

Universidad de las Ciencias Informáticas.



Empleo de las redes bayesianas en la
modelación de un algoritmo inteligente
para los módulos de evaluación
de los software educativos.



Trabajo de Diploma para optar por el
Título de Ingeniero Informático.

Autores:

*Misleidy Lairse Luján Herrera.
Leonel Rodríguez Espinosa.*

Tutor:

Msc. Ing. Ernesto Antonio González Díaz

*Ciudad de La Habana
Diciembre 2005*

RESUMEN.

El título del trabajo desarrollado es “El empleo de las redes bayesianas en la modelación de un algoritmo inteligente para los módulos de evaluación de los software educativos”.

En muchas de las ramas del conocimiento constantemente se está clasificando. Los procesos de clasificación son procesos que pueden ser complejos y engorrosos si se realizan de forma manual, es por ello que existen varios algoritmos cibernéticos que permiten modelar determinados procesos de clasificación pero no con la eficiencia requerida. Actualmente no existe ningún algoritmo que permita desarrollar procesos de clasificación probabilísticos de manera eficiente de acuerdo a los resultados.

El diagnóstico es uno de los procesos más importantes dentro de cualquier Sistema de Tutor Inteligente (STI) y por ello el modelado del alumno es un problema central en el diseño y desarrollo, por lo que el sistema debe ser capaz de determinar con la mayor precisión y rapidez posible cuál es el estado cognitivo del alumno, es decir, qué partes del dominio que se pretende enseñar ya domina y cuáles son las que aún desconoce.

El objetivo concreto es proponer un algoritmo que permitirá desarrollar de manera automática diferentes procesos de clasificación que desarrollados de forma manual son muy engorrosos e inexactos, además como es un algoritmo genérico podrá ser empleados en múltiples procesos de este tipo, independiente del objeto a clasificar.

Con este modelo se abre la posibilidad de obtener una evaluación más completa de los alumnos que la que se obtenía con los test basados en la Teoría de Respuesta al Ítem (TRI), lo cual hace que este tipo de evaluación mediante test fuese muy adecuada para integrarla con un Sistema Tutor Inteligente y por ende a los software educativos que ya se están produciendo en nuestras casas de software.

ÍNDICE DE CONTENIDO.

INTRODUCCIÓN.....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	4
1.1 – Introducción.....	4
1.2 – Informática Educativa.....	4
1.2.1 – Qué es la Informática Educativa.....	6
1.2.2 – Beneficios que aporta la Informática Educativa.....	8
1.3 – Software Educativo.....	10
1.3.1 – Conceptualización.....	10
1.3.2 – Características esenciales de los Programas Educativos.....	11
1.3.3 – Clasificación de los Programas Didácticos.....	12
1.3.3.1– Programas tutoriales.....	12
1.3.3.2 – Simuladores.....	13
1.3.3.3 – Constructores.....	14
1.3.3.4 – Programas herramientas.....	15
1.3.3.5 – Programas de presentación-demostración.....	16
1.3.3.6 – Sistemas de diálogo.....	16
1.3.3.7 – Juegos educativos.....	16
1.3.3.8 – Ejercicios y prácticas por computadora.....	17
1.3.4 – Funciones del Software Educativo.....	17
1.3.5 – Estructuras de los Programas Educativos.....	20
1.4 – Entrenadores.....	22
1.4.1 – Estructura de los entrenadores.....	22
1.4.2 –Test o Evaluador.....	25
1.5 – Conclusiones.....	26

CAPÍTULO 2: ANÁLISIS DE LAS TECNOLOGÍAS Y TENDENCIAS ACTUALES DE LA INTELIGENCIA ARTIFICIAL.....	27
2.1 – Introducción.	27
2.2 – La Inteligencia Artificial.	27
2.2.1 – Un poco de Historia.....	27
2.2.2 – Qué es la Inteligencia Artificial.....	28
2.3 – Qué es la Inteligencia Artificial Conectivista.	31
2.3.1 – Qué son las Redes Bayesianas.....	32
2.3.2 – Definición formal de las redes bayesianas.....	32
2.3.3 – Aprendizaje Bayesiano.....	34
2.3.4 – Aplicaciones y ejemplos.....	35
2.4 – Qué son las Redes de Markov.	38
2.4.1 – Campos de Markov.....	38
2.4.2 – Parametrización de las Redes de Markov.....	38
2.4.3 – Algoritmo.....	39
2.4.4 – Aplicaciones.....	40
2.5 – Qué son los Modelos Ocultos de Markov.	40
2.5.1 – Cadenas de Markov.....	41
2.5.2 – Definición de Modelos Ocultos de Markov.....	42
2.5.3 – Arquitecturas de HMMs.....	42
2.6 – Conclusiones.	44
CAPÍTULO 3: PROPUESTA DEL ALGORITMO.....	45
3.1 – Introducción.	45
3.2 – Modelado del alumno.	46
3.3 – Diagnóstico basado en el modelo de la Teoría de Respuesta al Ítem (TRI)	46
3.3.1 – Tests adaptativos informatizados.....	47
3.3.2 – Teoría de respuesta al ítem.....	49

3.3.3 – Estimación del nivel de habilidad.....	52
3.3.4 – Banco de ítems.	53
3.4 – Diagnóstico mediante redes bayesianas en los TAI.....	53
3.4.1 – Modelo estructural.....	54
3.4.1.1 – Diseño de la red bayesiana.	58
3.4.2 – Especificación de parámetros.....	60
3.4.3 – Algoritmo de la red bayesiana.	63
3.5 – Elementos básicos que deben estar presente en el desarrollo de la propuesta.....	66
3.5.1 – Captura de Requerimientos.....	66
3.5.2 – Casos de uso.	68
3.5.3 – Elementos de los casos de uso.	69
3.5.4 – Otras herramientas de la notación UML.	69
3.5.5 – Modelo de casos de uso del sistema.....	70
3.5.6 – Expansión de los Casos de Uso.	73
3.6 – Conclusiones.	78
CONCLUSIONES.....	79
RECOMENDACIONES.....	80
REFERENCIAS BIBLIOGRÁFICAS.....	81
BIBLIOGRAFÍA.....	85
GLOSARIO DE TÉRMINOS.	90

ÍNDICE DE FIGURAS.

Figura 1.1: <i>Estructura de un entrenador.</i>	23
Figura 2.1: <i>Representación de las Redes de Markov.</i>	38
Figura 2.2: <i>Cliques de una Red de Markov.</i>	39
Figura 3.1: <i>Diagrama de flujo de un test adaptativo.</i>	48
Figura 3.2: <i>CCIs correspondientes a preguntas con diferentes niveles de dificultad.</i>	50
Figura 3.3: <i>CCIs correspondientes a preguntas con diferentes factores de discriminación.</i>	51
Figura 3.4: <i>CCI con factores de adivinanza y distracción no nulos.</i>	51
Figura 3.5: <i>Red bayesiana como la concibe Millán.</i>	54
Figura 3.6: <i>Alternativas de las relaciones de agregación.</i>	58
Figura 3.7: <i>Diseño de la red bayesiana.</i>	59
Figura 3.8: <i>Relaciones concepto-tema.</i>	61
Figura 3.9: <i>Obtención de las probabilidades condicionadas.</i>	62
Figura 3.10: <i>Algoritmo de la red bayesiana en general.</i>	63
Figura 3.11: <i>Algoritmo de la red bayesiana, tests inicial.</i>	64
Figura 3.12: <i>Algoritmo de la red bayesiana, preguntas de contenido.</i>	65
Figura 3.13: <i>Diagrama de casos de uso del sistema.</i>	71
Figura 3.14: <i>Diagrama de casos de uso del profesor.</i>	72

INTRODUCCIÓN.

El desarrollo científico-técnico alcanzado en la época actual ha puesto en manos de la sociedad nuevas tecnologías que incrementan la productividad y el bienestar del hombre. Entre ellas se destaca la computadora, la que ha pasado a ocupar un lugar importante en todas las esferas de la vida social.

La computadora desarrolla un papel preponderante en la educación y la investigación, percibiéndose sus implicaciones futuras, así como sus ventajas como herramienta en el proceso de enseñanza y aprendizaje. Por su parte el software educativo es uno de los pilares del sistema educativo a distancia y se perfila como la herramienta base de las próximas generaciones de educandos, pero aún hoy día presenta algunas deficiencias con respecto a la evaluación del alumno.

Problema a resolver: Los procesos de clasificación del conocimiento del alumno son procesos que pueden ser complejos y engorrosos si se realizan de forma manual, es por ello que surge la necesidad de hacerlos de forma automática, para lograr mejoras en la evaluación de los estudiantes en los softwares educativos.

Actualidad: Actualmente los algoritmos que desarrollan procesos de clasificación probabilísticos no permiten obtener resultados de manera eficiente. El algoritmo que se propone en este trabajo permitirá desarrollar de manera automática diferentes procesos de clasificación que desarrollados de forma manual son muy engorrosos e inexactos, además como es un algoritmo genérico podrá ser empleado en múltiples procesos de este tipo independiente del objeto a clasificar.

Antecedentes: La realización de este trabajo se centra en los estudios teóricos de la doctora Eva Millán, la cual ha realizado un estudio bastante profundo sobre el empleo de redes bayesianas en los Test Adaptativos Informatizados (TAI) aunque presenta algunas deficiencias que se tratan de subsanar con la propuesta presentada.

Aporte teórico: Un modelo novedoso sobre los procesos de clasificación aproximados empleando redes bayesianas. Un algoritmo genérico para los diferentes procesos de clasificación aproximada.

Objeto de estudio: Inteligencia artificial conectivista, redes bayesianas, redes de Markov, aprendizaje automático, softwares educativos.

Campo de acción: Algoritmos de clasificación probabilísticos para los módulos de evaluación de los softwares educativos.

Idea a defender: Si existiese un algoritmo genérico que permita desarrollar procesos de clasificación probabilísticos en los módulos de evaluación de los softwares educativos de manera eficiente, los procesos docentes para los cuales han sido diseñados estos softwares se podrán desarrollar con más calidad permitiendo un nivel más personalizado en la atención de los estudiantes.

Objetivo general: Proponer un algoritmo que permita desarrollar procesos de clasificación probabilísticos en los módulos de evaluación de los softwares educativos de manera eficiente. Además que pueda ser empleado en múltiples procesos de este tipo independiente del objeto a clasificar debido a su carácter genérico.

Del análisis del objetivo general se derivan los siguientes *Objetivos específicos*:

- Realizar estudio del problema planteado mediante una búsqueda bibliográfica.
- Analizar el estado del arte.
- Seleccionar la red conectivista más idónea para la realización del algoritmo.
- Utilizar la notación UML para la representación del algoritmo.

Para lograr los objetivos propuestos se han trazado una serie de *tareas a desarrollar*, entre las que se encuentran:

- Fundamentación del problema.
- Valoración de las diferentes técnicas de la inteligencia artificial.

- Selección y fundamentación de la técnica a emplear para la realización del algoritmo inteligente.
- Representación del algoritmo propuesto utilizando la notación UML.

El presente trabajo está compuesto por tres capítulos: Fundamentación teórica, análisis de las tecnologías y tendencias actuales de la inteligencia Artificial y Propuesta del algoritmo.

El *Capítulo 1: Fundamentación Teórica*, trata sobre la informática educativa, los beneficios que aporta a los alumnos y profesores. Incluye además el estudio de los softwares educativos, concepto, características esenciales, clasificación, así como las funciones y la estructura de los programas educativos. Además se hace énfasis en los test o evaluadores dentro de los entrenadores; programas que proponen medir el nivel de los conocimientos o habilidades que posee un estudiante sobre una determinada materia.

El *Capítulo 2: Análisis de las tecnologías y tendencias actuales de la inteligencia artificial*, donde se presentan conceptos y aplicaciones de algunas de las disciplinas que forman parte de las múltiples ramas existentes en la Inteligencia Artificial. Además de definir qué es la Inteligencia Artificial, se hace un estudio detallado sobre los temas referentes a la Inteligencia Artificial Conectivista donde se encuentran las Redes Bayesianas, las Redes de Markov y los Modelos Ocultos de Markov. Este estudio fue necesario para la toma de decisión para elegir la red con la que se elaborará el diseño del algoritmo que se quiere obtener en esta investigación.

Capítulo 3: Propuesta del algoritmo, se exponen todos aquellos conceptos que se consideran necesarios para la comprensión de la solución propuesta, es decir, se habla brevemente del modelado de alumno, de los test adaptativos informatizados o computerizados (TAI), diagnóstico mediante redes bayesianas en los TAI para lo cual se describe cual es el modelo estructural de la propuesta, el diseño de la red bayesiana así como la especificación de los parámetros. En esta misma sección se discuten diferentes aproximaciones para simplificar el problema de la obtención de los parámetros, dependiendo del tipo de relaciones que se quieran lograr, además del desarrollo de la propuesta con ayuda de UML, destacando que este último aspecto se desarrolla para esclarecer la solución desarrollada.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.

1.1 – Introducción.

La informática es la ciencia que se dedica al tratamiento automatizado de la información, entendiéndose por información, un concepto, un hecho, una regla, un suceso, un conjunto de datos dentro de un contexto dado, un modelo matemático, etc.

El extraordinario desarrollo que se ha experimentado recientemente la informática y sus aplicaciones es quizás el centro del cambio tecnológico que vive nuestra época. El amplio proceso de digitalización, la fibra óptica, las computadoras y el Internet son las manifestaciones más visibles de este cambio que afecta significativamente todos los procesos de comunicación y manejo de información. Además el desarrollo de la Multimedia, el Hipertexto y la Hipermedia ha permitido la elaboración y explotación de softwares con las facilidades que la combinación de textos, sonidos, imágenes y animaciones pueden contribuir al procesamiento de la información en diferentes campos. Cada día estas técnicas se convierten en un instrumento eficaz de las comunicaciones y el acceso a la información. Entre los principales campos de acción se encuentra la educación, que será objeto de estudio de este trabajo.

Este capítulo aborda varios temas como: la informática educativa y los beneficios que aporta a los alumnos y profesores. Incluye además el estudio de los softwares educativos, concepto, características esenciales, clasificación, así como las funciones y la estructura de los programas educativos. Además se hace énfasis en los test o evaluadores dentro de los entrenadores; programas que proponen medir el nivel de los conocimientos o habilidades que posee un estudiante sobre una determinada materia.

1.2 – Informática Educativa.

Al mirar atrás se puede percatar que la obra que se ha logrado por la humanidad ha sido producto de la inteligencia, la creatividad y la voluntad del hombre. No es posible desconocer, en los albores del nuevo siglo, que ese potencial no puede dejarse al azar y hay que lograr que los nuevos cursos propicien

precisamente ese potencial humano en aras de elevar el desarrollo de la humanidad con las exigencias tecnológicas, sociales y económicas que el siglo XXI nos depara.

En la década de los cincuenta se dieron los primeros pasos en lo que hoy se denomina "Informática Educativa", cuando se desarrollaron las primeras aplicaciones y modelos de lo que entonces se llamó "Instrucción Asistida por Computadora" (CAI, por sus siglas en inglés). Actualmente, luego de varias décadas de un intenso desarrollo tecnológico, se ha revolucionado el potencial educativo de las aplicaciones informáticas. De manera paralela, los medios electrónicos de comunicación: la radio y la televisión, se han transformado en sus tecnologías y aplicaciones. La televisión satelital y por cable pueden ofrecer hoy día una gran diversidad de programas educativos y culturales, con nuevos formatos y lenguajes que tienen mayor eficacia y versatilidad; además, mediante el sistema de compresión digital, pueden transmitir páginas electrónicas, discos compactos, lo mismo que gran cantidad de datos e imágenes.[27]

El objetivo fundamental del pensamiento actual en Informática Educativa es hacer que el construir y el aprender sean visibles y la tecnología sea invisible, que lo importante sea la tarea de aprendizaje y no la tecnología. Que esta última sea un buen taller donde se construyan cosas interesantes, aprendizajes constructivos y significativos.

Diversas experiencias y evaluaciones de impacto apuntan a la conclusión de que las nuevas tecnologías tienen resultados muy positivos para los alumnos:

- Incrementan su motivación y creatividad.
- Fomentan el espíritu de indagación y una mentalidad orientada a la solución de problemas.
- Facilitan una aproximación interdisciplinaria al conocimiento y generan así un aprendizaje más integrado y mejor asimilado.
- Promueven la cooperación y el trabajo en grupo.
- Propician que los alumnos se sientan capaces de generar conocimiento valioso, y puedan compartirlo mediante su publicación.
- Permiten la adquisición de nuevas habilidades y lenguajes.

Todo ello redundará en un fortalecimiento de la autoestima de los alumnos, particularmente en la de aquellos que pudieran tener un desempeño desfavorable para incorporarse al nivel del grupo. [27]

1.2.1 – Qué es la Informática Educativa.

En los últimos años han venido surgiendo una serie de conceptos dentro del campo de la tecnología educativa. En ocasiones los conceptos de medios y tecnologías se confunden.

Los medios de enseñanza son: todo componente material o materializado del proceso docente, que en función del método sirve para construir representaciones esenciales de contenido-forma, es aquello que da significado y sentido a la actividad de estudio de los conocimientos, habilidades y valores que se expresan en el objetivo.

La tecnología educativa no solo son los medios sino además al conjunto de informaciones que existen sobre cómo aplicar eficientemente cada medio, la relación Problema-Objetivo-Contenido, la relación Objetivo-Contenido (Método-Medio), la organización de los contenidos en función de los objetivos y la selección adecuada de los métodos y medios.

La Informática Educativa es una disciplina que estudia el uso, efectos y consecuencias de las tecnologías de la información y el proceso educativo. Esta disciplina intenta acercar al aprendiz al conocimiento y manejo de modernas herramientas tecnológicas como la computadora y de cómo el estudio de estas tecnologías contribuyen a potenciar y expandir la mente, de manera que los aprendizajes sean más significativos y creativos. El desafío que presenta la informática educativa en el sector educativo será la aplicación racional y pertinente de las nuevas tecnologías de la información en el desarrollo del quehacer educativo. [25]

Como se puede notar la informática educativa es parte de la tecnología educativa y los conceptos de ambas ciencias incluyen elementos que van más allá del concepto de medios de enseñanza-aprendizaje.

La informática educativa se materializa en:

- Tareas de evaluación y selección de software educativo.
- Tareas de diseño y elaboración de software educativo.
- Estudios de los diferentes usos de la computadora.
- Recursos materiales (hardware y software).
- Formación de recursos humanos, a luz del nuevo papel que debe jugar el profesor.
- Fundamentos pedagógicos de la enseñanza asistida por computadora.
- Didáctica del estudio de la informática.
- Modelo de inserción de la informática en una disciplina o asignatura.
- Fundamentos de la educación a distancia.

