cuales puede contener información a enseñar.
- El alumno: interacciona con el sistema inteligente de enseñanza a través de una interfaz gráfica; el sistema guía al alumno presentándole las reglas de conocimiento almacenadas y el alumno decide con sus respuestas el camino a seguir, además el sistema puede presentar información multimedia, a petición del alumno.



Figura 4.3: Arquitectura de SITA

GEKA es una aplicación específica de SITA. GEKA desarrolla e implementa un sistema de teleformación avanzado que se aplica en la formación de operadores en el procedimiento de 'Cambio punzón matriz', fundamental para la utilización de las máquinas punzonadoras de la empresa de Máquinas-Herramientas GEKA (ubicada en Oiartzun, España).

4.2. Conclusión del capítulo

Hasta el momento, los STIs no han alcanzado totalmente las expectativas iniciales que se formularon a comienzos de los 80s en lo que respecta al potencial de las computadoras para mejorar sustancialmente la calidad y los logros del aprendizaje y la instrucción. Un examen crítico sobre la investigación en el área, considerando el actual entendimiento de la naturaleza de los procesos de aprendizaje, demuestra que este resultado no es sorprendente del todo.

Muchos usos educativos de las computadoras se fundamentaron, aunque siempre de manera implícita, en la errada aseveración de que las computadoras por sí mismas promoverían un buen aprendizaje. Asimismo, los desarrolladores han adherido a la concepción del aprendizaje como un proceso pasivo y altamente individual de absorción y acumulación del conocimiento. Esta visión contrasta con una nueva perspectiva sobre el uso productivo de las computadoras en educación. Los especialistas sostienen que las computadoras deberían estar mejor integradas en los programas educativos; deberían

ser utilizadas en ambientes de enseñanza/aprendizaje poderosos, como herramientas que apoyan a los aprendices, en interacción y colaboración con el profesor, con compañeros, y con otros medios de instrucción.

Recordando la temprana promesa de los STIs sobre su potencial de adaptación verdadera al estudiante, se ha percibido que este objetivo es difícil de alcanzar. Todavía, en muchos casos, los sistemas carecen de la habilidad de construir un diagnóstico dinámico (durante la interacción estudiante/STI), en tiempo real. Tampoco la adaptación llega a la provisión de realimentación, tanto en la sesión tutorial como las revisiones posteriores post-sesiones.

Para alcanzar el objetivo de construir buenos STIs, los desarrolladores utilizan diversas tecnologías para la implementación de los sistemas, con resultados dispares. Entre ellas puede citarse al Aprendizaje Automático, agentes pedagógicos, razonamiento basado en casos y la Web. En este Trabajo Final no se han considerado otras tecnologías, también usadas para la implementación de STIs, como redes bayesianas, redes neuronales o algoritmos genéticos.

El caso definido en la figura 4.1 es un buen ejemplo de adaptación donde se usa el razonamiento basado en casos. Asimismo, los agentes están demostrando un gran potencial para la adaptación. Lamentablemente las técnicas de Aprendizaje Automático, en general, no han sido aprovechadas apropiadamente.

La combinación de técnicas de RBC y métodos de filtrado puede servir para predecir estereotipos del estudiante incrementando la calidad de los cursos creados con STIs. Igualmente, RBC puede combinarse con agrupamientos (clusters) de modelos de aprendizaje para encontrar modelos de estudiantes similares a fin de personalizar la enseñanza y mejorar la adaptación de los cursos.

El uso de agentes de software para el aprendizaje humano es variado. Uno de los enfoques más comunes de los agentes es el de un asistente automatizado en un ambiente de enseñanza/aprendizaje. Para alcanzar una enseñanza más efectiva, tanto para los humanos como para los agentes, se necesita un entendimiento profundo del dominio a enseñar y del Modelo del Estudiante. Los agentes han demostrado aprender en ambientes dinámicos tales como el comercio electrónico o fútbol robótico. Si se concibe al estudiante humano como un agente en un ambiente cambiante, más adelante se podrá incluir agentes que aprendan a enseñar, más allá de un agente ‘facilitador’.

Un aspecto importante a considerar es la especificidad de los dominios. Casi siempre se corresponde a campos muy estrechos del conocimiento (integración simbólica, aritmética básica, programación LISP, etc.). Esta problemática está tratando de solucionarse con

representaciones del dominio más abarcativas, como la tecnología de los mapas conceptuales.

Como una síntesis integrada de tecnologías y teorías detrás de un STI, puede mencionarse a GEKA. Este sistema, de alguna manera, ha establecido una tendencia para el futuro; se ha desarrollado en la intersección de la inteligencia artificial, la ciencia cognitiva, la tecnología educativa, y la investigación sobre aprendizaje e instrucción.

Capítulo 5

Conclusión

Los Sistemas Tutoriales Inteligentes han demostrado ser efectivos para incrementar los niveles de aprendizaje y motivación de los estudiantes. Esa efectividad nace del diseño de estos sistemas conformado por cinco componentes: el Modelo del Estudiante, el módulo pedagógico, el dominio de conocimiento a enseñar, el módulo de comunicación y el modelo experto. Si bien la investigación ha sido hecha en todos los componentes, sólo pocos han sido desarrollados apropiadamente, en pos del objetivo primordial de los STIs. La Trinidad Tradicional ha sido la base para gran número de STI, pero J. Self ha sugerido que las tendencias recientes no son reflejadas adecuadamente en estos sistemas. Uno de los principales problemas encontrados en los componentes involucran su bajo nivel de reusabilidad general que lleva a los desarrolladores a tener dificultades en la implementación de funcionalidades compartidas entre diferentes STIs. El Razonamiento basado en casos se presenta como una solución posible para este problema.

La experiencia recogida por la comunidad STI es mucha. Una de las conclusiones más significativas establece que la organización del conocimiento es en sí misma tan importante como el conocimiento. Así también se ha comprobado que el proceso de razonamiento en muchos sistemas tutoriales no se corresponde con el razonamiento de los expertos humanos en situaciones reales (esto es evidente con sistemas basados en sistemas expertos como GUIDON). A fin de alcanzar un alto nivel de efectividad los STIs deberían acercarse al modo de trabajo de un ser humano [3]. Uno de los esfuerzos más importantes para alcanzar este nivel de efectividad es la creación de ambientes altamente interactivos de aprendizaje que usan simulaciones.

El Modelo del Estudiante es el área de investigación más activo en la comunidad de STIs. Se ha mostrado que la principal fuente de información de este modelo proviene del análisis de las acciones de los estudiantes. Las funciones de este modelo comprenden:

- Determinar la exactitud de las respuestas del estudiante al problema (el entendimiento del problema).

- Determinar la exactitud del conocimiento del estudiante referido al tema en estudio (dominio)
- Probar la consistencia de la solución dada por el estudiante considerando los datos del problema y el conocimiento del modelo general. Por ejemplo, en los sistemas de entrenamiento para diagnóstico de alguna enfermedad, probar la consistencia entre las hipótesis de la enfermedad y los síntomas que trata el estudiante.

El volumen y exactitud de la información que el sistema puede derivar de las respuestas del estudiante y de la observación de la actividad del estudiante dependen fuertemente de la complejidad del modelo en uso. Modelos más complejos dan lugar a modelos de estudiantes con mejores diagnósticos.

Los modelos desarrollados parten de modelos muy simples como un vector numérico y llegan hasta modelos complejos como el basado en agentes inteligentes de software o en redes bayesianas dinámicas atómicas, pasando por el tradicional modelo de superposición. Varios de ellos han sido exitosos como los sistemas OLAE[54] y Andes[54] desarrollados para la física.

Pero, el desarrollo del Modelo del Estudiante ha presentado varios problemas, a saber:

- En general, los objetivos de aprendizaje están plasmados en reglas de producción. Esta situación puede confundir a los estudiantes sobre las relaciones entre los conceptos tales como pre-requisitos y similitud.
- No es muy usado el modelo de capas, tomando como tales al dominio, razonamiento y guía al estudiante.
- Existen varios ejemplos que usan redes bayesianas que requieren tiempo exponencial.

Respecto al dominio, existe un movimiento hacia representaciones múltiples del conocimiento a fin de abastecer situaciones específicas y contextos. Se acepta que el experto y el aprendiz trabajan de maneras diferentes. El estudiante pasa por varias fases mientras logra las habilidades de resolución de problemas propios del experto. Se puede requerir representaciones alternativas del dominio a medida que estas fases son establecidas. Pero la representación múltiple del dominio no mapea bien los Modelos de Estudiante por superposición y de errores .