Varios expertos coinciden en señalar tres variantes para el desarrollo de la informática en el proceso docente. Estas variantes son:

- 1- *Como objeto de estudio.* Esta es la modalidad donde el estudiante aprende y desarrolla habilidades en el manejo de la computadora, se aplica fundamentalmente desde la disciplina computación o informática. Aunque es posible que en otra disciplina un profesor necesite que los estudiantes desarrollen determinadas habilidades en el manejo de programas específicos.
- 2- *Como medio de enseñanza.* En esta variante la computadora y los programas se convierten en medios de enseñanza–aprendizaje, por la información que transmiten, el cómo la transmiten, el apoyo que representan a la hora de impartir la clase. Cuando se produce un software con el fin de ser utilizado en la docencia, este se convierte, por su concepción y su diseño en un medio de enseñanza.
- 3- *Como herramienta de trabajo.* Aquí la computadora es empleada tanto por el profesor como por los alumnos con el fin de resolver un problema específico de la especialidad o de la asignatura. En estos casos se trabaja por ejemplo en la búsqueda de información en la red, comunicarse por correo o chat con el profesor, otros estudiantes o colegas, se trabaja con aplicaciones ofimáticas o aplicaciones específicas como por ejemplo un programa para resolver los cálculos de gabinete de un levantamiento topográfico, procesar el texto de un informe, presentar un informe, tabular y graficar datos, etc.

Ninguna de estas tres variantes están reñidas entre sí, por el contrario se complementan. [25]

La computadora y los softwares educativos, como medios de enseñanza, resultan un eficiente auxiliar del profesor en la preparación e impartición de las clases ya que contribuyen a una mayor ganancia metodológica y a una racionalización de las actividades del profesor y los alumnos.

Se concluye entonces que la Informática Educativa es concebida como la “sinergia entre la educación y la informática, donde cada una de estas ciencias aporta sus más excelsos beneficios en una relación ganar-ganar”. [3]

1.2.2 – Beneficios que aporta la Informática Educativa.

Incorporar la informática en la docencia trae como consecuencia múltiples beneficios entre los que se encuentran:

- Las operaciones automáticas pueden librar al individuo para acometer tareas conceptuales más importantes.
- Los estudiantes medios y más débiles reciben estímulos importantes al percibir que no deben ser brillantes manipuladores algebraicos para manipular el pensamiento abstracto.
- El estudio de los algoritmos subyacentes ayudan a entender la naturaleza de las operaciones.
- El permitir al usuario construir operaciones más complejas de las habituales se traduce en el mejor entendimiento conceptual.
- Trabajar con la computadora dota al estudio del factor experimental, lo que lleva al establecimiento de conjeturas, simulaciones, ejemplos y contraejemplos, etc.

Otra característica es que la computadora, a diferencia del profesor, no manifiesta impaciencia alguna al cometerse errores repetidamente por los estudiantes.

El uso de la computadora, y por ende de los softwares educativos, permite agrupar una serie de factores presentes en otros medios, pero a la vez agregar otros hasta ahora inalcanzables.

- Permite la interactividad con los estudiantes, retroalimentándolos, y evaluando lo aprendido, a través de ella se puede demostrar el problema como tal.
- Facilita las representaciones animadas.
- Incide en el desarrollo de las habilidades a través de la ejercitación. Permite simular procesos complejos, peligrosos o muy abstractos.
- Reduce el tiempo de que se dispone para impartir gran cantidad de conocimientos.
- Facilita el trabajo independiente y la atención a las diferencias individuales.
- Permite al usuario (estudiante o profesor) introducirse en las técnicas más avanzadas.

Por otra parte se puede afirmar que la computadora desempeña un papel importante en la racionalización del trabajo de profesores y alumnos en la educación superior, influyendo positivamente sobre la calidad del proceso a partir de las siguientes ventajas:

- La matematización y algoritmización del contenido de las disciplinas.
- La reducción del tiempo de transmisión y asimilación de los contenidos.
- La posibilidad de estudiar procesos que no es posible observar directamente.
- La representación visual del objeto estudiado.
- La interacción constante entre la fuente de información y el estudiante, lo que permite el análisis de múltiples alternativas de solución.
- Motivación de los alumnos.
- Información inmediata que posibilita al alumno volver sobre sus pasos.
- La automatización de los experimentos científicos que desarrollan hábitos y habilidades en el diseño de experimentos y el procesamiento de la información de forma interactiva.
- El desarrollo de hábitos y habilidades profesionales en el trabajo con sistemas automatizados de proyectos y de procesos tecnológicos.

La tendencia actual en el empleo de la computadora en la enseñanza, se está orientando cada vez más a las dos últimas variantes, debido a que los conocimientos y habilidades informáticas, que el estudiante necesita desarrollar, deben ser las de un usuario común, pero que esté preparado para buscar, adquirir y construir un conocimiento, buscar la solución a un problema, de determinada complejidad, emplear las

potencialidades de la computadora para comunicarse y establecer relaciones personales y profesionales basadas en el respeto mutuo.

1.3 – Software Educativo

El software educativo es uno de los pilares del sistema de educación a distancia y se perfila como la herramienta base de las próximas generaciones de educandos. Las mismas técnicas que utiliza el software educativo se pueden aplicar al desarrollo de sistemas utilizables en otras áreas para facilitar el manejo de terminales por usuarios no informáticos.

1.3.1 – Conceptualización.

Si se habla de computación e informática no se puede dejar de hablar de software puesto que sin éste la máquina no serviría de nada. Los softwares son programas de computadoras. Son las instrucciones responsables de que el hardware (la máquina) realice su tarea. Como concepto general, el software puede dividirse en varias categorías basadas en el tipo de trabajo realizado.

Los softwares desde el punto de vista genérico pueden clasificarse en

- Sistemas operativos.
- Aplicaciones de propósito específico.
- Aplicaciones de propósito general.
- Lenguajes de programación.

Los Softwares Educativos engloban todos los programas que han estado elaborados con fin didáctico, desde los tradicionales programas basados en los modelos conductistas de la enseñanza, los programas de Enseñanza Asistida por Computadora, hasta los aún programas experimentales de Enseñanza Inteligente Asistida por Computadora, que, utilizando técnicas propias del campo de los Sistemas Expertos y de la Inteligencia Artificial en general, pretenden imitar la labor tutorial personalizada que

realizan los profesores y presentan modelos de representación del conocimiento en consonancia con los procesos cognitivos que desarrollan los alumnos.

No obstante según esta definición, más basada en un criterio de finalidad que de funcionalidad, se excluyen del software educativo todos los programas de uso general en el mundo empresarial que también se utilizan en los centros educativos con funciones didácticas o instrumentales como por ejemplo: procesadores de textos, gestores de bases de datos, hojas de cálculo, editores gráficos. Estos programas, aunque puedan desarrollar una función didáctica, no han estado elaborados específicamente con esta finalidad.

1.3.2 – Características esenciales de los Programas Educativos.

Los programas educativos pueden tratar las diferentes materias (matemáticas, idiomas, geografía, dibujo), de formas muy diversas (a partir de cuestionarios, facilitando una información estructurada a los alumnos, mediante la simulación de fenómenos) y ofrecer un entorno de trabajo más o menos sensible a las circunstancias de los alumnos y más o menos rico en posibilidades de interacción; pero todos comparten cinco características esenciales:

- Son materiales elaborados con una finalidad didáctica, como se desprende de la definición.
- Utilizan la computadora como soporte en el que los alumnos realizan las actividades que ellos proponen.
- Son interactivos, contestan inmediatamente las acciones de los estudiantes y permiten un diálogo e intercambio de informaciones entre la computadora y los estudiantes.
- Individualizan el trabajo de los estudiantes, ya que se adaptan al ritmo de trabajo de cada uno y pueden adaptar sus actividades según las actuaciones de los alumnos.
- Son fáciles de usar. Los conocimientos informáticos necesarios para utilizar la mayoría de estos programas son similares a los conocimientos de electrónica necesarios para usar un video, es decir, son mínimos, aunque cada programa tiene unas reglas de funcionamiento que es necesario conocer.

1.3.3 – Clasificación de los Programas Didácticos.

Los programas educativos a pesar de tener unos rasgos esenciales básicos y una estructura general común se presentan con unas características muy diversas: unos aparentan ser un laboratorio o una biblioteca, otros se limitan a ofrecer una función instrumental del tipo máquina de escribir o calculadora, otros se presentan como un juego o como un libro, bastantes tienen vocación de examen, unos pocos se creen expertos... y, por si no fuera bastante, la mayoría participan en mayor o menor medida de algunas de estas peculiaridades. Para poner orden a esta disparidad, se han elaborado múltiples tipologías que clasifican los programas didácticos a partir de diferentes criterios.

Una clasificación interesante de los programas atiende a la posibilidad de modificar los contenidos del programa y distingue entre programas cerrados (que no pueden modificarse) y programas abiertos, que proporcionan un esqueleto, una estructura, sobre la cual los alumnos y los profesores pueden añadir el contenido que les interese. De esta manera se facilita su adecuación a los diversos contextos educativos y permite un mejor tratamiento de la diversidad de los estudiantes.

No obstante, de todas las clasificaciones existentes la que posiblemente proporciona categorías más claras y útiles a los profesores es la que tiene en cuenta el grado de control del programa sobre la actividad de los alumnos y la estructura de su algoritmo dentro de los que se encuentran: Programas tutoriales, Simuladores, Constructores, Programas herramientas, Programas de presentación-demostración, Sistemas de diálogo, Juegos educativos, Ejercicios y prácticas por computadora. [18]

1.3.3.1– Programas tutoriales.

Son programas que en mayor o menor medida dirigen, tutorizan, el trabajo de los alumnos. Pretenden que, a partir de unas informaciones y mediante la realización de ciertas actividades previstas de antemano, los estudiantes pongan en juego determinadas capacidades y aprendan o refuercen unos conocimientos y/o habilidades. Cuando se limitan a proponer ejercicios de refuerzo sin proporcionar explicaciones conceptuales previas se denominan programas tutoriales de ejercitación, como es el caso de los programas de preguntas y de los programas de adiestramiento psicomotor, que desarrollan la

coordinación neuromotriz en actividades relacionadas con el dibujo, la escritura y otras habilidades psicomotrices. A partir de la estructura de su algoritmo, se distinguen cuatro categorías:

- **Programas lineales**, que presentan al alumno una secuencia de información y/o ejercicios (siempre la misma o determinada aleatoriamente) con independencia de la corrección o incorrección de sus respuestas. No obstante, su interactividad resulta pobre y el programa se hace largo de recorrer.
- **Programas ramificados**. Basados inicialmente también en modelos conductistas, siguen recorridos pedagógicos diferentes según el juicio que hace la computadora sobre la corrección de las respuestas de los alumnos o según su decisión de profundizar más en ciertos temas.
- **Entornos tutoriales**. En general están inspirados en modelos pedagógicos cognitivistas, y proporcionan a los alumnos una serie de herramientas de búsqueda y de proceso de la información que pueden utilizar libremente para construir la respuesta a las preguntas del programa.
- **Sistemas tutoriales expertos**. Como los Sistemas Tutores Inteligentes, que elaborados con las técnicas de la Inteligencia Artificial y teniendo en cuenta las teorías cognitivas sobre el aprendizaje, tienden a reproducir un diálogo auténtico entre el programa y el estudiante, y pretenden comportarse como lo haría un tutor humano.

1.3.3.2 – Simuladores.

Presentan un modelo o entorno dinámico (generalmente a través de gráficos o animaciones interactivas) y facilitan su exploración y modificación a los alumnos, que pueden realizar aprendizajes inductivos o deductivos mediante la observación y la manipulación de la estructura subyacente; de esta manera pueden descubrir los elementos del modelo, sus interrelaciones, y pueden tomar decisiones y adquirir experiencia directa delante de unas situaciones que frecuentemente resultarían difícilmente accesibles a la realidad (control de una central nuclear, contracción del tiempo, pilotaje de un avión...). Se pueden diferenciar dos tipos de simulador:

- **Modelos físico-matemáticos:** Presentan de manera numérica o gráfica una realidad que tiene unas leyes representadas por un sistema de ecuaciones deterministas. Se incluyen aquí los programas-laboratorio, algunos trazadores de funciones y los programas que mediante un convertidor analógico-digital captan datos analógicos de un fenómeno externo a la computadora y presentan en pantalla un modelo del fenómeno estudiado o informaciones y gráficos que van asociados.
- **Entornos sociales:** Presentan una realidad regida por unas leyes no del todo deterministas. Se incluyen aquí los juegos de estrategia y de aventura, que exigen una estrategia cambiante a lo largo del tiempo.

1.3.3.3 – Constructores.

Son programas que tienen un entorno programable. Facilitan a los usuarios unos elementos simples con los cuales pueden construir elementos más complejos o entornos. De esta manera potencian el aprendizaje heurístico y, de acuerdo con las teorías cognitivistas, facilitan a los alumnos la construcción de sus propios aprendizajes, que surgen a través de la reflexión que realizan al diseñar programas y comprobar inmediatamente, cuando los ejecuten, la relevancia de sus ideas. El proceso de creación que realiza el alumno genera preguntas del tipo: ¿Qué sucede si añado o elimino el elemento X? Se pueden distinguir dos tipos de constructores:

- **Constructores específicos.** Ponen a disposición de los estudiantes una serie de mecanismos de actuación (generalmente en forma de órdenes específicas) que les permiten llevar a cabo operaciones de un cierto grado de complejidad mediante la construcción de determinados entornos, modelos o estructuras, y de esta manera avanzan en el conocimiento de una disciplina o entorno específico
- **Lenguajes de programación.** Como PASCAL, BASIC, C++ que ofrecen unos "laboratorios simbólicos" en los que se pueden construir un número ilimitado de entornos. Aquí los alumnos se convierten en profesores de la computadora. Además, con las interfaces convenientes, pueden controlar pequeños robots contruidos con componentes convencionales (arquitecturas, motores...), de manera que sus posibilidades educativas se ven ampliadas incluso en campos pre-tecnológicos.

1.3.3.4 – Programas herramientas.

Son programas que proporcionan un entorno instrumental con el cual se facilita la realización de ciertos trabajos generales de tratamiento de la información: escribir, organizar, calcular, dibujar, transmitir, captar datos.... A parte de los lenguajes de autor (que también se podrían incluir en el grupo de los programas constructores), los más utilizados son programas de uso general que provienen del mundo laboral y, por tanto, quedan fuera de la definición que se ha dado de software educativo. No obstante, se han elaborado algunas versiones de estos programas "para niños" que limitan sus posibilidades a cambio de una, no siempre clara, mayor facilidad de uso. Los programas más utilizados de este grupo son:

- **Procesadores de textos.** Son programas que, con la ayuda de una impresora, convierten la computadora en una fabulosa máquina de escribir. En el ámbito educativo debe hacerse una introducción gradual que puede empezar a lo largo de la Enseñanza Primaria, y ha de permitir a los alumnos familiarizarse con el teclado y con la computadora en general, y sustituir parcialmente la libreta de redacciones por un disco.
- **Gestores de bases de datos.** Sirven para generar potentes sistemas de archivo ya que permiten almacenar información de manera organizada y posteriormente recuperarla y modificarla. Entre las muchas actividades con valor educativo que se pueden realizar está revisar una base de datos ya construida para buscar determinadas informaciones y recuperarlas y recoger información, estructurarla y construir una nueva base de datos.
- **Hojas de cálculo.** Son programas que convierten la computadora en una versátil y rápida calculadora programable, facilitando la realización de actividades que requieran efectuar muchos cálculos matemáticos.
- **Editores gráficos.** Se emplean desde un punto de vista instrumental para realizar dibujos, portadas para los trabajos, murales, anuncios, etc. Además constituyen un recurso idóneo para desarrollar parte del currículum de Educación Artística: dibujo, composición artística, uso del color, etc.

1.3.3.5 – Programas de presentación-demostración.

Existen distintas clases de programas de demostración. Muchos suelen limitarse a presentar nueva información. Otros sirven para ejemplificar o ilustrar conceptos previamente estudiados. Aunque su estructura es a menudo secuencial, estos programas también ofrecen otras formas de presentación, con índices de actividades o menús de opciones disponibles.

La mayoría de estas aplicaciones son aptas para ser usadas directamente por el maestro al frente de la clase a modo de pizarrón electrónico. Las posibilidades que ofrece la computadora de incluir representaciones gráficas, con animación o movimiento, y crear efectos sonoros, junto con su gran rapidez y capacidad de cálculo, son recursos a tener en cuenta al evaluar el uso de lenguajes y sistemas para crear programas de demostración.

1.3.3.6 – Sistemas de diálogo.

Los sistemas de diálogo tienen dos modalidades. Cuando la computadora tiene la iniciativa, el programa presenta un cuestionario seleccionado al azar, entre las muchas preguntas que el sistema es capaz de formular a partir de unos datos básicos. En la segunda modalidad, el estudiante decide interrogar al programa sobre cualquier información y el sistema puede contestar a preguntas cuyas respuestas no hayan sido previstas, mediante inteligencia artificial.

1.3.3.7 – Juegos educativos.

Los juegos educativos tienen mucha aplicación en la Enseñanza Asistida por Computadora. El elemento lúdico suele convertir un ejercicio en un desafío motivador. El alumno considera a la computadora como un adversario al que puede ganar. Los juegos pueden tener también desventajas. Tanto el alumno como el maestro pueden confiar demasiado en el funcionamiento automático de las actividades como mecanismo de aprendizaje. Para evitar el riesgo de un aprendizaje poco profundo, es conveniente que las clases incluyan actividades adicionales para consolidar lo aprendido por medio de la computadora. Un

ejemplo es el programa "La máquina del tiempo" en el que los niños conocen las formas de vida de diversas culturas a lo largo de la historia.

1.3.3.8 – Ejercicios y prácticas por computadora.

Existe un campo de aplicación muy amplio para las prácticas y ejercicios realizados por computadora, siempre y cuando estos programas se adapten flexiblemente a la secuencia y metodología de cada maestro. La ejercitación de ciertas habilidades es siempre imprescindible. La informática tiene planteado un desafío en la renovación de los métodos de enseñanza. Los ejercicios de asociación o emparejamiento, por ejemplo, pueden ser sencillos y eficaces como forma de trabajo.

Este formato de ejercicio también sirve para realizar actividades de descubrimiento, como forma de presentación que permite al alumno descubrir información o establecer nuevas relaciones. Pero corresponderá siempre al maestro determinar si el ejercicio justifica el uso de la computadora, o si por el contrario, su realización sería más práctica utilizando medios convencionales como papel, lápiz, pizarrón, voz viva, material impreso, etc. Para tomar esta decisión, el maestro deberá tener en cuenta las principales ventajas de un sistema de cómputo, que pueden ser aplicables o no, en función de cada objetivo de aprendizaje. [18]

1.3.4 – Funciones del Software Educativo.

Los programas didácticos, cuando se aplican a la realidad educativa, realizan las funciones básicas propias de los medios didácticos en general y además, en algunos casos, según la forma de uso que determina el profesor, pueden proporcionar funcionalidades específicas.

Por otra parte, como ocurre con otros productos de la actual tecnología educativa, no se puede afirmar que el software educativo por sí mismo sea bueno o malo, todo dependerá del uso que de él se haga, de la manera cómo se utilice en cada situación concreta. En última instancia su funcionalidad y las ventajas e inconvenientes que pueda comportar su uso serán el resultado de las características del material, de su

adecuación al contexto educativo al que se aplica y de la manera en que el profesor organice su utilización. Entre las funciones que pueden realizar los programas están:

1. **Función informativa.** La mayoría de los programas a través de sus actividades presentan unos contenidos que proporcionan una información estructuradora de la realidad a los estudiantes. Como todos los medios didácticos, estos materiales representan la realidad y la ordenan. Los programas tutoriales, los simuladores y, especialmente, las bases de datos, son los programas que realizan más marcadamente una función informativa.

2. **Función instructiva.** Todos los programas educativos orientan y regulan el aprendizaje de los estudiantes ya que, explícita o implícitamente, promueven determinadas actuaciones de los mismos encaminadas a facilitar el logro de unos objetivos educativos específicos. Además condicionan el tipo de aprendizaje que se realiza pues, por ejemplo, pueden disponer un tratamiento global de la información (propio de los medios audiovisuales) o a un tratamiento secuencial (propio de los textos escritos).
Con todo, si bien la computadora actúa en general como mediador en la construcción del conocimiento y el metaconocimiento de los estudiantes, son los programas tutoriales los que realizan de manera más explícita esta función instructiva, ya que dirigen las actividades de los estudiantes en función de sus respuestas y progresos.

3. **Función motivadora.** Generalmente los estudiantes se sienten atraídos e interesados por todo el software educativo, ya que los programas suelen incluir elementos para captar la atención de los alumnos, mantener su interés y, cuando sea necesario, focalizarlo hacia los aspectos más importantes de las actividades. Por tanto la función motivadora es una de las características de este tipo de materiales didácticos, y resulta extremadamente útil para los profesores.

4. **Función evaluadora.** La interactividad propia de estos materiales, que les permite responder inmediatamente a las respuestas y acciones de los estudiantes, les hace especialmente adecuados para evaluar el trabajo que se va realizando con ellos. Esta evaluación puede ser de dos tipos:

- *Implícita*, cuando el estudiante detecta sus errores, se evalúa, a partir de las respuestas que le da la computadora.
- *Explícita*, cuando el programa presenta informes valorando la actuación del alumno. Este tipo de evaluación sólo la realizan los programas que disponen de módulos específicos de evaluación.

5. **Función investigadora.** Los programas no directivos, especialmente las bases de datos, simuladores y programas constructores, ofrecen a los estudiantes interesantes entornos donde investigar: buscar determinadas informaciones, cambiar los valores de las variables de un sistema, etc. Además, tanto estos programas como los programas herramienta, pueden proporcionar a los profesores y estudiantes instrumentos de gran utilidad para el desarrollo de trabajos de investigación que se realicen básicamente al margen de los ordenadores.

6. **Función expresiva.** Dado que los ordenadores son unas máquinas capaces de procesar los símbolos mediante los cuales las personas representan sus conocimientos y se comunican, sus posibilidades como instrumento expresivo son muy amplias.

Desde el ámbito de la informática que se está tratando, el software educativo, los estudiantes se expresan y se comunican con la computadora y con otros compañeros a través de las actividades de los programas y, especialmente, cuando utilizan lenguajes de programación, procesadores de textos, editores de gráficos, etc.

Otro aspecto a considerar al respecto es que los ordenadores no suelen admitir la ambigüedad en sus "diálogos" con los estudiantes, de manera que los alumnos se ven obligados a cuidar más la precisión de sus mensajes.

7. **Función metalingüística.** Mediante el uso de los sistemas operativos y los lenguajes de programación los estudiantes pueden aprender los lenguajes propios de la informática.

8. **Función lúdica.** Trabajar con los ordenadores realizando actividades educativas es una labor que a menudo tiene unas connotaciones lúdicas y festivas para los estudiantes. Además, algunos programas

refuerzan su atractivo mediante la inclusión de determinados elementos lúdicos, con lo que potencian aún más esta función.

9. **Función innovadora.** Aunque no siempre sus planteamientos pedagógicos resulten innovadores, los programas educativos se pueden considerar materiales didácticos con esta función ya que utilizan una tecnología recientemente incorporada a los centros educativos y, en general, suelen permitir muy diversas formas de uso. Esta versatilidad abre amplias posibilidades de experimentación didáctica e innovación educativa en el aula. [18]

1.3.5 – Estructuras de los Programas Educativos.

Toda la aplicación multimedia estará conformada por pequeños módulos programados que solamente pueden describir acciones inmediatas, formando un verdadero sistema, así, el itinerario arbóreo que se recorre dependerá de los caminos previos recorridos por el usuario y por las decisiones que tome del marco hipertextual, por las características del entorno, por interacciones detectadas por la aplicación y por acciones que tomará el alumno de acuerdo a diversos eventos que se presente. Por lo tanto, el camino que llevará al alumno por diversos itinerarios no sólo estará definido por los hipervínculos en el que estudiante por voluntad propia intentará recorrer, sino por los diversos módulos (conectados en paralelo y en serie y bidireccionales), que mediante diversas estrategias establecidas se le presenta al alumno.

Al inicio de la aplicación, existe un proceso de identificación de las características del entorno (alumno+hardware+medios), de esta forma, la aplicación multimedia se adaptará y trabajará de acuerdo a las condiciones del entorno, donde también estará incluida una evaluación diagnóstica del entorno, contemplándose las características cognitivas del usuario y sus conocimientos previos, cabe aclarar que dicho proceso de evaluación se encuadra “como un proceso sistemático de recogida de información válida y fiable para formar juicios de valor sobre elementos y hechos educativos, de acuerdo con unos criterios, con el fin de valorarlos y tomar decisiones.” [7]

En consecuencia, lo que se persigue con el proceso de evaluación es recoger los datos necesarios para actuar en consecuencia.

Un software educativo esta compuesto por varios módulos dentro de los cuales se pueden citar:

1. **Módulo de presentación y de gestión de menús.** Comprende las pantallas de presentación y despedida del programa y las pantallas de gestión de los menús principales. La interfaz es el entorno a través del cual los programas establecen el diálogo con sus usuarios, y es la que posibilita la interactividad característica de estos materiales. Está integrada por dos sistemas:
 - *El sistema de comunicación programa-usuario:* Facilita la transmisión de informaciones al usuario por parte de la computadora, incluye las pantallas a través de las cuales los programas presentan información a los usuarios, el empleo de periféricos como: altavoces, sintetizadores de voz, robots, módems, convertidores digitales-analógicos etc.
 - *El sistema de comunicación usuario-programa:* Facilita la transmisión de información del usuario hacia la computadora, incluye el uso del teclado y el mouse (ratón), mediante los cuales los usuarios introducen en la computadora un conjunto de órdenes o respuestas que los programas reconocen así como otros periféricos como: micrófonos, lectores de fichas, teclados conceptuales, pantallas táctiles, lápices ópticos, módems, lectores de tarjetas, convertidores analógico-digitales.
2. **Módulo de actividades interactivas.** Contiene las diferentes actividades educativas que el programa puede presentar a los alumnos. Este módulo le permite al alumno o usuario por ejemplo interactuar con el micromundo diseñado para manejar los contenidos teóricos del programa educativo, a través de los ejercicios diseñados para tal efecto.
3. **Módulos de ayuda.** La ayuda de un programa es una de las principales fuentes de información y documentación sobre su funcionalidad y operatividad, de allí la importancia de conocer las facilidades que ésta brinda al usuario y saber utilizarla de la manera más eficiente. Hay que determinar las formas de acceso a estas ayudas, que pueden ser:
 - Ayudas sobre el funcionamiento del programa.
 - Ayudas didácticas, sobre los contenidos.
4. **Módulo de evaluación.** Gestiona el almacenamiento de información sobre las actuaciones de los alumnos y la posterior presentación de informes. Habrá que determinar las informaciones que son relevantes, cómo se accederá a ellas y cómo se presentarán.

5. **Módulos auxiliares.** Por ejemplo: gestión de posibles modificaciones de parámetros, utilidades para los alumnos (calculadora, diccionario...), etc. [7]

1.4 – Entrenadores.

Los softwares educativos deben evaluar a cada alumno de acuerdo a sus capacidades para saber el nivel de conocimiento que tiene y a partir de ahí presentar las preguntas que contribuyan al aprendizaje del estudiante pero esta atención diferenciada es muy difícil de lograr. A esto se le llama módulo de evaluación del alumno en los entrenadores.

Al software diseñado con el propósito de desarrollar una determinada habilidad se designa como *entrenador*, específicamente una habilidad manual o motora, en el estudiante que lo emplea. Muchos entrenadores utilizan la simulación de situaciones reales, en menor o mayor grado, con lo cual el estudiante puede entrenarse en la solución de tareas de diferentes grados de complejidad y los acerca a las formas cotidianas en que se presentan y resuelven determinados problemas. Existen diferentes tipos de entrenadores que van desde los más simples y lineales hasta aquellos que son capaces de identificar y caracterizar al estudiante que lo emplea y proponer una estrategia de entrenamiento de acuerdo a las características individuales de cada usuario. A este tipo de software algunos lo llaman repasador, especialmente cuando la habilidad a desarrollar es intelectual. Más adelante, en el capítulo 2, se hace referencia a algunos ejemplos de entrenadores existentes. [24]

Son usados fundamentalmente para desarrollar el tercer aspecto de la instrucción: la práctica. Esta fase es muy importante y en ella se desarrolla la fluidez y la soltura requerida como por ejemplo, habilidades básicas de cálculo, en el manejo del vocabulario o la ortografía, en la solución de problemas, etc.

1.4.1 – Estructura de los entrenadores.

La estructura y flujo general de un entrenador, al igual que un tutorial, tiene una sección introductoria, seguida de un ciclo que se repite muchas veces:

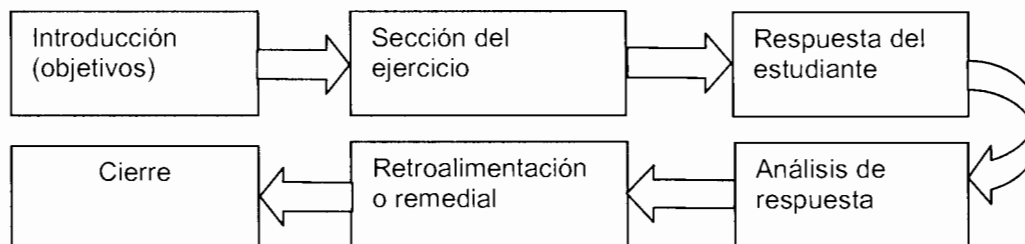


Figura 1.1: Estructura de un entrenador.

Nótese que aquí no existe la presentación de la información. A continuación se explican solo aquellos aspectos que son diferentes a los tutoriales, pues el resto son válidos aquí también, para ello se realiza el análisis de dos de las estructuras de un entrenador:

1. Etapa de Introducción.

En esta etapa es donde se introducen los contenidos y objetivos del software educativo, pero solo se hará referencia específicamente a lo relacionado con el control inicial del alumno.

Algunos entrenadores permiten que el estudiante defina el número de ejercicios que va a realizar, así como su nivel de dificultad, ahora bien, aunque esto suceda, se puede pensar si están realmente capacitados para tomar esta decisión, es ahí donde entra la importancia de la evaluación del alumno en los software educativos.

Otro control que puede existir es que el alumno seleccione qué tipo de información desea practicar.

2. Selección del ejercicio.

La selección de los ejercicios puede ser en un orden específico o de manera aleatoria, en cualquier caso, debe velarse porque el nivel de dificultad sea parejo.

En ocasiones la selección se hace a partir de la actuación del estudiante, de manera que se practiquen con más frecuencia aquellos ejercicios en los que se ha tenido más dificultad. Puede suceder que los ejercicios en los que no se ha tenido éxito, no se descarten, sino que se presenten, de acuerdo a un orden dado, en otro momento del entrenamiento.

Como una de las cosas que se enfatizan en un entrenador es lograr fluidez y soltura para contestar, muchas veces se toma en consideración el tiempo que se dará para entrar dicha respuesta, para definir un nivel de dificultad.

En otro orden de cosas, en un entrenador la ejercitación puede producirse a través de preguntas de selección múltiple, completar, o de respuestas cortas, etc. También existen respuestas del tipo pareja asociada, que son cualquier par de eventos o palabras relacionadas.

Debe tenerse en cuenta, además, que el tiempo de la sesión de entrenamiento debe estar en correspondencia con la edad a fin de evitar la fatiga y el aburrimiento. Para lograr esto, en general, si se tiene un tema muy amplio, debe dividirse en pequeñas sesiones de trabajo.

En forma general estos programas pueden tener incluidas simulaciones, como una forma poderosa de enseñar sobre algún aspecto del mundo imitándolo, el estudiante puede interactuar con el programa simulando situaciones reales.

Los estudiantes no sólo se motivan por la simulación, sino que también aprenden interactuando con ella de manera similar a como pudieran hacerlo en situaciones reales. En casi todos los casos, en la simulación también hay simplificación de la realidad al omitir o cambiar detalles. En este mundo simple los alumnos:

- Resuelven problemas.
- Aprenden procedimientos.
- Entienden las características de los fenómenos y como controlarlos.
- Aprenden que acciones tomar en ciertas situaciones, etc.

En cada caso el propósito es ayudar al estudiante a construir un modelo mental útil, de parte del mundo, darle una oportunidad de probarlo sin riesgos y eficientemente y que aprenda realizando actividades en un contexto similar al real.

1.4.2 –Test o Evaluador

Este tipo de programa se propone medir el nivel de los conocimientos o habilidades que posee un estudiante sobre una determinada materia. Las limitaciones en la comunicación en lenguaje natural reducen considerablemente la variedad de preguntas a realizar, por eso la mayor parte de los examinadores utilizan preguntas de verdadero o falso, de selección y de completamiento.

Se analizan los evaluadores de manera diferente a como se hizo con los anteriores tipos de software utilizados en la enseñanza, ya que en ellos hay que tomar dos decisiones importantes:

- La naturaleza y contenido de la prueba.
- La manera en la cual será aplicada.

Para la definición de la naturaleza de la prueba hay un conjunto de factores que se tienen en cuenta:

- Propósito de la prueba: puede ser utilizada para determinar qué no sabe el estudiante y decidir qué información se le debe suministrar.
- Los objetivos que cubre.
- Decidir si habrá o no retroalimentación.
- Definición de la calificación que determinará el nivel de aprobado o de pasar el test.
- Tipo de datos que será recogido: calificación final, respuestas individuales a cada pregunta, cambios de respuestas, solicitud de ayuda, etc.
- Presentación de los resultados: calificación, recomendaciones, etc.
- Otros factores pueden ser: si el número de preguntas satisface el propósito, tiempo de aplicación, forma de selección de las preguntas, etc.

Además de definir la naturaleza de la prueba hay que decidir cómo se aplica y tener en cuenta los siguientes aspectos:

- Acceso a la información: debe ser claro y fácil acceder a las preguntas, resultados, ayudas, etc.
- Control del usuario: debe poder decir qué hará y cuándo, por ejemplo, contestar las preguntas en cualquier orden, cambiar las respuestas, etc.
- Control del sistema para evitar que haya dificultades accidentales, por ejemplo, borrar un registro que se haya hecho, terminar sin querer la prueba, etc. [24]

1.5 – Conclusiones.

A manera de conclusiones se puede plantear que existen softwares educativos que se emplean solo para evaluar al estudiante, aunque la tendencia actual es construir softwares educativos que funcionen como tutoriales, simuladores y evaluadores, es decir sistemas que presenten el contenido, que simulen si es necesario y que evalúen el desempeño del estudiante.

Por otro lado se puede plantear que el módulo de evaluación de estos sistemas constituye una herramienta de vital importancia debido a que al evaluar el desempeño del estudiante se pueden trazar estrategias pedagógicas, claras y objetivas, encaminadas a satisfacer las necesidades docentes de los estudiantes.

En este sentido no existen actualmente algoritmos que puedan realizar una evaluación objetiva del estudiante y posteriormente trazar la estrategia pedagógica a seguir de acuerdo a su individualidad, este es el objetivo primordial de este trabajo.

CAPÍTULO 2: ANÁLISIS DE LAS TECNOLOGÍAS Y TENDENCIAS ACTUALES DE LA INTELIGENCIA ARTIFICIAL.

2.1 – Introducción.

La tecnología hoy día, lleva un acelerado paso en su desarrollo. La inteligencia Artificial ha hecho importantes aportes que han ayudado a dar soluciones a problemas reales existentes en el mundo actual.

En este capítulo se presentan conceptos y aplicaciones de algunas de las disciplinas que forman parte de las múltiples ramas existentes en la Inteligencia Artificial. Además de definir qué es la Inteligencia Artificial, se hace un estudio detallado sobre los temas referentes a la Inteligencia Artificial Conectivista donde se encuentran las Redes Bayesianas, las Redes de Markov y los Modelos Ocultos de Markov. Este estudio fue necesario para la toma de decisión de elegir la red con la que elaborará el diseño del algoritmo que se quiere obtener en esta investigación.

2.2 – La Inteligencia Artificial.

2.2.1 – Un poco de Historia.

La Inteligencia Artificial (IA) es una de las disciplinas más nuevas. Formalmente se inicia en 1956, durante un congreso en Dartmouth (USA) donde se propone el término de Inteligencia Artificial para agrupar todos los métodos, técnicas e intentos de simular el intelecto humano en la computadora. Casi todos los especialistas están de acuerdo con esto, y quien acuñó este término fue el matemático Josh McCarthy (creador del LISP), profesor del Instituto de Tecnología de Massachusetts.

Además de McCarthy, son considerados padres de esta disciplina Marvin Minsky también del Instituto de Tecnología de Massachusetts y Herbert Simon y Allen Newell de la Universidad de Carnegie-Mellon, sin embargo el estudio de la inteligencia contemplada como el razonamiento humano viene siendo estudiado por los filósofos hace más de 2 milenios.

La Inteligencia Artificial, en adelante IA, puede considerarse como una de las disciplinas más nuevas, siendo curiosamente considerada a la vez como una gran desconocida y una de las que más interés despierta. Esto es debido a que poca gente tiene claro qué es la IA, pero sin embargo es considerada por una gran mayoría de científicos como la disciplina donde han pensado alguna vez en trabajar. Pero, ¿qué es realmente la IA? Existen gran variedad de definiciones respecto a lo que es la IA, algunas más válidas que otras, pero se puede resumir diciendo que la IA trata de desarrollar sistemas que piensen y actúen racionalmente.

En los últimos años la IA ha ido evolucionando, quizás con mayor celeridad que otras disciplinas, motivada probablemente por su propia inmadurez. Todo esto ha llevado a que la IA actualmente abarque una gran cantidad de áreas, desde algunas muy generales como razonamiento, búsqueda, etc. a otras más específicas como los sistemas expertos, sistemas de diagnóstico, etc. Se puede indicar, sin lugar a dudas, que la IA puede ser aplicada hoy día a infinidad de disciplinas científicas y es que la IA es susceptible de aparecer allí donde se requiera el intelecto humano. En este aspecto cada nueva incursión de la IA en un campo diferente supone el empleo de una metodología de aplicación propia del investigador que lo intenta pues a diferencia de otras disciplinas, no existen ciertos estándares o criterios consensuados que unifiquen el proceso de aproximación de la IA a otras áreas y, en consecuencia, a problemas reales. La aparición de una metodología común a la hora de abordar la aplicabilidad de la IA puede resultar un elemento muy positivo en la corta vida de esta disciplina.