Específicamente, la incorporación de estrategias tutoriales múltiples en el módulo pedagógico es un tema abierto de investigación. Las nuevas tendencias están trasladando las estrategias tutoriales desde la enseñanza basada en hechos hacia la enseñanza basada

en la resolución de problemas y el análisis de estas estrategias. El conocimiento experto instruccional de un STI posee diversas fuentes y no existe acuerdo sobre cual de ellas es la más apropiada. Chi y DuBoulay [21] [30] coinciden en que las fuentes de conocimiento más importantes son los maestros humanos experimentados, teorías de diseño instruccional, observación de los estudiantes, teorías de aprendizaje y teorías creadas a la medida de casos particulares. Hay una brecha entre planificación instruccional para la organización del conocimiento del dominio y la estrategia de enseñanza para la adaptación dinámica del comportamiento del STI.

En los dominios complejos los sistemas tutoriales pueden controlar la secuencia de acciones para inferir el entendimiento de los estudiantes. Por ejemplo, el STI sobre Acciones Tácticas de los Oficiales (TAO), desarrollado por la marina de Estados Unidos, enseña reglas tácticas a los oficiales que dirigen los sensores del armamento a bordo de las naves. El sistema aplica reglas de coincidencia de patrones para detectar secuencias de acciones que indican si el estudiante entiende las tácticas.

Además de la investigación sobre los módulos de un STI, un aspecto importante a destacar es la reducción del tiempo y costo de desarrollo de estos sistemas porque su construcción requiere gran cantidad de trabajo debido a que casi siempre se comienza desde el inicio. Las estrategias seguidas para lograrlo incluyen el desarrollo de *herramientas de autoría* (programas que permiten la construcción de cursos virtuales en donde un profesor puede generar automáticamente y con base en una adecuada asistencia técnica y pedagógica una aproximación a los contenidos y estructura de un curso virtual) para crear STI de forma modular. Resolviendo este problema puede producirse un enorme salto en la investigación del área teniendo en cuenta que muchos más sistemas pueden ser construidos y por lo tanto estudiarse la efectividad de la educación basada en computadoras.

En particular, la autora de esta tesis está trabajando en el diseño e implementación de un STI, llamado InfoSem, actualmente en desarrollo en la Facultad de Ingeniería de la Universidad Nacional de Mar del Plata [44, 45]. El dominio de InfoSem es la matemática propia de las carreras de Ingeniería (actualmente se usan tópicos de análisis numérico). El modelo del estudiante es un modelo de superposición con tres capas de adaptación. En el modelo del instructor diseñado se adhiere a la posición de Du Boulay sobre la experiencia de profesores universitarios como fuente de conocimiento pedagógico. Este conocimiento es un material enriquecido y muy útil para construir un modelo pedagógico adaptado a nuestro ámbito cultural. El enfoque que se propone se basa en estereotipos del docente a fin de administrar la actividad de los maestros que actúan con el STI. Los estereotipos se basan en la tipología creada por Anthony Grasha y Sheryl Hruska Riechmann [38].

La tipología se fundamenta en la relación simbiótica entre las cualidades personales

de los profesores, el proceso instruccional que emplean para transmitir el contenido de sus disciplinas y los estilos de aprendizaje de los estudiantes. Este modelo ilustra las interdependencias entre los tres elementos expuestos y las modalidades con que los docentes pueden utilizarlos en una clase [39] [40]. Los autores de este modelo consideran que su estudio es confiable y válido porque no extrapolan tipos de personalidad a los ambientes creados en un salón de clase como lo hacen los estudios del ciclo de aprendizaje de Kolb/McCarthy, el indicador de tipos de Myers-Briggs y modelo de aprendizaje de Felder-Silverman. Para encontrar los estereotipos dominantes se analizó las experiencias personales educativas de profesores de la Facultad de Ingeniería. Para alcanzar tal fin se usó el inventario de estilos de enseñanza del Prof. Grasha ¹ y observación de clases. El inventario consta de cuarenta preguntas sobre actitudes, percepciones, inclinaciones y preferencias de los profesores con respecto al dictado de sus cursos. Asimismo el cuestionario indaga sobre algunas características personales (género, opinión sobre su agrado sobre el curso) y algunos datos del curso (cantidad de alumnos y cantidad de veces que lo dictó).

Siguiendo la tipología de Grasha, se dispone de 5 perfiles de estudiantes y su correspondencia con los estereotipos docentes. Por ejemplo, el estereotipo docente del *experto* (cuyas características básicas son clases magistrales-orales, lectura de libros investigaciones independientes, exámenes objetivos) se corresponde con el perfil de estudiante *Dependiente/Participativo* (cuyos rasgos principales son: necesidad de guía, participación responsable, les agradan las actividades competitivas). Estas relaciones dan basamento a las estrategias tutoriales usadas.