Hoy día todo relato futurista incluye siempre alguna referencia al tema (ya sea en forma de androide o de simple ordenador), e incluso en muchas ocasiones llega a ser el núcleo de la historia.

2.2.2 – Qué es la Inteligencia Artificial.

La inteligencia artificial es la ciencia que enfoca su estudio a lograr la comprensión de entidades inteligentes. Es evidente que las computadoras que posean una inteligencia a nivel humano (o superior) tendrán repercusiones muy importantes en la vida diaria.

Sin embargo, lo que en su día fue producto de la imaginación de diversos autores, hoy es una ciencia real sobre la que se trabaja. Y es que no hay dudas de la labor que desempeña en la sociedad la ciencia ficción, como precursora de nuevos campos de desarrollo e investigación.

Existen numerosas definiciones de IA, dependiendo del autor o el campo de especialización. Para tener una idea, a continuación se muestran seis modelos bastante representativos de esta ciencia:

- “IA es la atribuida a las máquinas capaces de hacer operaciones propias de seres inteligentes.” [16]
- “La IA es el estudio de las computaciones que permiten percibir, razonar y actuar.” [13]
- “La IA es el estudio de técnicas de resolución de problemas de complejidad exponencial mediante el uso de conocimiento sobre el campo de aplicación del problema.” [13]
- “Un campo de estudio que se enfoca a la explicación y emulación de la conducta inteligente en función de procesos computacionales.” [13]
- “El arte de crear máquinas con capacidad de realizar funciones que cuando son hechas por personas requieren de inteligencia.” [13]
- “El estudio de cómo lograr que las computadoras realicen tareas que, por el momento, los humanos hacen mejor.” [13]

La Inteligencia Artificial es una ciencia joven, comparada con el surgimiento de las demás ciencias, que aunque está lejos de las posibilidades relatadas en la ficción, tiene en la actualidad diversas aplicaciones de gran importancia donde entre sus innumerables campos de acción se pueden mencionar:

➤ *Juegos.*

Fueron precisamente, las primeras aplicaciones de la IA, pues constituían una excelente demostración de que las computadoras podían realizar actividades inteligentes. Actualmente se han desarrollado múltiples programas para jugar al ajedrez, las damas, resolver el cubo de Rubik, entre otros.

➤ *Solución general de problemas.*

Para la solución de problemas no algorítmicos o intratables computacionalmente la IA ofrece una alternativa eficiente. Se han desarrollado, de esta forma, demostradores de teoremas, simplificadores de fórmulas matemáticas, planificadores, entre otros.

➤ *Sistemas Expertos.*

Constituye hoy día el área de aplicación de la IA de mayor éxito. Los sistemas expertos permiten almacenar y utilizar el conocimiento de uno o varios expertos humanos en un dominio de aplicación concreto. Su uso incrementa la productividad, mejora la eficiencia en la toma de decisiones o simplemente permite resolver problemas cuando los expertos no están presentes. Muchos son los ejemplos de sistemas expertos desarrollados. Entre ellos: MYCIN (para el diagnóstico médico), PROSPECTOR (para la detección de yacimientos minerales), DENDRAL (para el análisis de la estructura de sustancias químicas), etc.

➤ *Procesamiento de lenguaje natural.*

Los programas de procesamiento de lenguaje natural utilizan las técnicas de IA para permitir que las computadoras comprendan y generen interfaces en lenguaje natural, lo cual haría más fácil el uso de estos programas por el usuario. La comprensión del lenguaje natural incluye el análisis de la sintaxis o gramática, la semántica o significado y el contexto del lenguaje. Un ejemplo de programas de procesamiento de lenguaje natural son los traductores de idiomas.

➤ *Visión.*

La visión por computadora comprende el análisis y evaluación de la información visual. El uso de las técnicas de IA permite a las computadoras examinar imágenes reales e identificar sus objetos y rasgos. Un sistema típico está compuesto por una cámara de video que capta las imágenes, un convertidor analógico-digital que envía las señales a la computadora y un software de IA, que mediante técnicas de búsqueda y reconocimiento de patrones, analiza la imagen. Un ejemplo de aplicación es el análisis de fotografías aéreas.

➤ *Robótica.*

La robótica es el campo de la ingeniería dedicada a simular ciertas capacidades físicas del hombre. Los robots no pueden actuar ante situaciones cambiantes y tomar decisiones si no se dotan de softwares de IA. Un robot dotado de inteligencia puede decidir cambiar el orden de la secuencia de sus operaciones, eliminar algunos pasos o modificar alguno de ellos. Para eso se dotan a los robots de

sensores de presión, temperatura, de luz, de cámaras de video, etc., además de softwares para la toma de decisiones. Al incorporar la IA a los robots aumenta su flexibilidad y utilidad.

➤ *Educación.*

La IA puede ser usada en la educación y el entrenamiento, mediante la creación de programas que actúan como tutores inteligentes. Los tutoriales convencionales son rígidos en el sentido de que sus textos son fijos para todos los estudiantes. Al dotarlos de técnicas de IA., ellos son capaces de variar su contenido en dependencia del conocimiento, habilidad e inteligencia del estudiante. Por otro lado, pueden proveerse de interfaces en lenguaje natural e incluso pueden adquirir nuevo conocimiento automáticamente en interacción con el tutor humano.

➤ *Programación automática.*

El desarrollo de un software tiene varias etapas. La primera consiste en la definición del problema y el objetivo a resolver. Luego se diseña funcionalmente el software, identificando los datos, los procesos y sus interrelaciones. Por último, se implementa en un lenguaje de programación concreto y se compila, llevándolo a código de máquina. La programación automática, o sea, la automatización del proceso de desarrollo de softwares significa permitirle al programador especificar qué es lo que quiere, en lugar de cómo se debe realizar. Con la programación automática se han obtenido muchos beneficios: reducción de errores, incremento de la complejidad de los sistemas que se implementan y liberación del programador de los detalles de implementación, dirigiendo su atención al estudio profundo del problema, lo que le permite llegar a soluciones más rápidas y correctas. [16]

2.3 – Qué es la Inteligencia Artificial Conectivista.

La inteligencia artificial conectivista es todo lo relacionado con las nuevas formulaciones de la IA o técnicas de redes, es decir redes bayesianas, redes neuronales, redes semánticas etc., generalmente conocidas como visión conexionista.

2.3.1 – Qué son las Redes Bayesianas.

Las redes bayesianas o probabilísticas se fundamentan en la teoría de la probabilidad y combinan la potencia del teorema de Bayes con la expresividad semántica de los grafos dirigidos; las mismas permiten representar un modelo causal por medio de una representación gráfica de las independencias y dependencias entre las variables que forman parte del dominio de aplicación.

Una red bayesiana es un grafo acíclico dirigido, las uniones entre los nodos tienen definidas una dirección, en el que los nodos representan variables aleatorias y las flechas representan influencias causales; el que un nodo sea padre de otro implica que es causa directa del mismo.

Estas redes son utilizadas en diversas áreas de aplicación como por ejemplo en medicina [2], ciencia [5] y economía [10]. Las mismas proveen una forma compacta de representar el conocimiento y métodos flexibles de razonamiento – basados en las teorías probabilísticas – capaces de predecir el valor de variables no observadas y explicar las observadas. Entre las características que poseen las redes bayesianas, se puede destacar que permiten aprender sobre relaciones de dependencia y causalidad, permiten combinar conocimiento con datos y pueden manejar bases de datos incompletas.

2.3.2 – Definición formal de las redes bayesianas.

Una red bayesiana es un grafo acíclico dirigido en el que los nodos representan variables aleatorias que pueden ser continuas o discretas; en las siguientes definiciones se utilizarán letras mayúsculas para denotar los nodos (X) y las correspondientes letras minúsculas para designar sus posibles estados (x_i).

Los estados que puede tener una variable deben cumplir con dos propiedades:

1. Ser mutuamente excluyentes, es decir, un nodo sólo puede encontrarse en uno de sus estados en un momento dado.
2. Ser un conjunto exhaustivo, es decir, un nodo no puede tener ningún valor fuera de ese conjunto.

La estructura de una red bayesiana se puede determinar de la siguiente manera:

1. Se asigna un vértice o nodo a cada variable (X_i) y se indica de qué otros vértices es una causa directa; a ese conjunto de vértices "causa del nodo X_i " se lo denota como el conjunto π_{X_i} y se le llamará "padres de X_i "
2. Se une cada padre con sus hijos con flechas que parten de los padres y llegan a los hijos.
3. A cada variable X_i se le asigna una matriz $P(X_i | \pi_{X_i})$ que estima la probabilidad condicional de un evento $X_i = x_i$ dada una combinación de valores de los π_{X_i} .

A continuación se indican algunas definiciones propias de la terminología de las redes bayesianas:

- *Probabilidad conjunta*

Dado un conjunto de variables $\{X, Y, \dots, Z\}$ la probabilidad conjunta especifica la probabilidad de cada combinación posible de estados de cada variable $P(x_i, y_j, \dots, z_k) \forall i, j, \dots, k$, de manera que se cumple que:

$$\sum P(x_i, y_j, \dots, z_k) = 1$$

- *Probabilidad condicional.*

La probabilidad condicional por definición es: $P(y_j | x_i) = \frac{P(y_j, x_i)}{P(x_i)}$, dado $P(x_i) > 0$.

Análogamente, si se intercambia el orden de las variables: $P(x_i | y_j) = \frac{P(y_j, x_i)}{P(y_j)}$

A partir de las dos fórmulas anteriores se obtiene: $P(y_j | x_i) = \frac{P(y_j)P(x_i | y_j)}{P(x_i)}$ esta expresión se

conoce como el *Teorema de Bayes*, que en su forma más general es:

$$P(y_j | x_i) = \frac{P(y_j)P(x_i | y_j)}{\sum_j P(x_i | y_j)P(y_j)}$$

al denominador se le conoce como el *Teorema de la Probabilidad Total*.

En las redes bayesianas el conjunto de valores que componen la probabilidad condicional de un hijo dados sus padres, se representa en las llamadas *tablas de probabilidad condicional*.

- *Independencia*

Por definición, se cumple que Y es independiente de X si y sólo si:

$$P(y_j, x_i) = P(y_j)P(x_i) \forall i, j.$$

Esto implica que:

$$P(y_j, x_i) = P(y_j) \forall i, j$$

$$P(x_i, y_j) = P(x_i) \forall i, j$$

Las redes bayesianas están diseñadas para hallar las relaciones de dependencia e independencia entre todas las variables que conforman el dominio de estudio. Esto permite realizar predicciones sobre el comportamiento de cualquiera de las variables desconocidas a partir de los valores de las otras variables conocidas. Esto presupone que cualquier variable de la base de datos puede comportarse como incógnita o como evidencia según el caso. [11]

2.3.3 – Aprendizaje Bayesiano.

El aprendizaje se puede ver como el proceso de encontrar la hipótesis más probable, dado el conjunto de ejemplo de entrenamiento y un conocimiento a priori sobre la probabilidad de cada hipótesis.

El obtener una red bayesiana a partir de datos, es un proceso de aprendizaje que se divide en dos etapas: el aprendizaje estructural y el aprendizaje paramétrico. La primera de ellas, consiste en obtener la estructura de la red bayesiana, es decir, las relaciones de dependencia e independencia entre las

variables involucradas y tiene como finalidad obtener las probabilidades a priori y condicionales requeridas a partir de una estructura dada. Por otra parte el aprendizaje paramétrico obtiene las probabilidades a priori y condicionales requeridas dada una estructura y las bases de datos.

El método para aprendizaje estructural de redes bayesianas se basa en el algoritmo desarrollado por Chow y Liu para aproximar una distribución de probabilidad por un producto de probabilidades de segundo orden, lo que corresponde a un árbol. La probabilidad conjunta de n variables se puede representar como:

$$P(X_1, X_2, \dots, X_n) = \prod_{i=1}^n P(X_i)P(X_i | X_{j(i)}) \text{ donde } X_{j(i)} \text{ es la causa o padre de } X_i.$$

Rebane y Pearl extendieron el algoritmo de Chow y Liu para poliárboles. En el caso de un poliárbol, la probabilidad conjunta es:

$$P(X) = \prod_{i=1}^n P(X_i | X_{j1(i)}, X_{j2(i)}, \dots, X_{jm(i)})$$

donde $\{X_{j1(i)}, X_{j2(i)}, \dots, X_{jm(i)}\}$ es el conjunto de padres de la variable X_i

Las redes bayesianas están diseñadas para hallar las relaciones de dependencia e independencia entre todas las variables que conforman el dominio de estudio. Esto permite realizar predicciones sobre el comportamiento de cualquiera de las variables desconocidas a partir de los valores de las otras variables conocidas. Esto presupone que cualquier variable de la base de datos puede comportarse como incógnita o como evidencia según el caso. [11]

2.3.4 – Aplicaciones y ejemplos.

Las redes bayesianas son utilizadas en diversas áreas de aplicación como por ejemplo el diagnóstico médico. Las mismas proveen una forma compacta de representar el conocimiento y métodos flexibles de razonamiento – basados en las teorías probabilísticas como ya se ha mencionado anteriormente – capaces de predecir el valor de variables no observadas y explicar las observadas. Entre las

características que poseen las redes bayesianas, se debe destacar que permiten aprender sobre relaciones de dependencia y causalidad, permiten combinar conocimiento con datos, evitan el sobreajuste de los datos y pueden manejar bases de datos incompletas.

Se puede citar algunos ejemplos:

1. El sistema OLAE [19] es una herramienta que recopila información sobre alumnos que resuelven problemas a nivel introductorio de física, analiza esos datos con métodos probabilísticos (redes bayesianas) y determina lo que sabe el alumno. OLAE genera automáticamente para cada problema una red bayesiana que relaciona el conocimiento (representado en forma de reglas de primer orden) con acciones concretas, como por ejemplo ecuaciones escritas. Usando la red resultante, OLAE observa el comportamiento del alumno y calcula las probabilidades de que el alumno conozca y use cada una de las reglas.
2. El sistema POLA (Probabilistic On-Line Assessment) [6] es una extensión del sistema OLAE para determinar no sólo las reglas que sabe el alumno sino el camino seguido por el mismo para la resolución del problema, tratando la incertidumbre en la interpretación de las acciones del alumno de forma consistente utilizando probabilidades. Es decir, mientras que OLAE sólo realiza lo que Anderson y otros llaman *traza del conocimiento* (determinación de qué sabe el alumno, incluyendo conocimiento correcto y errores), POLA realiza también la traza del modelo (seguimiento de la forma de resolver un problema). En particular, cuando existan varios caminos de resolución que sean consistentes con la acción que ha tomado el alumno, POLA tendrá la capacidad de decidir qué camino es más probable que haya sido el seguido por el alumno.
3. El sistema HYDRIVE [22][23] modela la habilidad que tiene un alumno para diagnosticar averías en el sistema hidráulico de los aviones F-15. El problema empieza con un vídeo en el que un piloto, que está a punto de aterrizar o de despegar, describe algún problema en el funcionamiento. La interfaz de HYDRIVE permite que el alumno intente diagnosticar la avería por los procedimientos usuales y le permite consultar tanto videos de las componentes como material de apoyo técnico. El comportamiento del alumno es observado por el sistema con el propósito de evaluar cómo el alumno

hace uso de la información disponible para dirigir las acciones que le permiten diagnosticar la avería. El sistema de diagnóstico de HYDRIVE evalúa la calidad de acciones de diagnóstico de averías concretas y caracteriza el conocimiento del alumno mediante el uso de variables más generales como conocimiento del sistema, estrategias y procedimientos usados.

4. El sistema SHERLOCK II [17] es una simulación realista que permite tutorizar a los alumnos en el diagnóstico de averías en aviación. En este sistema se asocia a cada variable de conocimiento una distribución de probabilidad difusa (dpd) con cinco valores, que van desde *ningún conocimiento* a *conocimiento total*. Esta dpd se actualiza, aumentándola o disminuyéndola en diferentes grados según factores como el tipo de evidencia disponible (acciones del alumno, pistas que se le han presentado hasta ahora, etc.). Ejemplos de tales variables son: *habilidad para usar aparatos de medición, habilidad para interpretar los resultados de una prueba, etc.*
5. El sistema ML-MODELER [14] es el módulo del alumno de un sistema adaptativo para la enseñanza de Química, que modela dinámicamente el proceso de aprendizaje de un alumno y es capaz de proporcionar tutorización adaptativa. ML-MODELER compara la traza de la solución del alumno con la traza de la solución experta, genera hipótesis sobre los errores del alumno e infiere (utilizando razonamiento basado en casos) los métodos de aprendizaje que el alumno ha utilizado para alcanzar el estado actual de conocimiento. De esta forma, ML-MODELER es capaz de modelar no sólo qué errores y qué áreas conceptuales están siendo problemáticas para el alumno, sino también el posible uso incorrecto de técnicas de aprendizaje como analogía, generalización y especificación.
6. El sistema MDF [1] es un tutor de matemáticas desarrollado para enseñar operaciones básicas con diferentes tipos de números (números enteros, fracciones, números mixtos y decimales). En MDF, cada tipo de problema se considera un tema, y hay relaciones de prerrequisito entre ellos. Cada tema tiene asociada una serie de habilidades, que son pasos en el proceso de resolución del problema. Por ejemplo, el tema *sumar fracciones* tiene asociadas las siguientes habilidades: encontrar el mínimo común múltiplo, calcular fracciones equivalentes, sumar numeradores y simplificar fracciones.

2.4 – Qué son las Redes de Markov.

2.4.1 – Campos de Markov.

Ciertos procesos, como un campo magnético o una imagen, se pueden ver como una serie de estados en el espacio o una malla de estados. A este tipo de modelos se les conoce como Campos de Markov (CAM). Estos procesos son una extensión de las cadenas de Markov en los cuales el índice de tiempo se substituye por un índice espacial, las cuales se tratarán posteriormente.

Podemos considerar los CAM como un caso especial de un modelo más general que son *las Redes de Markov*. Las Redes de Markov (RM) son modelos gráficos probabilísticos no dirigidos, que representa las dependencias e independencias de un conjunto de variables.

Por ejemplo: Un subconjunto *A* de variables es independiente de otro subconjunto *C* dado un tercer subconjunto *B*, si los nodos en *B* separan en el grafo *A* de *C*.