En InfoSem, el Modelo del Estudiante tiene como objetivo capturar información de la actividad del estudiante para lograr buenos perfiles de los alumnos reales. Se usa el modelo de superposición sobre un mapa conceptual (un ejemplo de mapa conceptual fue presentado en la sección 2.2.1). El modelo posee tres capas de adaptación: establecimiento de las habilidades cognitivas para el curso, selección de las características instruccionales según estilos profesor-estudiante y planificación de las acciones instruccionales para un estudiante particular. La evaluación del conocimiento del estudiante se hace a través de pruebas estándar como la metodología de selección múltiple. Hace un tiempo se incorporó otro tipo de prueba para mejorar la evaluación de los conocimientos del estudiante. El objetivo perseguido fue alcanzar la evaluación de la comprensión y de las habilidades del pensamiento de alto nivel. La metodología usada está basada en el método DistSem [81] el cual extrae la estructura de una red semántica en base a las distancias estimadas entre significados y representa esa información en una matriz de similitudes. Las pruebas basadas en DistSem son utilizadas en los tres roles educativos distintos de la evaluación (diagnóstica, formativa y sumativa).

¹<http://fcrcweb.ftr.indstate.edu/tstyles3.html>

Para finalizar el capítulo de la conclusión, se bosquejan las futuras líneas de investigación de la autora de esta tesis:

1. **Agregar a InfoSem un diagnóstico adaptativo del estudiante centrado la aplicación de un modelo difuso.** Uno de los principales obstáculos en el proceso de diagnóstico en un sistema computacional es la incertidumbre. Esta incertidumbre se origina en la naturaleza abstracta de la cognición humana como es la interpretación del docente sobre el rendimiento del estudiante. La evaluación del logro involucra, generalmente, términos lingüísticos como excelente, bueno, regular, etc. los cuales están fuertemente imbuidos de subjetividad. Estas etiquetas lingüísticas surgen a partir de diversos componentes de la evaluación como cuestionarios, planillas especializadas (como DistSem), etc. El evaluador propone estas etiquetas lingüísticas con las cuales el sistema debe llegar al diagnóstico. Una de las mejores metodologías actuales para administrar estas etiquetas lingüísticas es la lógica difusa. De esto modo se espera alcanzar un mejor nivel de adaptación a los usuarios de InfoSem.

2. **Estudiar la posible incorporación de la teoría ACT-R (Adaptive Components of Thought - Rational) de John Anderson como teoría de aprendizaje.** La inclusión de esta teoría de aprendizaje busca enriquecer el modelo del estudiante de InfoSem. ACT-R [5] es la continuación de la teoría ACT*, presentada en el sección 3.2.1. ACT-R concibe a la cognición compleja como la interacción entre conocimiento declarativo y procedimental. Sus hipótesis básicas están inspiradas en el avance de la neurociencia cognitiva. ACT-R describe una forma de especificación sobre cómo el cerebro se auto-organiza de manera tal que permite el procesamiento individual de módulos para producir cognición. Los puntos sobresalientes de la teoría establecen:
 - a) Existe una distinción entre dos tipos de conocimiento (*declarativo* representado por piezas de conocimientos llamadas *chunks* y *procedimental* sobre cómo se hacen diversas tareas cognitivas reflejadas en reglas de producción)
 - b) Existe una distinción entre el nivel simbólico que involucra estructuras de conocimiento discretas y el nivel subsimbólico que involucra procesos neuronales basados en la activación que determinan la disponibilidad de las estructuras simbólicas

Los dos niveles de análisis del conocimiento son importantes para la educación. El primer nivel, el simbólico, es donde se adquieren las estructuras de conocimiento que son los objetivos de un programa educativo. El segundo nivel, el subsimbólico, es donde la práctica juega un rol crítico en la acumulación de las activaciones de nivel básico y en la fortaleza de las reglas de producción. Solamente cuando estas activaciones y producciones

han alcanzado niveles adecuados se logra el conocimiento. Alcanzar tal nivel robusto de conocimiento hace que los estudiantes elijan consistentemente estrategias para usar ese conocimiento.

Un punto de investigación particular es analizar la relación entre el modelo del instructor basado en los estereotipos de los docentes y los principios de la teoría ACT-R.