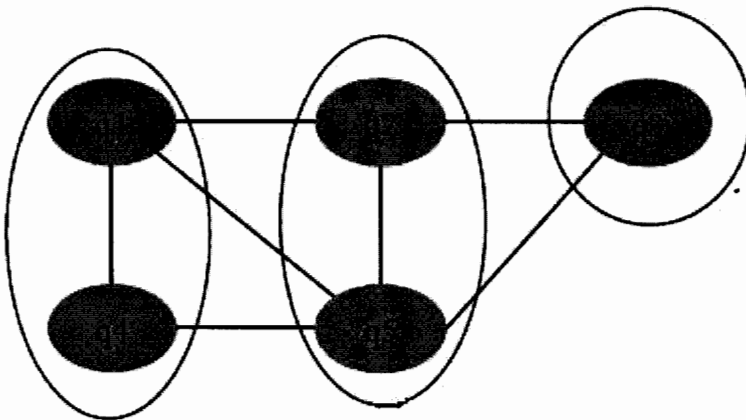


Figura 2.1: Representación de las Redes de Markov.

2.4.2 – Parametrización de las Redes de Markov.

Podemos en general representar la probabilidad conjunta de una RM como el producto de las probabilidades locales (potenciales) de conjuntos completos con un factor de normalización:

$$P(X_1, X_2, \dots, X_n) = \frac{1}{Z} P(C_1)P(C_2) \dots P(C_m)$$

Es suficiente considerar sólo los *cliques*, pero también se pueden considerar otros conjuntos.

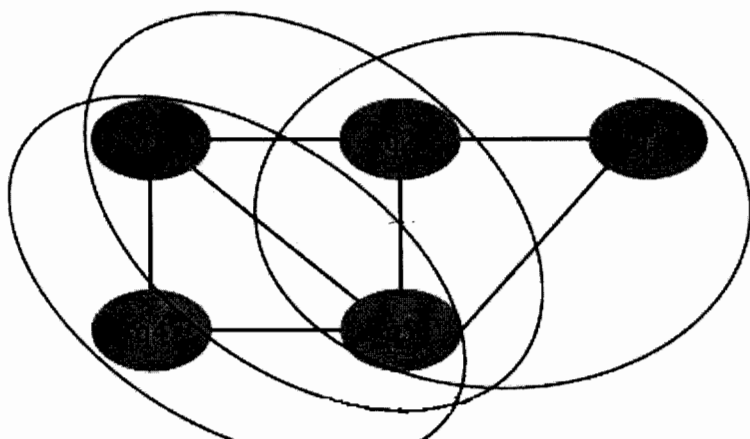


Figura 2.2: *Cliques de una Red de Markov.*

2.4.3 – Algoritmo.

El obtener la configuración de mayor probabilidad (mínima energía) en forma directa es casi imposible, por lo que se plantea como un problema de optimización. Se busca la configuración de mayor probabilidad, sin tener que calcular directamente las probabilidades de cada configuración.

El problema incluye 3 aspectos:

1. *Representación*: CAM con un esquema de vecindad y los potenciales asociados.
2. *Función objetivo*: Función de energía a minimizar.
3. *Algoritmo de optimización*: simulación estocástica (Metrópolis).

Algoritmo básico:

- Inicializar con un valor aleatorio cada variable.
- Repetir:

- Calcular el valor de energía (potencial) de cada variable en base a la función deseada y los valores de vecinos.
- Si el valor de energía es menor al anterior cambiar de valor.
- Si no, con cierta probabilidad también cambiar de valor.
- Hacer para todas las variables en el campo.
- Hasta que se cumplan n iteraciones u otro criterio de terminación.
- Obtener configuración “óptima”.

Formas de optimización:

- Metrópolis: con probabilidad fija; se puede pasar a estados de mayor energía.
- Recocido simulado: se va decrementando la probabilidad de pasar a estados de mayor energía (temperatura).
- ICM: tomar siempre el estado de menos energía. [28]

2.4.4 – Aplicaciones.

La principal aplicación de este modelo es el procesamiento de imágenes dentro de la cual se emplea en:

- Eliminación de ruido.
- Filtrado de una imagen.
- Segmentación de texturas.
- Visión estereoscópica.
- Recuperación de forma (3-D).
- Reconocimiento de objetos.

2.5 – Qué son los Modelos Ocultos de Markov.

Un Modelo Oculto de Markov (Hidden Markov Model (HMM)) es un proceso estocástico que consta de un proceso de Markov no observado (oculto) $q = \{q_t\} t \in N$ y un proceso observado $O = \{o_t\} t \in N$ cuyos estados son dependientes estocásticamente de los estados ocultos, es decir, es un proceso bivariado (q, O) . Los HMMs se pueden considerar también como sistemas generativos estocásticos, los cuales se emplean en la modelación de series de tiempo.

2.5.1 – Cadenas de Markov.

Una cadena de Markov $q = \{q_t\} t \in \mathbb{N}$ es un proceso estocástico de Markov discreto. Un proceso estocástico se llama de Markov si conocido el presente, el futuro no depende del pasado, esto quiere decir, que dada una variable estocástica q_{t-1} que denota el estado del proceso en el tiempo $t-1$, entonces la probabilidad de transición en el momento t se define como: $P[q_t = \sigma_t \mid q_{t-1} = \sigma_{t-1}]$.

Formalmente, una cadena de Markov se define como (Q, A) , donde $Q = \{1, 2, \dots, N\}$ son los posibles estados de la cadena y $A = (a_{ij})_{n \times n}$ es una matriz de transición de estados en el modelo.

Si $A(t) = a_{ij}(t)_{n \times n}$ es independiente del tiempo entonces el proceso se llama homogéneo y las probabilidades de transición de estados son de la forma $a_{ij}(t) = P[q_t = j \mid q_{t-1} = i]$.

Un vector de probabilidades de salida por cada estado (matriz),

$$B = \{b_{ik}\} \text{ donde } b_{ik} = P(O_t = o_k \mid S_t = q_i).$$

En forma compacta: $\lambda = \{A, B, \Pi\}$.

La condición fundamental de que sea una cadena de Markov establece que las probabilidades de transición y emisión dependen solamente del estado actual y no del pasado, esto es:

$$P[q_t = j \mid q_{t-1} = i, q_{t-2} = k, \dots] = P[q_t = j \mid q_{t-1} = i] = a_{ij}(t). \quad [28]$$

Por ejemplo, la secuencia resultante de lanzar una moneda k veces es un proceso estocástico. Los estados del sistema son dos: cara y sello. Estos a su vez conforman el espacio muestral pues son el conjunto de todos los estados posibles de nuestro experimento. Se supone que los resultados de lanzar una moneda 10 veces son los siguientes: $ccscscscscs$.

Si se hace que el número de lanzamientos k creciera hasta el infinito, se puede aproximar a la probabilidad de que ocurra el evento cara o sello de la siguiente forma:

$$P(\text{cara}) = \text{veces que aparece cara} / k$$

2.5.2 – Definición de Modelos Ocultos de Markov.

Un modelo oculto de Markov es una cadena de q junto con un proceso estocástico que toma valores en un alfabeto Σ y el cual depende de q . Estos sistemas evolucionan en el tiempo pasando aleatoriamente de estado a estado y emitiendo en cada momento al azar algún símbolo del alfabeto Σ . Cuando se encuentra en el estado $q_{t-1} = i$, tiene la probabilidad a_{ij} de moverse al estado $q_t = j$ en el siguiente instante y la probabilidad $b_j(k)$ de emitir el símbolo $o_t = v_k$ en el tiempo t . Solamente los símbolos emitidos por el proceso q son observables, pero no la ruta o secuencia de estados q , de ahí el calificativo de "oculto" de Markov, ya que el proceso de Markov q es no observado.

El siguiente ejemplo ilustra un proceso q independiente del tiempo. Supóngase que en un salón se encuentra un número N muy grande de urnas de vidrio. Dentro de cada urna se tiene una cantidad M de bolas de colores. Un mago está en el salón y de acuerdo con algún procedimiento aleatorio elige una urna inicial. De ésta saca al azar una bola y registra su color como una observación. La bola es retornada a la urna de la cual fue seleccionada. A continuación selecciona una nueva urna de acuerdo con un procedimiento aleatorio que depende de la urna actual y la elección de alguna bola es repetida. Este proceso completo se realiza en un tiempo T y genera una secuencia de observación finita de colores O de longitud T , la cual puede modelarse como la salida observable de un HMM. Se asume que las urnas son seleccionadas independientemente. [8]



Figura 2.3: Modelo de urnas y bolas de N estados que ilustra el caso general de un HMM con símbolos discretos.

2.5.3 – Arquitecturas de HMMs.

Un HMM puede ser representado como un grafo dirigido de transiciones y emisiones. La arquitectura específica que permita modelar de la mejor forma posible las propiedades observadas depende en gran medida de las características del problema. [8]

Las arquitecturas mas usadas son:

- 1) *Ergódicas o completamente conectadas* en las cuales cada estado del modelo puede ser alcanzado desde cualquier otro estado en un número finito de pasos.

- 2) *Izquierda-derecha, hacia adelante o Bakis* las cuales tienen la propiedad de que en la medida que el tiempo crece se avanza en la secuencia de observación asociada O , y en esa misma medida el índice que señala el estado del modelo permanece o crece, es decir, los estados del sistema van de izquierda a derecha (Ver figura 2.4). En secuencias biológicas y en reconocimiento de la voz estas arquitecturas modelan bien los aspectos lineales de las secuencias.

- 3) *Izquierda-derecha paralelas*, son dos arquitecturas *izquierda-derecha* conectadas entre sí (Ver figura 2.5)

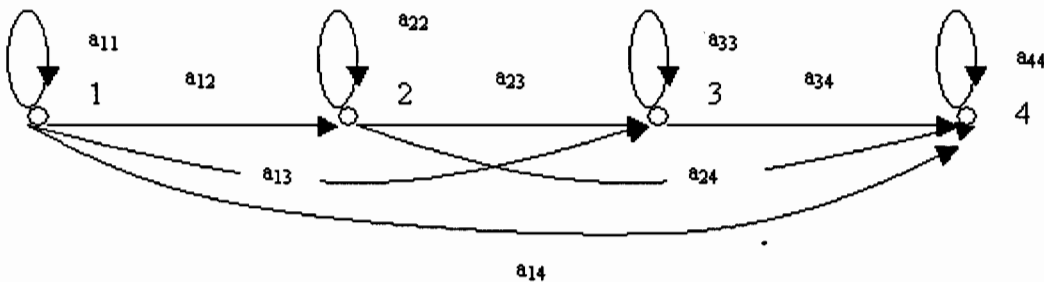


Figura 2.4: Modelo izquierda-derecha con 4 estados.

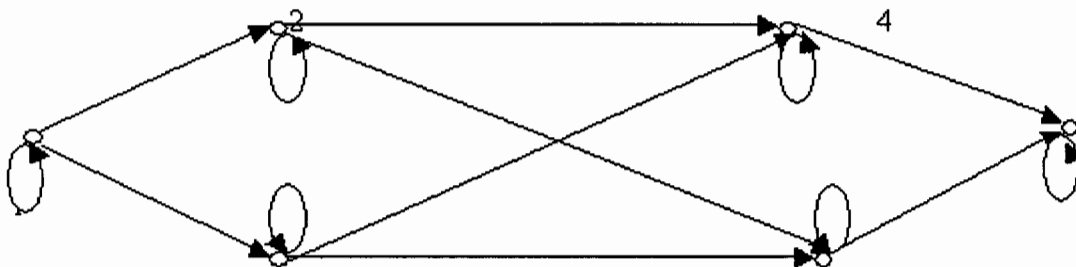


Figura 2.5: Modelo izquierda derecha paralelo con 6 estados.

2.6 – Conclusiones.

La inteligencia artificial es una de las ciencias de la computación más prometedoras y que más aplicación tendrá en el futuro.

Los algoritmos inteligentes son de amplia utilización en la resolución de problemas a los cuales los algoritmos de complejidad polinomial no dan solución, o problemas para los cuales no existen algoritmos. Dentro de la inteligencia artificial existen múltiples tendencias una de ellas es la que esta basada en los modelos conexionistas.

Después de estudiadas las Redes Bayesianas, las Redes de Markov y los Modelos Ocultos de Markov se toma la decisión sobre la red más idónea a utilizar en la elaboración de un algoritmo inteligente para los módulos de evaluación de los estudiantes en los softwares educativos, esta es la Red Bayesiana y es seleccionada por las siguientes razones:

- Se necesita de una red de grafos acíclicos y dirigidos ya que, cuando se elabora la red se debe recorrer de un nodo a otro de forma dirigida, o sea, cada nodo tiene un padre y un descendiente por los cuales se transita, las uniones entre los nodos tienen definidas una dirección que parten de los padres y llega a sus hijos lo que implica que es causa directa del mismo; de ahí que se pueda estimar en que parte de la red se encuentra en cada momento y cuales son los caminos a recorrer, además el estado inicial es excluyente del final, es decir, no hay ninguna conexión entre ellos lo que lo hace acíclico.
- Por su parte las redes de Markov son modelos gráficos probabilísticos no dirigidos, que representa las dependencias e independencias de un conjunto de variables donde la configuración de mayor probabilidad (mínima energía) en forma directa es casi imposible, por lo que no se sabría a ciencia cierta que nodo es padre de quien y cual es la trayectoria idónea a seguir, por su parte un Modelo Oculto de Markov es un proceso estocástico que consta de un proceso de Markov no observado (oculto) y un proceso observado. Solamente los símbolos emitidos por el proceso q son observables, pero no la ruta o secuencia de estados, ya que el proceso de Markov q es no observado, lo cual no permite conocer el camino a seguir, ni la relación entre los nodos.

CAPÍTULO 3: PROPUESTA DEL ALGORITMO.

3.1 – Introducción.

El diagnóstico es uno de los procesos más importantes dentro de cualquier Sistema de Tutor Inteligente (STI) y por ello el modelado del alumno es un problema central en el diseño y desarrollo, por lo que el sistema debe ser capaz de determinar con la mayor precisión y rapidez posible cuál es su estado cognitivo del alumno, es decir, qué partes del dominio que se pretende enseñar ya domina y cuáles son las que aún desconoce.

Pero no siempre se logra un correcto funcionamiento de los STI puesto que casi siempre el problema del modelo del alumno se resuelve mediante la aplicación de heurísticos diseñados a tal fin, por ende, la falta de consistencia hace que el comportamiento del sistema sea impredecible, sobre todo en situaciones diferentes a las previstas inicialmente por sus diseñadores, esto también es válido para los software educativos.

Para la solución del problema se propone utilizar la construcción de Test Adaptativos Informatizados (TAI) en el marco teórico de las redes bayesianas para mejorar la precisión del diagnóstico, lo cual es más adecuado si el objetivo del test no es meramente evaluar al alumno sino llevar a cabo el proceso de diagnóstico.

Este capítulo se estructura de la siguiente manera: primero se exponen todos aquellos conceptos que se consideran necesarios para la comprensión de la solución propuesta, es decir, se habla brevemente del modelado de alumno, de los test adaptativos computerizados, diagnóstico mediante redes bayesianas en los TAI para lo cual se describe cual es el modelo estructural de la propuesta, el diseño de la red bayesiana así como la especificación de los parámetros. En esta misma sección se discuten diferentes aproximaciones para simplificar el problema de la obtención de los parámetros, dependiendo del tipo de relaciones que se quieran lograr, además del desarrollo de la propuesta con ayuda de UML, destacando que este último aspecto se desarrolla para esclarecer la solución desarrollada.

3.2 – Modelado del alumno.

El problema del modelado del alumno puede dividirse en dos componentes:

1. **Seleccionar la estructura de datos** (*modelo del alumno*) que será usada para representar toda la información relativa al alumno: estado cognitivo, estrategias instructoras preferidas, pantallas visitadas, ejercicios resueltos, etc.
2. **Elegir el procedimiento que se utilizará para realizar el diagnóstico**, es decir, para inferir dada la información generada en la interacción del alumno con el sistema (problemas resueltos, pantallas visitadas, etc.) el estado cognitivo del alumno. Evidentemente ambas componentes están íntimamente relacionadas, y por tanto lo ideal es diseñarlas y desarrollarlas simultáneamente.

3.3 – Diagnóstico basado en el modelo de la Teoría de Respuesta al Ítem (TRI).

Los test adaptativos informatizados (TAI) surgieron como alternativa a los test tradicionales de lápiz y papel con objeto de mejorar tanto los tiempos necesarios para obtener una estimación precisa del conocimiento del alumno a partir de sus respuestas a un conjunto de preguntas tipo test como la precisión de dicha estimación. Con la ayuda de la gran capacidad de almacenamiento y procesamiento de los ordenadores y usando como marco teórico la teoría de la probabilidad, se pudo cumplir ambos objetivos. Sin embargo, los test computerizados adaptativos actuales están contruidos para medir el rendimiento del alumno en función de una única variable, y, según el propósito con el que se realice el test, esta medición resulta insuficiente. Por ejemplo, si se pretende utilizar un TAI para realizar el diagnóstico en un sistema tutor inteligente, es claro que una medida del conocimiento general del alumno no sería suficiente, sino que se necesita una información más completa para saber exactamente qué partes del curriculum resultan problemáticas o no son suficientemente bien comprendidas por el alumno, con objeto de determinar cuál es la estrategia tutora más apropiada y de seleccionar el material adecuado con el nivel de detalle que precise el alumno según su estado de conocimiento actual. Para solucionar este problema, se propone la construcción de test adaptativos en el marco teórico de las redes bayesianas.

En los métodos de evaluación tradicionales, el almacenamiento y análisis de la información se realizaba de forma estática. El uso de los ordenadores en los procesos de evaluación abrió la posibilidad de hacer ambas cosas de una forma dinámica.

3.3.1 – Tests adaptativos informatizados.

Un test adaptativo informatizado es básicamente un test administrado por la computadora donde la presentación de cada pregunta (o ítem) y la decisión de finalizar el test se toman de forma dinámica basándose en la respuesta del alumno y en la estimación de su nivel de conocimiento.

Esta filosofía básica es descrita por Wainer y Mislevy con las siguientes palabras:

“La noción básica de un test adaptativo es imitar automáticamente lo que haría un examinador humano sabio”. [30]

Es decir, si un examinador propone una pregunta que resulta ser demasiado fácil, la siguiente debería ser más difícil (y viceversa).

En términos más precisos, un TAI es un algoritmo iterativo que comienza con una estimación inicial del nivel de conocimiento del alumno y que tiene los siguientes pasos:

1. Todas las preguntas que no se han administrado todavía son examinadas para determinar cual será la mejor para ser propuesta a continuación, según el nivel de conocimiento estimado del alumno.
2. La pregunta es planteada y el alumno responde.
3. De acuerdo con la respuesta del alumno, se realiza una nueva estimación de su nivel de conocimiento.
4. Los pasos del 1 al 3 se repiten hasta que se cumpla alguno de los criterios de parada o terminación definidos.

Este proceso se ilustra en la figura 3.1.