Bibliografía

- [1] Anderson, J. R. (1983). *The Architecture of Cognition*. Cambridge, Massachusetts: Harvard University Press.
- [2] Anderson, J. R., y Reiser, B. J. (1985). *The LISP tutor*. Byte, 10:159-175..
- [3] Anderson John R. (1988). *The Expert Module*. In Martha C. Polson & J.J. Richardson. 1988. Foundations of Intelligent Tutoring Systems. LEA. Hillsdale, New Jersey.
- [4] Anderson J.R., Corbett A.T., Koedinger K.R., y Pelletier R. (1995). *The cognitive tutors: lessons learned*. Journal of the Learning Sciences. 4(2):167-207.
- [5] Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., y Qin, Y . (2004). *An integrated theory of the mind*. Psychological Review 111, (4). 1036-106
- [6] Arona G. M., Huapaya C. R. (2002) *La representación del dominio para la generación de Sistemas Tutoriales Inteligentes*. Revista Internacional Información Tecnológica ISSN: 0716-8756. 13(1):185-192.
- [7] Ausubel D. P. (1968). *Educational Psychology: A Cognitive View*. New York: Holt, Rinehart & Winston.
- [8] Barross, B., Mizoguchi, R., Verdejo F. (2001) *A platform for collaboration analysis in CSCL. An ontological approach*. Proceedings Artificial Intelligence in Education AIED2001. [En red] <http://sensei.ieec.uned.es>.
- [9] Baylor, A.L. and Kim, Y. (2005) *Simulating Instructional Roles through Pedagogical Agents*. International Journal of Artificial Intelligence in Education. 15: 95-115.
- [10] Bloom, B. S. (1984). *The 2 sigma problem: The search for methods of group instruction as effective as one-to-one tutoring*. Educational Researcher, 13(6):4-16.
- [11] Brown, J.S., y Burton, R.R. (1978) *Diagnostic models for procedural bugs in basic mathematical skills*. In Cognitive Science, pp. 155-192.
- [12] Brown, J. S., Burton R.R. y de Kleer J. (1982) *Pedagogical, natural language and knowledge engineering techniques in SOPHIE I, II AND III*. En: D. Sleeman & J.S. Brown (eds.) Intelligent Tutoring Systems, New York: Academic Press, pp. 227-282.
- [13] Bruner, J. S. (1961). *The act of discovery*. Harvard Educational Review, 31:21-32.

- [14] Burns , Capps. (1988). *Foundation of Intelligent Tutoring Systems: An Introduction*. En M. Polson & Richardson (eds). *Foundation of Intelligent Tutoring Systems*. Lawrence Erlbaum Associates Publishers, USA.
- [15] Burton, R. y J. Brown (1982) *An Investigation of Computer Coaching for Informal Learning Activities*. En: D. Sleeman & J. S. Brown (eds.) *Intelligent Tutoring Systems*, New York: Academic Press, pp. 79-98.
- [16] Burton R.(1982) *Diagnosing Bugs in a simple Procedural Skill*. En D.II. Sleeman, J.S. Brown (eds.) *Intelligent Tutoring Systems*, New York: Academic Press,pp. 157-183.
- [17] Carbonell, J. R. (1970). *AI in CAI: An artificial intelligence approach to computer assisted instruction*. *IEEE Transaction on Man Machine System*. 1(4):190-202.
- [18] Chan T.W. y Bassin A.B. (1990) *Learning Companion Systems en Intelligent Tutoring Systems: At the Crossroads of Artificial Inyelligence and Education* C. Fraser y G. Gauthier Eds. Capítulo 1. NJ: Ablex Publishing Corporation.
- [19] Chan, T.-W. (1991). *Integration-Kid: A Learning Companion System*. In J. Mylopoulos, R. Reiter (Eds.) *Proceedings of the Twelfth International Conference on Artificial Intelligence IJCAI-91*. 2:1094-1099. Morgan Kaufmann Publishers Inc.
- [20] Chan T. W. (1995) *Social Learning Systems en Emerging Technologies in Education* . T.W. Chan y J. Self eds. AACE, pp. 71-96.
- [21] Chi M., S. Siler, H. Jeong, T. Yamauchi y R. Hausmann (2001) *Learning from human tutoring*, *Cognitive Science*, 25, 471-533.
- [22] Clancey, W. J. (1982) *Knowledge-based tutoring: the GUIDON program*, MIT Press, Cambridge, MA.
- [23] Cohen, P. R. y Feigenbaum W. (1982) *The Handbook of AI*. Volume 3. Los Altos, California: W. Kaufmann.
- [24] Conati, C., Gertner, A. y VanLehn, K. (2002). *Using bayesian networks to manage uncertainty in student modeling*. *User Modeling and User-Adapted Interaction*, 12,371-417.
- [25] Corbett, A. T., Anderson, J. R. (1992). *LISP intelligent Tutoring System: Research in Skill Acquisition*. In J. H. Larkin & R. W. Chabay (Eds.), *Computer-Assisted Instruction and Intelligent Tutoring Systems: Shared Goals and Complementary Approaches* (pp. 73-109). Hillsdale, New Jersey: Lawrence Erlbaum Associates, Publishers.
- [26] Davis, R.B. (1990). *Discovery learning and constructivism*. In R.B. Davis, C.A. Mahler, & N. Noddings (Eds.), *Constructivist views on the teaching and learning of mathematics*. Reston, VA: National Council of Teachers of Mathematics.
- [27] Dennett, D. C. (1987). *The Intentional Stance*. The MIT Press: Cambridge, MA.