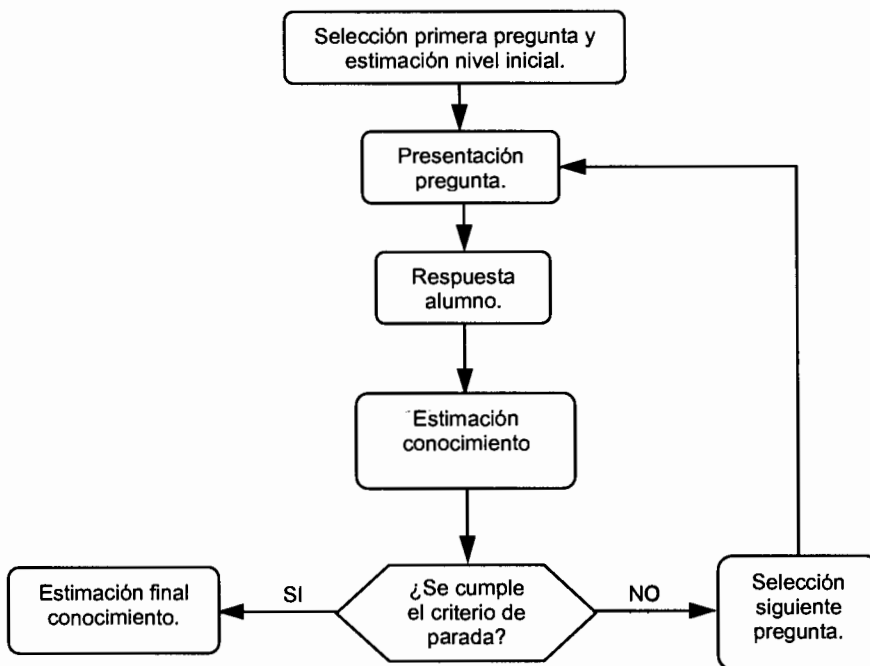


Figura 3.1: Diagrama de flujo de un test adaptativo.

Los TAIs tienen importantes ventajas frente a los tests tradicionales a lápiz y papel entre las que destacan: (a) decremento significativo en la longitud de los tests; (b) estimaciones más precisas del nivel de conocimiento del alumno; (c) mejora en la motivación de los alumnos; (d) se puede almacenar un gran banco de preguntas, incluyendo enunciados y posibles respuestas con contenido multimedia.

Los elementos básicos de un TAI son:

- **Modelo de respuesta del ítem.** Este modelo describe como el sujeto responde al ítem según su nivel de conocimiento. Cuando se llevan a cabo mediciones del nivel de conocimiento, cabe esperar que el resultado obtenido no dependa del instrumento utilizado, es decir, la medida ha de ser invariante con respecto al tipo de test y al sujeto al que se le aplica el test.
- **Banco de preguntas.** Constituye uno de los elementos fundamentales para la creación de un TAI. Para definir un banco de preguntas eficiente se deben especificar las distintas áreas de conocimiento del dominio. Una vez hechas las especificaciones del contenido del test, el banco de preguntas debe contener ítems en suficiente número, variedad y niveles de dificultad. [12]

- **Nivel de conocimiento de entrada.** Elegir de forma adecuada el nivel de dificultad de la primera pregunta que se realice en el test puede reducir sensiblemente la longitud del mismo. Para ello se pueden usar diferentes criterios como tomar el nivel medio de los sujetos que han realizado el test previamente, o crear un perfil de sujeto y usar el nivel medio de los alumnos con un perfil similar.
- **Método de selección de preguntas.** Un test adaptativo selecciona el siguiente ítem que va a ser presentado en cada momento en función del nivel estimado del conocimiento del alumno y de las respuestas a los ítems previamente administrados. Seleccionar el mejor ítem puede mejorar la precisión en la estimación del nivel de conocimiento y reducir la longitud del test.
- **Criterio de terminación.** Para decidir cuándo debe finalizar un test se pueden usar diferentes criterios tales como parar cuando se haya alcanzado una precisión determinada en la medida del nivel de conocimiento, cuando se hayan planteado un número determinado de ítems, etc. [29]

3.3.2 – Teoría de respuesta al ítem.

La mayor parte de las aplicaciones prácticas de la teoría de la medida en Psicología y Educación están basadas en la Teoría Clásica de Tests (TCT), cuyas deficiencias alentaron la búsqueda de modelos alternativos. Entre los que mayor difusión han tenido destacan los basados en *la Teoría de la Respuesta al Ítem* (TRI), inicialmente conocida como *teoría del rasgo latente*. [15]

La TRI parte de varios supuestos. Los principales son:

1. *Unidimensionalidad:* en una prueba todos los ítems están midiendo una y sólo una característica de los examinados.
2. *Independencia local:* dado un nivel de habilidad, las respuestas a los ítems no pueden estar correlacionadas entre sí. Si hay correlación entre preguntas, ésta sólo se explica por habilidad.
3. Que todos los alumnos hayan tenido *experiencias educativas similares*.
4. Que la prueba no haya sido apurada.
5. Que no haya *efectos de contexto* no controlados: algunas preguntas se comportan de modo diferente según la posición que tengan en la prueba. [9]

La TRI, partiendo de hipótesis restrictivas, intenta dar fundamentos probabilísticos al problema de la medición de rasgos no observables. Su nombre es debido a que se consideran los ítems como las unidades básicas de los tests.

Todos los modelos TRI tienen unas características comunes: (a) suponen la existencia de rasgos o aptitudes latentes que permiten predecir o explicar la conducta de un examinando ante un ítem de un test; (b) la relación entre el rasgo y la respuesta del sujeto al ítem puede describirse por medio de una función monótona creciente, denominada Curva característica del ítem (CCI).

Hay muchas funciones que cumplen esta propiedad de monotonía creciente que debe tener una CCI. Las diferencias entre ellas son la forma y los parámetros que las describen. Para explicar las características intrínsecas de la pregunta se han usado varios parámetros.

a) La *dificultad del ítem*, que describe qué cantidad de aptitud requiere el ítem para ser resuelto correctamente, o dicho de otra forma, la posición del ítem en la escala de aptitud. En la figura 3.2 aparecen tres posibles CCIs que sólo difieren en el nivel de dificultad.

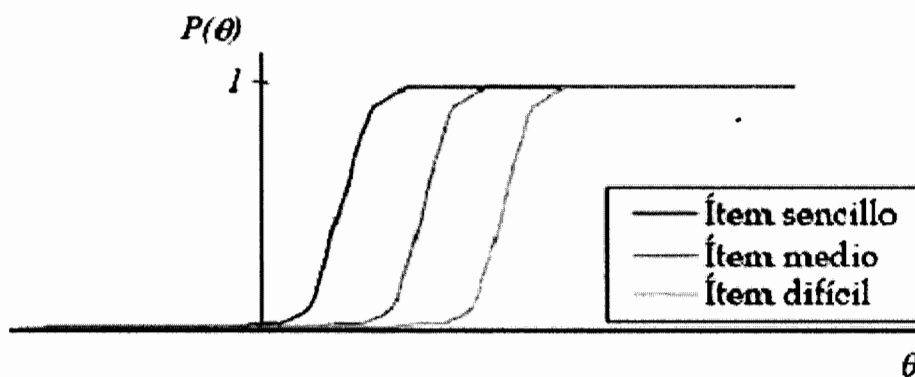


Figura 3.2: CCIs correspondientes a preguntas con diferentes niveles de dificultad.

b) El *nivel de discriminación*, que nos indica hasta qué punto el ítem permite diferenciar entre los sujetos que tienen una aptitud inferior a la posición del ítem y los que tienen una aptitud superior. En la figura 3.3 se muestra CCIs con diferentes factores de discriminación.

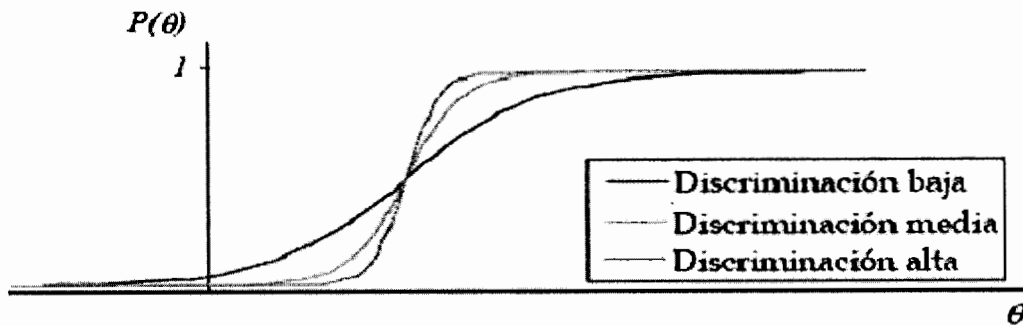


Figura 3.3: CCIs correspondientes a preguntas con diferentes factores de discriminación.

En esta gráfica se observa que a mayor factor de discriminación, más rápidamente crece la probabilidad de contestar correctamente a la pregunta para aptitudes superiores al nivel de dificultad.

Para representar el caso en que la CCI tiene asíntotas izquierda/derecha diferentes de 0/1, se tienen otros dos parámetros:

c) El *factor de adivinanza* c , que representa la probabilidad de que los alumnos de aptitud muy baja contesten correctamente a la pregunta, es decir, "adivinen" la respuesta correcta.

d) El *factor de distracción* d . $1-d$ representa la probabilidad de que los sujetos con aptitud muy alta no contesten correctamente a la pregunta debido a fallos no debidos a la falta de aptitud. Ver figura 3.4.

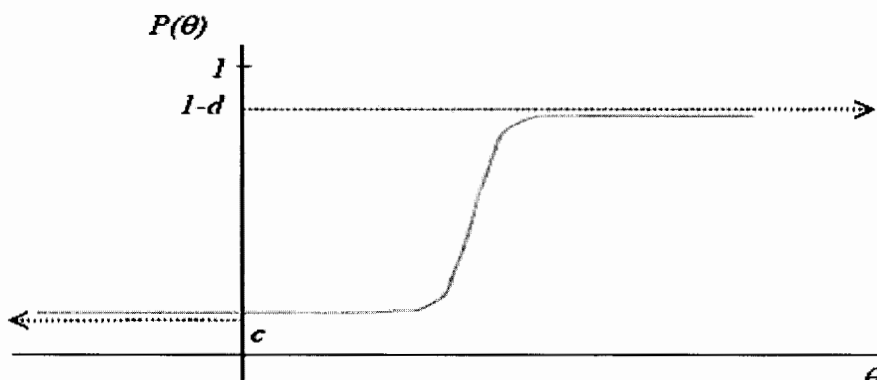


Figura 3.4: CCI con factores de adivinanza y distracción no nulos.

Los primeros modelos aparecidos son conocidos con el nombre de "modelos normales" ya que la forma de la CCI era la de una distribución normal. Las dificultades para el manejo analítico de esta función llevaron a los *modelos logísticos*, basados en la función de distribución logística, entre los que destacan los de un parámetro [26], y los de dos y tres parámetros [4]. Todos estos modelos están basados en la suposición de independencia local que afirma que si la aptitud θ que explica el rendimiento en el test permanece constante, las respuestas de los examinados a un par de ítems cualquiera, son estadísticamente independientes. En el modelo de tres parámetros la CCI_i del ítem i indica la probabilidad de que el alumno responda correctamente ($u_i=1$) supuesto que $\theta=x$ mediante la expresión:

$$P_i(x) = P(u_i = 1 / \theta = x) = c_i + \frac{1 - c_i}{1 + e^{-1.7a_i(x-b_i)}}$$

donde a_i es el índice de discriminación, b_i es el grado de dificultad y c_i es el factor de adivinanza. [22]

3.3.3 – Estimación del nivel de habilidad.

Este modelo de respuesta se usa para obtener la estimación del rasgo θ . Hay varios métodos para ello. Por ejemplo, en el *método de máxima probabilidad* [15] se busca el valor de θ que hace máxima la función de probabilidad, en otras palabras, para estimar el nivel de habilidad de un examinado conociendo los parámetros se aplica la siguiente fórmula:

$$L(\bar{u} / \theta = x) = L(u_1, \dots, u_n / \theta = x) = \prod_{i=1}^n P_i(x)^{u_i} (1 - P_i(x))^{1-u_i}$$

donde $u = (u_1, \dots, u_n)$ es el vector de las respuestas del alumno o valor dicotómico, es decir, para $i = 1, \dots, n$, u_i es 1 si la respuesta al ítem i ésimo es correcta y 0 en caso contrario, y $P_i(x)$ es la probabilidad de responder correctamente al ítem i cuando el nivel de conocimiento es $\theta = x$.

Basta con encontrar los valores de θ para los cuales la ecuación sea máxima. Este sería el nivel de habilidad del examinado.

Por otra parte, el *método bayesiano* calcula el nivel de conocimiento para el que la distribución a posteriori es máxima. Esta distribución es proporcional al producto de la función de probabilidad y la función de densidad a priori, es decir:

$$P(\theta / \mu) \propto L(\theta / \mu) f(\theta)$$

Normalmente los métodos bayesianos suponen que θ tiene la distribución normal, así que f suele ser la función de densidad de la normal. Otra opción es suponer que θ tiene la distribución uniforme, en cuyo caso el estimador bayesiano y el estimador máximo verosímil coinciden.

3.3.4 – Banco de ítems.

Si se utiliza el modelo logístico de tres parámetros como modelo de respuesta, un buen banco debe tener las siguientes características:

- Los ítems deben tener elevados factores de discriminación, la mayoría por encima de 1, para realizar estimaciones precisas con pocos ítems.
- La distribución de frecuencias de los niveles de dificultad debe ser aproximadamente uniforme, es decir, debe existir un número parecido de ítems para cada nivel de dificultad.

Estas características favorecerán la calidad psicométrica del test, es decir, harán que el test produzca estimaciones más precisas utilizando un menor número de preguntas.

3.4 – Diagnóstico mediante redes bayesianas en los TAI.

Las redes bayesianas pueden ser utilizadas en el problema de diagnóstico del alumno y por ende serán utilizadas en la propuesta realizada. Para ello se define en primer lugar el modelo estructural que servirá como soporte del proceso evaluador (nodos, enlaces y parámetros).

La propuesta no es más que el diagnóstico del conocimiento del alumno utilizando redes bayesianas en los TAI (Tests Adaptativos Informatizados), respetando los cinco supuestos planteados con anterioridad.

Eva Millán [21] ha desarrollado un trabajo muy interesante en esta rama de unir la TRI y las redes bayesianas, pero a veces viola los supuestos de la TRI, por ejemplo ella incluye preguntas que tienen relación con dos o más contenidos violando el supuesto de *Unidimensionalidad* del ítem (ver figura 3.5). Esto determina que no se tenga un correcto uso de la teoría y si los supuestos no se ajustan a la hipótesis, la tesis no es válida. Por lo que se puede arribar a la conclusión de que el algoritmo tal y como lo concibe Millán tiene ciertas deficiencias con la TRI en los supuestos que son subsanados con el algoritmo que se propone.

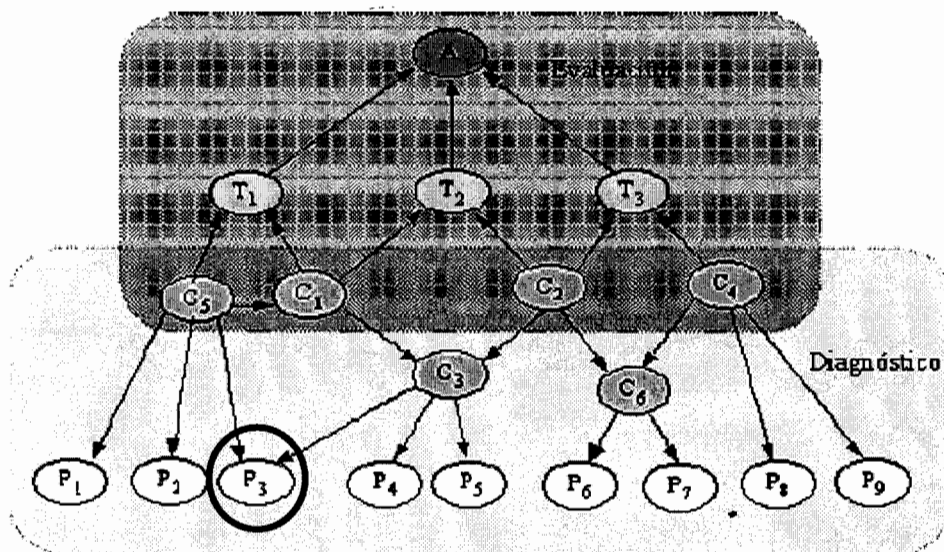


Figura 3.5: Red bayesiana como la concibe Millán.

3.4.1 – Modelo estructural.

Para utilizar redes bayesianas para los test adaptativos lo primero que se debe hacer es definir el modelo estructural, es decir, definir qué variables van a formar parte de la red y qué tipo de relaciones van a existir entre dichas variables.

Los nodos que están involucrados son los siguientes:

- *Nodos pregunta*, se representan por **P** y que se podrán responder correcta o incorrectamente.
- *Nodos pregunta inicial*, se representan por **I** y que se podrán responder correcta o incorrectamente.

- *Nodos concepto y tema*, que se representan por **C** y **T**, respectivamente, y que se considerarán sabidos o no sabidos.
- *Nodos Nivel*: se representa por **N**, agrupa los conceptos y temas en diferentes niveles para realizar el diagnóstico parcial de los alumnos.
- *Nodo Evaluación del Nivel*: Se representa por **Eval.Nivel** y calcula la probabilidad conjunta del estudiante en todas las preguntas iniciales y estima el nivel más apropiado para el mismo.

Se llama *concepto elemental* a una parte mínima de conocimiento, en el sentido de que es una parte del conocimiento que no admite descomposición en partes más pequeñas. Los conceptos elementales se consideran las unidades básicas del conocimiento.

Para representar un concepto elemental se utiliza una variable aleatoria que se denota por C y tendrá una distribución de Bernouilli, es decir, tomará el valor 1 cuando el alumno conozca ese concepto elemental y el valor 0 cuando no lo conozca.

La ley de probabilidad de la variable C vendrá por tanto dada por:

$$P(C = x) = p^x (1 - p)^{1-x}$$

donde p es la probabilidad de que el alumno conozca el concepto C , y x puede tomar los valores 0 ó 1.

Se llamará *tema* a un par (C, w) , donde:

- C es un conjunto de conceptos elementales $C = \{C_1, \dots, C_n\}$ que se suponen independientes entre sí.
- $w = (w_1, \dots, w_n)$ es un vector de pesos que miden la importancia relativa de cada concepto en el tema al que pertenece. Sin pérdida de generalidad, se supone que:

$$\sum_{i=1}^n w_i = 1$$

Para representar el conocimiento del alumno acerca de un tema, se utiliza una variable aleatoria T que se define de la siguiente forma:

$$T = \sum_{j=1}^n w_j C_j$$

La ley de probabilidad de la variable aleatoria T viene entonces dada por:

$$P(T = \sum_{j \in S} w_j) = \prod_{j \in S} p_j \prod_{k \in S^c} (1 - p_k), \text{ para cada } S \subseteq \{1, \dots, n\},$$

donde $p_i = P(C_j = 1)$, para $j = 1, \dots, n$.

Nodo recolección de evidencia.