- [28] Dillenbourg P. (1996). M Baker, A. Blaye y C. O'Malley. *The evolution of research on collaborative learning*. Learning in Humans and Machines, Pergamon Press.
- [29] Dillenbourg P. (1999) *Collaborative Learning. Cognitive and Computational Approaches*. New York: Pergamon Earli.
- [30] Du Boulay B., Luckin R. (2001) *Modelling Human Teaching Tactics and Strategies for Tutoring Systems*. International Journal of Artificial Intelligence in Education. 12(3), 235-256.
- [31] Du, Z. y McCalla, G.I. (1991). *A Case-Based Mathematics Instructional Planner*. Proceedings of the International Conference on the Learning Sciences, Northwest Univ.,pp. 122-129.
- [32] Duchastel, P. (1989). *ICAI Systems: Issues in Computer Tutoring*. *Computers and Education*, 13(1): 95-100.
- [33] Elorriaga J.A e Fernández-Castro I. (2000). *Tutoring Systems that learn: an application of Case-Based Reasoning for Instruccional Planning*. Workshop:Applying Machine Learning to ITS Design.
- [34] Elsom-Cook, M. (1988). *Guided discovery tutoring and bounded user modeling*. In J. Self (Eds), *Artificial intelligence and human learning*. Chapman and Hall.
- [35] Ertmer P. A. y Newby, T. J. (1993). *Behaviorism, cognitivism, constructivism: Comparing critical features from an instructional design perspective*. *Performance Improvement Quarterly*, 6 (4), 50-70.
- [36] Franklin, S. Graesser, A. (1996). *Is it an agent, or just a program?: A taxonomy for autonomous agents*. Proceedings of the Third International Workshop on Agent Theories, Architectures and Languages. Berlin: Springer-Verlag, pp. 21-35.
- [37] Gagne, R. (1985). *The Conditions of Learning (4th ed.)*. New York: Holt, Rinehart & Winston .
- [38] Grasha A. F. (1996) *Teaching with style: A practical guide to enhancing learning by understanding teaching and learning styles* . Pittsburgh, PA: Alliance Publishers , 1996.
- [39] Grasha A. F.(1994) *A matter of style: The teacher as expert, formal authority, personal model, facilitator, and delegator*. *College Teaching*. 42, 142-149.
- [40] Grasha A. F.(1995) *Teaching With Style: The Integration of Teaching and Learning Styles in the Classroom*. The Professional and Organizational Development Network in Higher Education. <http://www.podnetwork.org>. , 1995, [Consulta: febrero, 2008].
- [41] Hammond K. J., (1989). *Case-based planning*. Academic Press Inc.
- [42] Hartley R., Paiva A. , Self J.(1995). *Externalising Learner Models*. Proceedings de International Conference on Artificial Intelligence in Education, editado por J. Greer, pp. 509-516. Washington. AACE.

- [43] Huapaya C. R., Arona G. M. (2002) *Sistemas de autoría especializados en STIs matemáticos*. Revista Brasileira de Informática na Educação. ISSN 1414-5685. 10(2):37-48 en papel y en <http://gmc.ucpel.tche.br/rbie-artigos>.
- [44] Huapaya C.R., Arona G., Lizarralde F. y Vivas J. (2005) *INFOSEM: Distancia Semántica entre Conceptos como base de la Evaluación Cognitiva*. En el libro *Las Ciencias del Comportamiento en los albores del Siglo XXI*. UNMDP-CONICET. ISBN 987-544107-4. pp. 405-411.
- [45] Huapaya C., Lizarralde F., Vivas J., Arona G.(2007) *Modelo de evaluación del conocimiento en un Sistema Tutorial Inteligente*. Revista Iberoamericana de Tecnología en Educación y Educación en Tecnología. ISSN 1850-9959. 2: 20-29.
- [46] Jonassen, D. , Reeves, T. , Hong N. ,Harvey, D. Y Peters, K. (1997) *Concept Mapping as Cognitive Learning and Assessment Tools*. Journal of Interactive Learning Research. 8(3/4):289-308.
- [47] Jonker, C.M. and Treur, J., *Compositional Verification of Multi-Agent Systems: a Formal Analysis of Pro-activeness and Reactiveness*(1988). En: W.P. de Roever, H. Langmaack, A. Pnueli (eds.), *Proceedings of the International Workshop on Compositionality, COMPOS'97*. Lecture Notes in Computer Science, 1536:350-380. Springer Verlag.
- [48] Koedinger, K. R., Anderson, J. R. (1993). *Reifying Implicit Planning in Geometry: Guidelines for Model-Based Intelligent Tutoring System Design*. In S. P. Lajoie, Ed. & S. J. Derry, Ed (Eds.), *Computers as Cognitive Tools* (pp. 15-45). Hillsdale, New Jersey: Lawrence Erlbaum Associates, Publishers.
- [49] Kolodner, J. L. (1991). *Helping Teachers Teach Science Better: Case-Based Decision Aiding for Science Education*. The International Conference on the Learning Sciences, Proceedings of the 1991 Conference, Birnbaum, L. (Ed.), AACE, pp.274-280.
- [50] Kolodner, J.(1993). *Case-Based Reasoning*. San Mateo: Morgan Kaufmann.
- [51] Koschmann, T.(1996). *CSCL: Theory and practice of an emerging paradigm*. Mahwah N.J. Lawrence Erlbaum
- [52] Lea M., Rogers P., Postmes T. (2002) *SIDE-VIEW: Evaluation of a system to develop team players and improve productivity in Internet collaborative learning groups*. British Journal of Educational Technology, 33 (1), 2002, 53-63.
- [53] Martens A., Uhrmacher A. (2002) *Adaptive Tutoring Processes and Mental Plans*. Intelligent Tutoring Systems : 6th International Conference, ITS 2002, Lecture Notes in Computer Science, Springer Berlin / Heidelberg, pp 71-80
- [54] Martin J. y VanLehn (1995). *Student assessment using Bayesian nets*. Int. J. Human-Computer Studies 42, 575-591

- [55] Merrill, M.D. (1983). *Component Display Theory*. In C. Reigeluth (ed.), *Instructional Design Theories and Models*. Hillsdale, NJ: Erlbaum Associates.
- [56] Meyer J. J. Ch., Schobbens PY (1997) *Formal Models of Agents: An Introduction*. Model Age Workshop: pp. 1-7.
- [57] Mitchell T. M., Keller, R. M. y Kedar-Cabelli, S. T. (1986). *Explanation-based generalization: A unifying view*. *Machine Learning* 1(1):47-80.
- [58] Moen, E. y Boersma, K. (1995). *The Significance of Concept Mapping for Education and Curriculum Development*. *Journal of Interactive Learning Research*. 8(3/4):487-502.
- [59] Murray T. (1996). *Special Purpose Ontologies and the Representation of Pedagogical Knowledge*.
- [60] Novak, J. y Gowling, B. (1988) *Aprendiendo a Aprender*. Martinez Roca.
- [61] Nwana H. (1996) *Software Agents: An Overview*. *Knowledge Engineering Review*, 11(3):1-40, Cambridge University Press.
- [62] Or-Bach, R. y Bar-On, E. (1989). *PROBIT - Implications of an ITS development process*. *Proceedings of the 2nd International Jerusalem Convention on Education*, Jerusalem, Israel.
- [63] Pagés, C., Martínez, J. Gutiérrez, O. Y Díez, T. (2005). *Sistema inteligente de tutorización avanzada (SITA). Un caso de aplicación: GEKA*. *Revista de Educación a Distancia*, ISSN. 2:1578-7680.
- [64] Papagni M., Cirillo V., Micarelli A., Boylan P. (1997), *Teaching through Case-based Reasoning: an ITS Engine Applied to Business Communication*, en *Proceedings of the 8th World Conference on Artificial Intelligence in Education AI-ED*, Kobe, Japan, pp. 111-8.
- [65] Papert, S. (1980). *Mindstorms: Children, computers and powerful ideas*. New York, New York: Basic Books.
- [66] Papert, S. y Harel, I. (1991). *Situating Constructionism*. En: *Constructionism*. Toronto. Ablex Publishing Corporation. Papert, S. y Harel, I. Eds. Disponible en: <http://www.papert.org/articles/SituatingConstructionism.html>. [Consulta: 27/01/2008].
- [67] Rao A. S. (1996). *AgentSpeak(L): BDI Agents Speak Out in a Logical Computable Language*. MAAMAW 1996: pp. 42-55
- [68] Rasseneur D., Delozanne E., Jacoboni P., Grugeon B. (2002) *Learning with Virtual Agents: Competition and Cooperation in AMICO*. *Intelligent Tutoring Systems: 6th International Conference, ITS 2002, Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, pp 61-70.