Con estos nodos se pretende recoger todo aquello que nos pueda proporcionar información sobre el estado de conocimiento del alumno. Se distinguen varios tipos de fuentes de información o evidencias pero se utilizará solo la tipo test multirespuesta:

Preguntas tipo *test multirespuesta*. Son preguntas tipo test en la que el alumno tiene que elegir entre varias posibles respuestas, y en las que sólo una de ellas es correcta. Para representar este tipo de preguntas, se utiliza una variable que se denota por P y que tendrá una distribución de Bernouilli, es decir, tomará el valor 1 cuando el alumno elija la respuesta correcta, y el valor 0 cuando elija una respuesta incorrecta. La ley de probabilidad de P vendrá por tanto dada por:

$$P(x) = p^x (1 - p)^{1-x},$$

Donde p es la probabilidad de que el alumno responda correctamente a la pregunta P , y x toma los valores 0 ó 1. [20]

Los tipos de relaciones serán:

- **Relaciones concepto-pregunta:** Se considera que conocer un determinado concepto influirá positivamente en contestar adecuadamente una pregunta relacionada con él, y no conocerlo influirá en contestarla incorrectamente.
- **Relaciones de prerrequisito:** Se establecen entre los conceptos, entre los temas, o bien entre tema y concepto. Se considera que el conocimiento de un concepto (o tema) es condición indispensable para el conocimiento de un concepto (o tema) que tiene al anterior como prerrequisito.
- **Relaciones de agregación:** Estas relaciones son más difíciles de modelar utilizando un enfoque puramente bayesiano. En este caso, lo que se pretende es, a partir de la probabilidad de conocer cada uno de los conceptos de un tema, construir una calificación para el tema, y a partir de éstas, construir una calificación global de la asignatura, aunque esta evaluación no forma parte de nuestra propuesta, si se puede tener en cuenta al realizar un software para una asignatura en específico.

Por ejemplo, se tiene un elemento E que se subdivide en una colección finita de elementos más específicos E_1, E_2, \dots, E_n , elementos que estarán representados por las variables aleatorias E_1, E_2, \dots, E_n , y cada uno se va a considerar dominado o no dominado que se ajusta a elementos que describan temas conceptos o los que describa habilidades. Para modelar la relación existente entre dominar el elemento E y dominar los elementos E_1, E_2, \dots, E_n utilizando relaciones de causalidad se tienen dos alternativas: [20]

Alternativa 1: Se considera que dominar los elementos particulares tiene influencia causal en dominar el elemento general.

Alternativa 2: Se considera que dominar el elemento más general tiene influencia causal en dominar cada uno de los elementos específicos que lo componen.

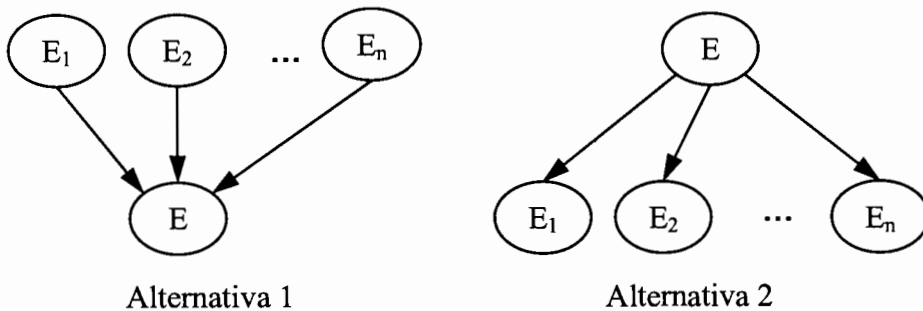


Figura 3.6: Alternativas de las relaciones de agregación.

Para cada una de las alternativas se tiene que:

a) En la alternativa 1 los parámetros a especificar serían las probabilidades a priori de conocer cada elemento, $\{P(E_i), i = 1, \dots, n\}$ y la distribución de probabilidad condicionada de E dados sus padres $P(E / \{E_1, \dots, E_n\})$ (en total, $n + 2^n - 1$ valores). Además esta red implicaría que los $E_i, i = 1, \dots, n$, son mutuamente independientes a priori.

b) En la alternativa 2 los parámetros a especificar serían: la probabilidad a priori de $E, p(E)$ y las distribuciones de probabilidad condicionada de los $\{P(E_i / E), i = 1, 2, \dots, n\}$ (un total, $2n + 1$ valores). En cuanto a independencias, esta estructura implica la independencia de los $E_i, i = 1, \dots, n$ conocido el valor de E .

3.4.1.1 – Diseño de la red bayesiana.

En la red bayesiana que se plantea se observa que hay tres partes que se solapan en nodos tipo concepto y en nodos tipo nivel de la red de diagnóstico (Ver figura 3.7):

- La parte donde aparecen las preguntas iniciales, el evaluador de nivel y los niveles de la *red de Evaluación de Nivel*, que se utilizará para, a partir de las preguntas iniciales, evaluar el nivel de conocimiento del alumno, para la obtención del nivel correspondiente según sus posibilidades.

- La parte en la que aparecen los nodos conceptos, preguntas y nivel de la *red de Diagnóstico*, que es la parte de la red bayesiana que se utilizará para, a partir del nivel y las respuestas del alumno, inferir qué conceptos domina y cuáles no domina.
- La parte en la que aparecen los nodos conceptos y temas de la *red Evaluación Final*, que se utilizará para, a partir de la probabilidad de saber cada concepto, proporcionar una calificación para cada tema y nivel en general.

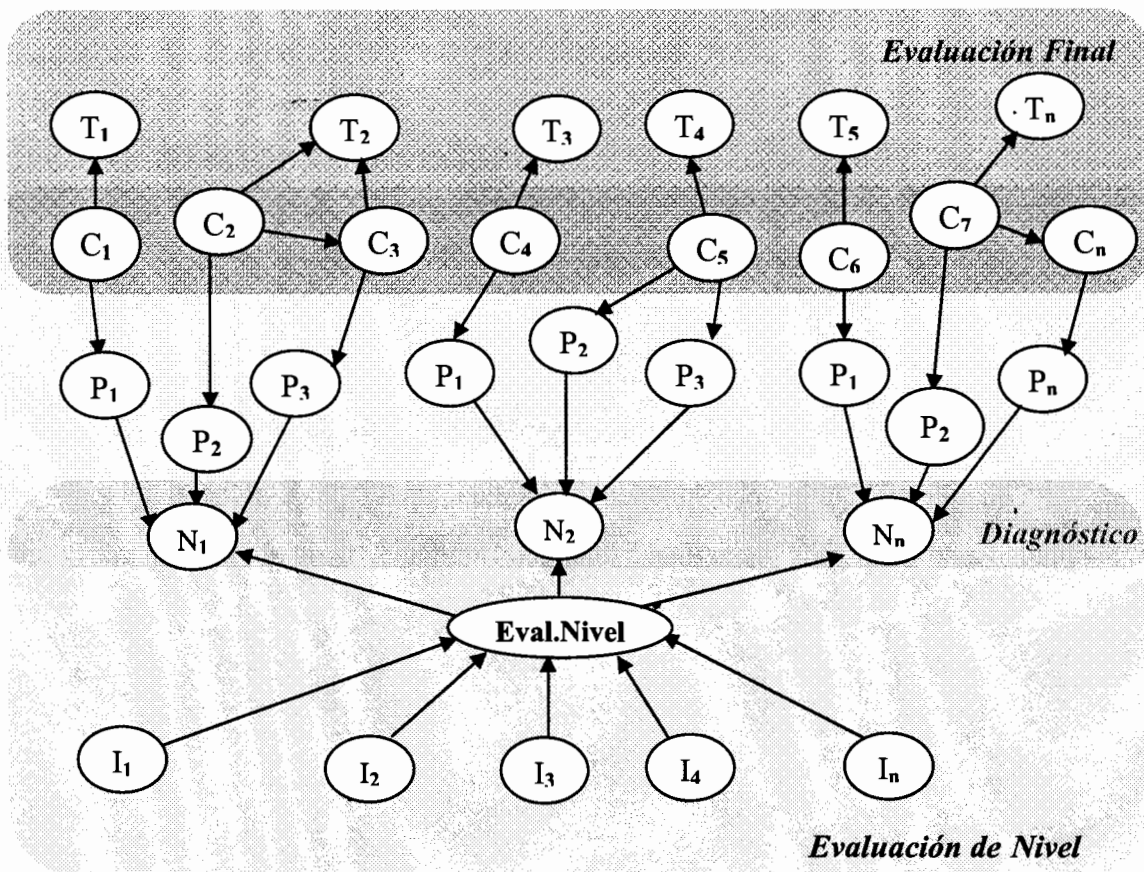


Figura 3.7: Diseño de la red bayesiana.

Cada una de estas partes se modela con un tipo de red bayesiana diferente: la parte de evaluación se modela con una red bayesiana clásica, mientras que para la parte de diagnóstico se utiliza una red bayesiana dinámica, puesto que en este caso es claro que los nodos tipo evidencia cambian con el

tiempo, es decir, el hecho de que un alumno conteste correctamente a una pregunta relacionada con ciertos conceptos no quiere decir que siempre que se plantee una pregunta relacionada con tales conceptos la contestará también correctamente.

3.4.2 – Especificación de parámetros.

Una vez determinado el tipo de nodos y las relaciones que se establecen entre ellos, es necesario especificar los parámetros. Es bien conocido que el problema de la especificación de los parámetros es uno de los problemas más difíciles en redes bayesianas. Para facilitar su obtención en este caso, se propone lo siguiente:

1. **La probabilidad a priori de saber cada concepto:** Si se dispone de alguna información del alumno que va a tomar el test, se puede utilizar esta información. En caso contrario, se puede decir que, para cada concepto, es igualmente probable que el alumno lo conozca o no, y, con este argumento, partir de una distribución uniforme que sería de 0,5.
2. **Para las relaciones de prerrequisito:** Se tiene que dar la probabilidad condicionada de conocer un concepto dado que se conocen o no sus prerrequisitos. Esta probabilidad será cero si alguno de los prerrequisitos no es conocido. En caso de que todos ellos sean conocidos, será un número que reflejará qué parte del conocimiento "nuevo" es distinta del conocimiento de sus prerrequisitos.
3. **Para las relaciones de agregación** (que se utilizarán en la evaluación del alumno): se propone la siguiente aproximación: el profesor dará un peso que cuantificará la importancia que tiene cada concepto dentro de cada tema. La probabilidad condicionada se construirá a partir de una suma ponderada de dichos pesos.

Para clarificar esto, se muestra un ejemplo sencillo:

Suponga que el tema T consta de los contenidos C_1 , C_2 y C_3 . Cada una de estas variables es un nodo en la red que toma dos posibles valores: *sí* o *no*. Así, la red resultante sería la que aparece en la Figura 3.8:

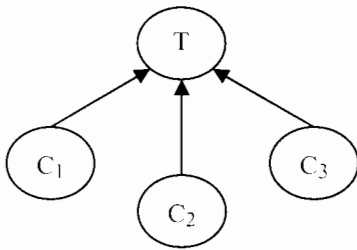


Figura 3.8: Relaciones concepto-tema.

Suponga que los pesos de dichos conceptos son 20, 30 y 50, respectivamente.

Para determinar las probabilidades $P(T/C_1, C_2, C_3)$ se utiliza una suma ponderada de los pesos de cada concepto según el concepto se domine o no, por ejemplo:

$$P(T/C_1 = sí, C_2 = no, C_3 = sí) = \frac{20 + 0 + 50}{20 + 30 + 50}$$

El comportamiento de la inferencia bayesiana en este caso es el siguiente:

Si se instancia todos los nodos concepto a sí, se obtiene $P(T=sí) = 1$, que debe interpretarse como que el alumno domina el 100% el tema.

Si se instancia por ejemplo los nodos concepto 1 y concepto 2 a sí, y el nodo concepto 3 a no, se obtiene $P(T=sí) = 0.5$, que debe interpretarse como que el alumno domina el 50% del tema.

De este modo, se consigue emular la forma de calificar de un profesor humano utilizando una red bayesiana.

Como se ha dicho antes, las relaciones de agregación se utilizarán cuando se desee evaluar al alumno en el tema y/o obtener una evaluación más detallada relativa a cada uno de los conceptos.

Se puede apreciar que a diferencia de la TRI, con redes bayesianas es posible tener estimaciones más precisas acerca de qué partes de un concepto o un tema domina o no el alumno, y esta información permitirá ofrecerle el mejor tipo de ayuda posible si nuestro sistema de evaluación se integra dentro de un sistema tutor.

4. En las relaciones tipo concepto-pregunta la cuestión es más delicada. En principio, para cada pregunta sería necesario especificar cuál es la probabilidad de responder correctamente a las preguntas

dadas todas las combinaciones posibles de conocer o no los conceptos necesarios para responder a esa pregunta. Esto llevaría a tener que pedirle al profesor probabilidades difíciles de estimar, puesto que si por ejemplo son cuatro los conceptos necesarios para contestar correctamente una pregunta el profesor debería dar 24 probabilidades (tantas como combinaciones de saber o no los conceptos). Así por ejemplo debería dar la probabilidad de que un alumno conteste bien a un problema de suma de fracciones dado que sabe como calcular el mínimo común múltiplo, sumar numeradores, pero no obtener la fracción equivalente ni simplificar el resultado.

Para evitar este problema, se propone la siguiente simplificación plantada por Eva Millán y que se piensa que sería útil aplicarla en nuestro modelo:

Parece claro que cuantos más conceptos domine el alumno entre los que son necesarios para contestar a la pregunta, mayor probabilidad tendrá de elegir la respuesta correcta (mejor descartará las incorrectas). Además, cuanto más importante sea o sean el o los conceptos que le faltan, menor probabilidad tendrá de elegir la respuesta correcta. Luego lo que se necesitará, para cada pregunta, es una curva que conecte de forma "suave" las probabilidad de acertar cuando no sabe nada ($1/n$ de respuestas posibles) con las probabilidad de acertar cuando sabe todo ($1-s$, donde s es la probabilidad de tener un fallo aleatorio, que podría ser común para todas las preguntas). [20]

Así, si le pedirá al profesor que ordene los conceptos necesarios por orden de importancia (el profesor puede también decir que considera que todos ellos son igualmente importantes, en cuyo caso la curva tendría trozos planos), se podría generar los parámetros necesarios evaluando en los puntos adecuados una función de tipo exponencial (como las CCI de la TRI clásica), como se representa en la Figura 3.9:

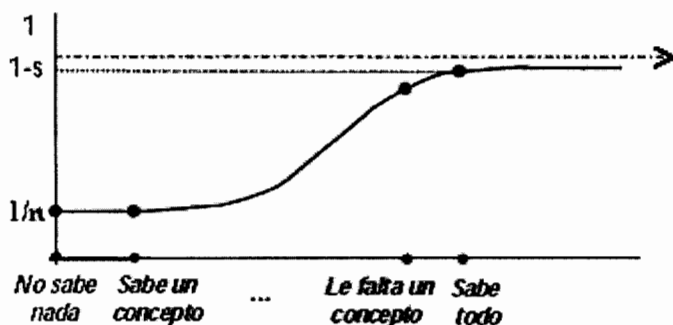


Figura 3.9: Obtención de las probabilidades condicionadas.

3.4.3 – Algoritmo de la red bayesiana.

El algoritmo propuesto para la red bayesiana planteada es el siguiente:

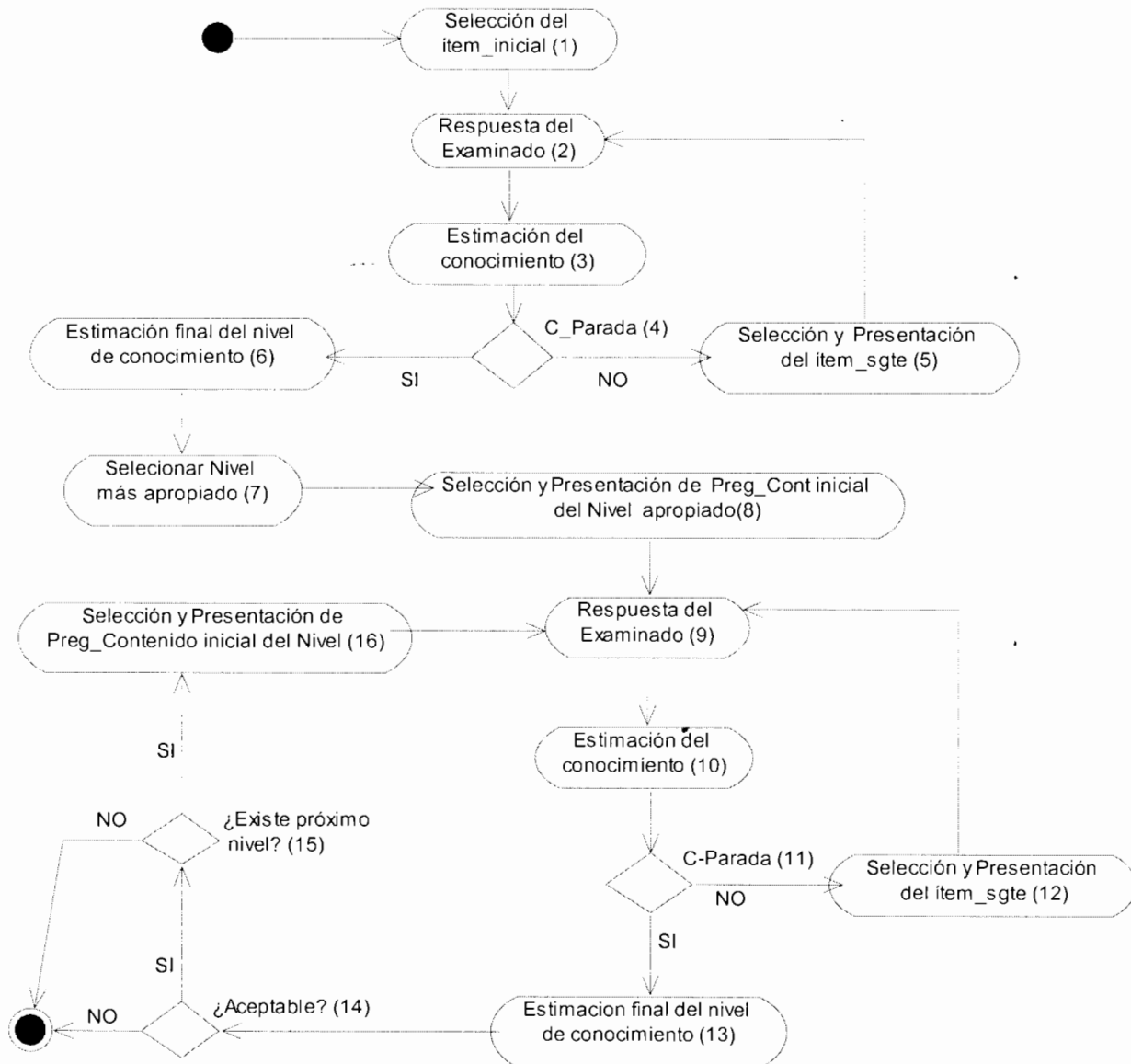


Figura 3.10: Algoritmo de la red bayesiana en general.