- [69] Reigeluth, C. M. y Stein, F.S. (1983). *The Elaboration Theory of Instruction*. En C. M. Reigeluth (ed.). *Instructional design: theories and models: an overview of their current status*. Hildsdale, New Jersey: L.Erlbaum. pp. 335-381.
- [70] Reigeluth C., Buderson C. y D. Merrill.(1994) *Is there a Design Science of Instruction?* EnM.D. Merrill y D.G. Twitchell ed. *Instructional Design Theory*. Englewood Cliffs, NJ. Ed. Tech. Publ.
- [71] Reigeluth C. (1999). *Green Book .Module 3: Concept Classification*.
- [72] Schlimmer, J.C., Fisher, D.H.: (1986) *A case study of incremental concept induction*. Proceedings of the 5th National Conference on Artificial Intelligence, Philadelphia, Morgan Kaufmann pp. 496-501.
- [73] Self J. (1995) (draft) *Computational Mathematics: Towards a Science of Learning Systems Design*. Computer Based Learning Unit.University of Leeds. <http://www.cbl.leeds.ac.uk/jas/cm>. Leeds.
- [74] Self J. (1999) *The defining characteristics of intelligent tutoring systems research: ITSs care, precisely*. *International Journal of Artificial Intelligence in Education*. 10:350-364.
- [75] Shortliffe, E.H.(1976) *Computer-Based Medical Consultations: MYCIN*. Elsevier/North Holland, New York.
- [76] Shute, V. 1995. *Smart: Student Modeling Approach for Responsive Tutoring*. *User Modeling and User-Adapted Interaction*, 5(1):1-44.
- [77] Sison, R. y Shimura, M. (1998). *Student modeling and machine learning* *International Journal of Artificial Intelligence in Education*. 9, 128-158.
- [78] Skinner, B.F. (1968). *The Technology of Teaching*. New York: Appleton-Century-Crofts.
- [79] Solomos, K. Avouris, N. (1999). *Learning from multiple collaborating intelligent tutors: An agent-based approach*. *Journal of Interactive Learning Research*, 10(3/4), 243-263.
- [80] Trojahn dos Santos C., Frozza R., Dhamer A., Paschoal Gasparly L. (2002) *DÓRIS - Pedagogical Agent in Intelligent Tutoring Systems*. *Intelligent Tutoring Systems : 6th International Conference, ITS 2002, Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, pp 91-104
- [81] Vivas J. (2004), *Método Distsem: procedimiento para la evaluación de distancias semánticas*. *Revista Perspectivas en Psicología*, 1:56-61.
- [82] VanLehn K.(1988) *Student modeling*. En *Foundations of Intelligent Tutoring Systems*, eds.: Polson M.C. y Richardson J.J. Lawrence Erlbaum.
- [83] Wenger E. (1987). *Artificial Intelligence and Tutoring Systems*. Los Altos. CA: Morgan Kauffmann.

- [84] Webb N. (1982) *Student Interaction and Learning in Small Group*. Review of Educational Research, 52, pp. 421-445.
- [85] Wooldridge, M., y Jennings, N. R. (1995). *Intelligent Agents - Theories, Architectures, and Languages*. LNAI-Verlag. ISBN 3-540-58855-8 890.
- [86] Young R. M., Gordon D. Plotkin, R. F. Linz: *Analysis of an Extended Concept-Learning Task*. IJCAI 1977: 348
- [87] Zabala A. (1993). *Los enfoques didácticos*, en Coll, Salvador y otros El constructivismo en el aula . Barcelona.