3.4.3 – Algoritmo de la red bayesiana.

El algoritmo propuesto para la red bayesiana planteada es el siguiente:

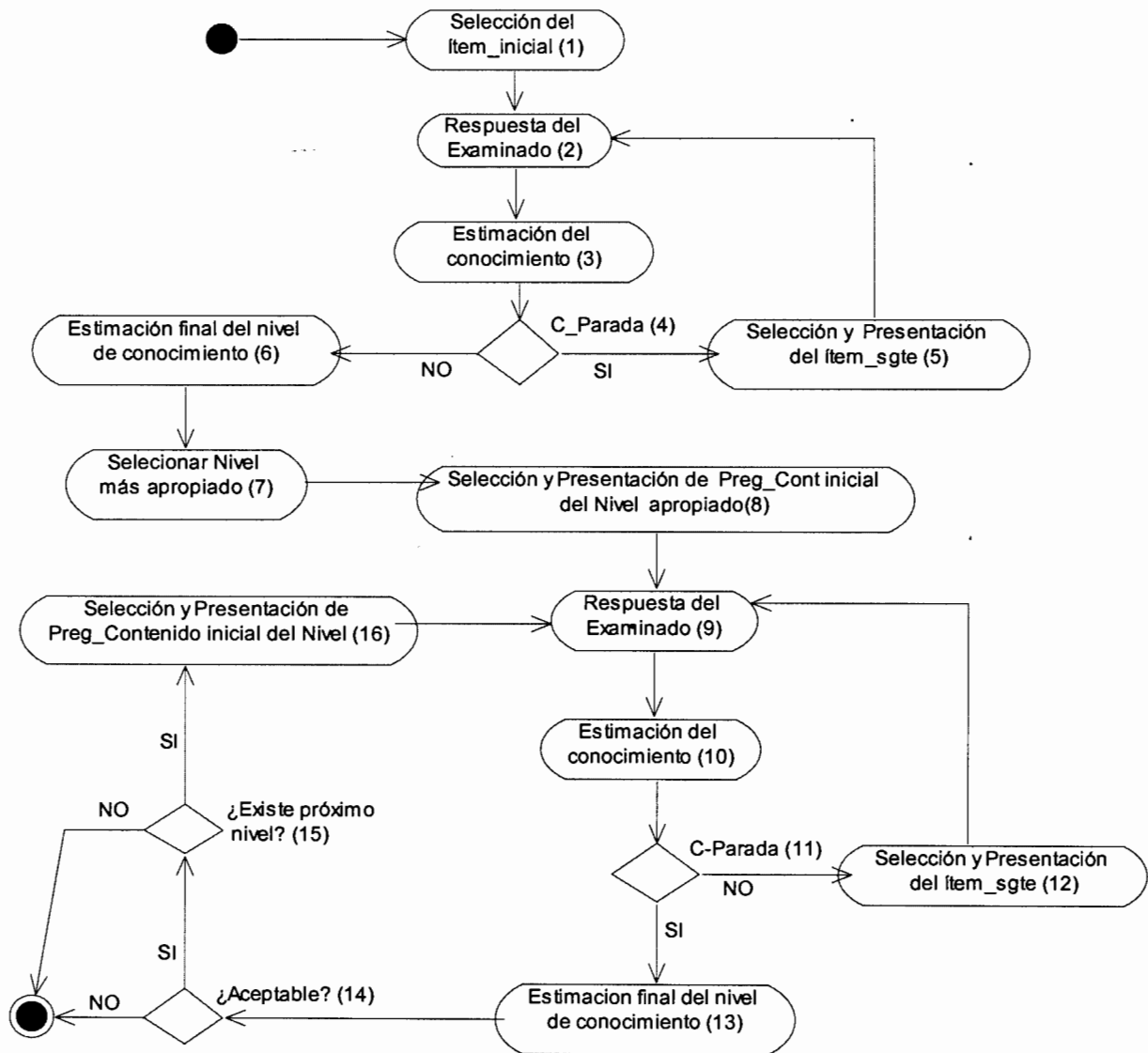


Figura 3.10: Algoritmo de la red bayesiana en general.

Este consta de dos partes fundamentales, la primera es la encargada de evaluar al estudiante en el tests inicial o de entrada y estimar un nivel de conocimiento acorde a los resultados obtenidos:

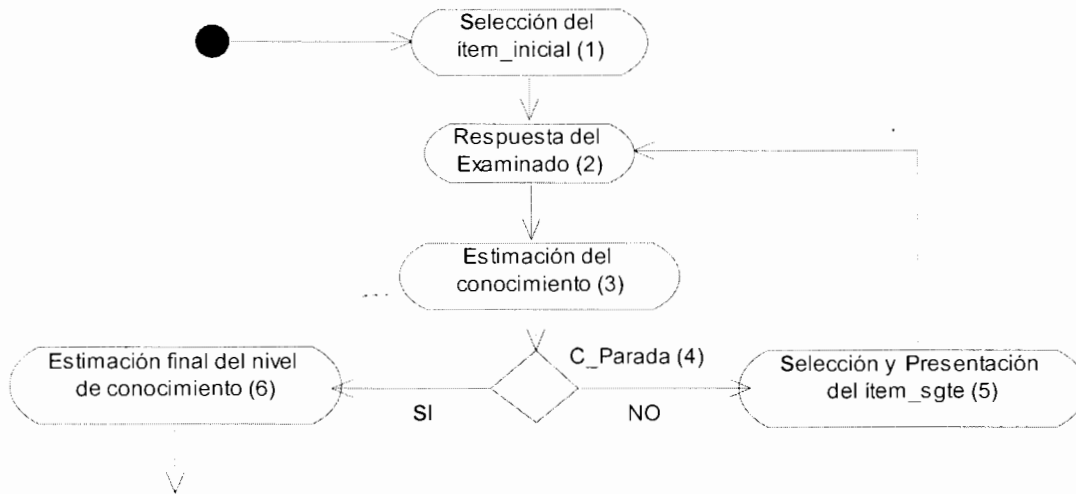


Figura 3.11: Algoritmo de la red bayesiana, tests inicial.

Los procesos 1 y 5 acceden a un *banco de preguntas*, a las cuales el profesor puede acceder para adicionar o eliminar preguntas a través del módulo de administración, mediante un *método de selección de preguntas* se realiza la más indicada. El proceso 1 tiene que escoger en dependencia del *nivel inicial* que se determine. El examinado responde en el proceso 2. La respuesta es comparada con el *modelo de respuesta* que está asociado a cada pregunta y por tanto forma parte del banco de preguntas. El proceso 3 estima el nivel de conocimiento del examinado según un *método de puntuación* y verifica el *criterio de parada* en el proceso 4 en caso de que no se haya cumplido el criterio se pasa al proceso 5, repitiéndose el ciclo, en caso contrario se estima el nivel de conocimiento por ultima vez y se pasa al proceso 7 donde se selecciona el nivel más apropiado para el estudiante según el criterio fijado por el profesor.

La otra parte es la encargada de dado el nivel estimado anteriormente, seleccionar y presentar las preguntas de contenido correspondientes y estimar el nuevo nivel, si este es aceptable, pasa a las preguntas de contenido del nivel superior, en caso contrario sale del sistema.

Este consta de dos partes fundamentales, la primera es la encargada de evaluar al estudiante en el tests inicial o de entrada y estimar un nivel de conocimiento acorde a los resultados obtenidos:

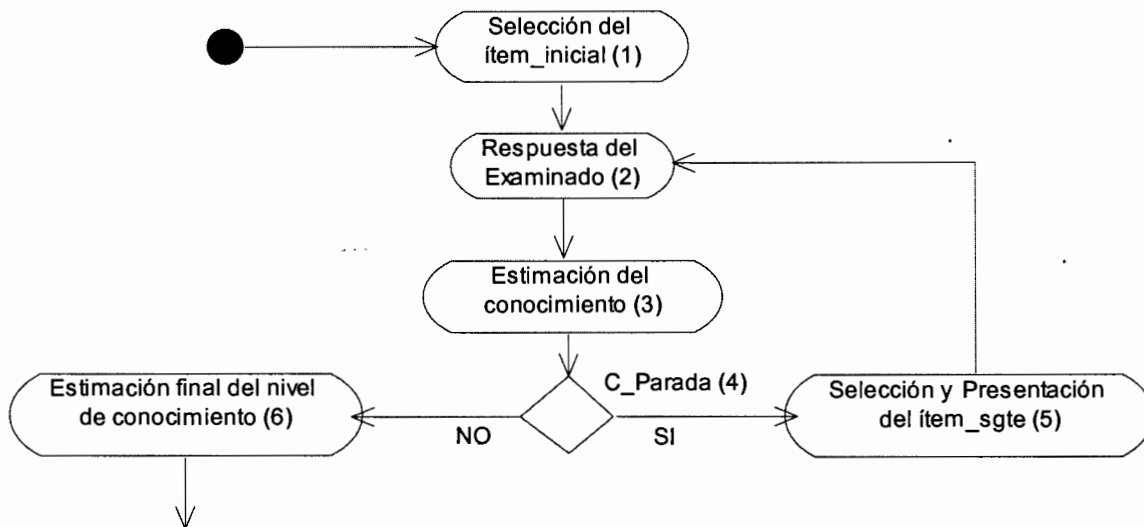


Figura 3.11: Algoritmo de la red bayesiana, tests inicial.

Los procesos 1 y 5 acceden a un *banco de preguntas*, a las cuales el profesor puede acceder para adicionar o eliminar preguntas a través del módulo de administración, mediante un *método de selección de preguntas* se realiza la más indicada. El proceso 1 tiene que escoger en dependencia del *nivel inicial* que se determine. El examinado responde en el proceso 2. La respuesta es comparada con el *modelo de respuesta* que está asociado a cada pregunta y por tanto forma parte del banco de preguntas. El proceso 3 estima el nivel de conocimiento del examinado según un *método de puntuación* y verifica el *criterio de parada* en el proceso 4 en caso de que no se haya cumplido el criterio se pasa al proceso 5, repitiéndose el ciclo, en caso contrario se estima el nivel de conocimiento por ultima vez y se pasa al proceso 7 donde se selecciona el nivel más apropiado para el estudiante según el criterio fijado por el profesor.

La otra parte es la encargada de dado el nivel estimado anteriormente, seleccionar y presentar las preguntas de contenido correspondientes y estimar el nuevo nivel, si este es aceptable, pasa a las preguntas de contenido del nivel superior, en caso contrario sale del sistema.

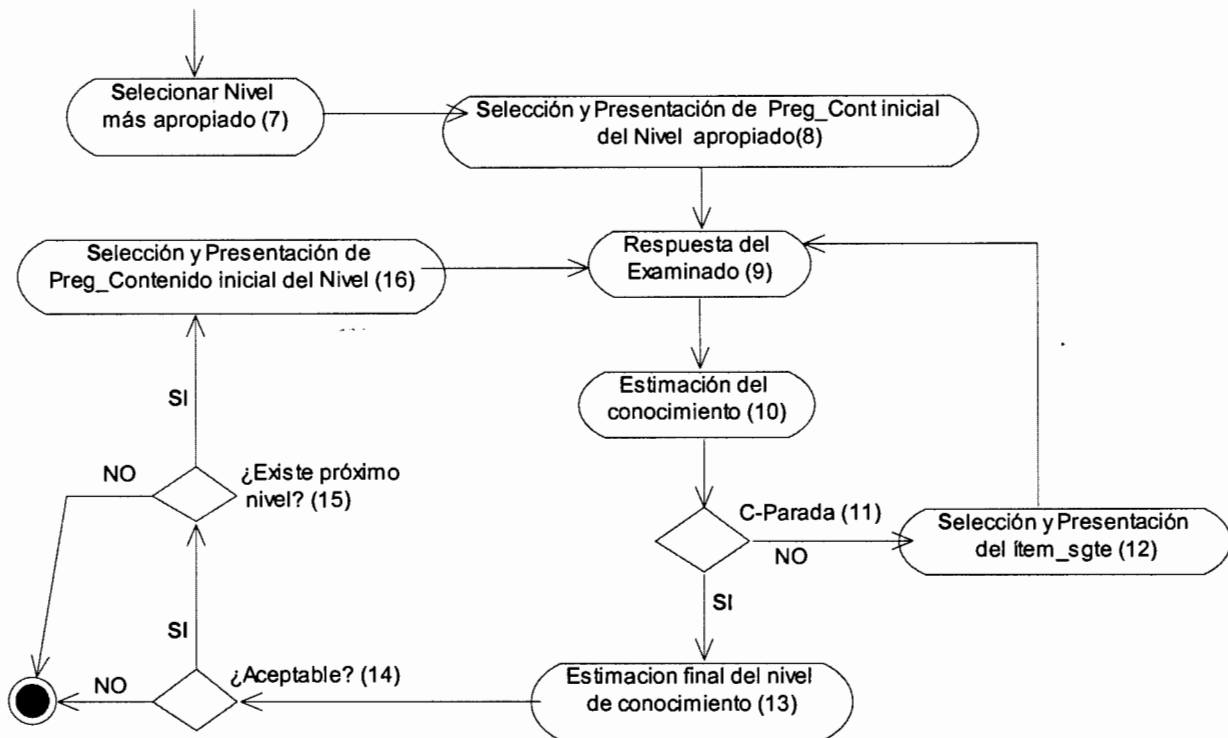


Figura 3.12: Algoritmo de la red bayesiana, preguntas de contenido.

Después de la selección del nivel más apropiado para medir el conocimiento de temas y contenidos específicos incluidos, se selecciona y presenta la pregunta de contenido inicial del nivel en el proceso 8, este proceso también accede al banco de preguntas al igual que el proceso 12. El examinado responde en el proceso 9. La respuesta es comparada con el *modelo de respuesta* que está asociado a cada pregunta. El proceso 10 estima el nuevo nivel de conocimiento del examinado según el *método de puntuación* y verifica el *criterio de parada* en el proceso 11, de no cumplirse se pasa al proceso 12, repitiéndose el ciclo, en caso contrario se estima el nivel final de conocimiento por ultima vez y se pasa al proceso 14, en este proceso se chequea si es aceptable o no el resultado para pasar al nivel inmediato superior, de no cumplirse se sale del sistema puesto que se estima que el nivel anterior lo cumplió y no está apto para el inmediato superior, en caso contrario se pasa al proceso 15 donde se chequea que exista un próximo

nivel, de no existir se sale del sistema, en caso contrario se pasa al proceso 16 donde se selecciona y presenta la pregunta inicial del nuevo nivel, luego se pasa al proceso 9 nuevamente y se repite todo el ciclo a partir de este proceso.

3.5 – Elementos básicos que deben estar presente en el desarrollo de la propuesta.

Ahora se definen algunos de los elementos fundamentales que se deben tener en cuenta a la hora de implementar esta propuesta con ayuda de UML, cabe destacar que esto es solo un acercamiento a lo que se quiere lograr con dicha red, aunque puede estar sujeto a cambios pues este proceso requiere de un estudio más profundo acorde a las características específicas del software a desarrollar.

3.5.1 – Captura de Requerimientos.

Es la primera etapa en el desarrollo de cualquier producto de software, donde se define y estudia el “QUE” del problema a resolver. En este caso, se plantean los requisitos funcionales y los requisitos no funcionales.

Requisitos funcionales.

1. Realizar autenticación de usuario.
 - 1.1. Verificar si es administrador o estudiante.
2. Registrar al usuario.
3. Permitir al administrador actualizar el sistema.
 - 3.1. Adicionar preguntas y respuestas.
 - 3.2. Modificar preguntas y respuestas.
 - 3.3. Eliminar preguntas y respuestas.
 - 3.4. Eliminar usuario.
4. Permitir al administrador generar reportes del estudiante.
 - 4.1. Nivel alcanzado.
 - 4.2. Preguntas resueltas.

5. Selección de las preguntas que debe responder el estudiante según su nivel de conocimiento.
 - 5.1. Permitir el acceso al banco de preguntas.
6. Presentación de las preguntas seleccionadas que debe responder el estudiante según su nivel de conocimiento.
 - 6.1. Garantizar la presentación de las preguntas al estudiante después del proceso de selección.
7. Estimar nivel de conocimiento del estudiante dado una o varias preguntas.
 - 7.1. Comparar la respuesta del estudiante con el modelo establecido.
 - 7.2. Verificar criterio de parada.
 - 7.3. Asignar al estudiante el nivel estimado.
8. Chequear nivel del estudiante.
 - 8.1. Chequear si es aceptable el nivel actual para pasar al nivel superior.
 - 8.2. Chequear si existe el próximo nivel para continuar con la evaluación.
9. Mostrar al estudiante su nivel actual a modo de información.

Requisitos no funcionales.

El área de programación del proyecto se encarga de analizar y determinar los requerimientos no funcionales aunque se plantea como propuesta los siguientes:

Requisito de usabilidad.

1. Tiempo de entrenamiento para usuarios del sistema.
 - 1.1. Debe planificarse un tiempo de entrenamiento de 2 semanas para usuarios del sistema.

Requisitos de performance

1. Capacidad
 - 1.1. El sistema podrá hospedar alrededor de 500 usuarios.
2. Utilización de recursos
 - 2.1. El sistema utilizará almacenamiento duro para guardar todas las informaciones de los usuarios y todo el contenido de la aplicación, en general.

Requisitos de seguridad

1. Garantizar que la información sea vista de acuerdo a los roles establecidos en el sistema.
2. Garantizar que las funcionalidades del sistema se muestren de acuerdo al nivel del usuario que esté activo.
3. Protección contra acciones no autorizadas o que puedan afectar la integridad de los datos.

Restricciones de diseño

1. Herramientas de desarrollo gráfico.
 - 1.1. Se utiliza el Rational Rose como herramienta de desarrollo gráfico para la modelación del sistema.

Requisitos de ayudas del sistema.

1. Ayuda del sistema
 - 1.1. Se implementará una sección de ayuda para los usuarios la cual trate el funcionamiento del sistema y el modo de trabajarlo.

Interfaces

1. Interfaz de usuario
 - 1.1. Se define una *Portada Estándar para Usuarios No Registrados* la cual será el punto de entrada de todos los usuarios del sistema. .

Requisitos de confiabilidad

1. Disponibilidad
El sistema estará disponible durante las horas clases, tanto para el trabajo de los usuarios como para las acciones de mantenimiento.
2. Tiempo medio de reparación.
 - 2.1 El sistema debe ser capaz de recuperarse en tan solo segundos de los fallos operacionales.

3.5.2 – Casos de uso.

Un caso de uso es en esencia, una interacción típica entre un usuario y un sistema de cómputo. Dentro de las propiedades de los casos se pueden citar:

- El caso de uso capta alguna función visible para el usuario.
- El caso de uso puede ser pequeño o grande.
- El caso de uso logra un objetivo discreto para el usuario.

En su forma más simple, el caso de uso se obtiene hablando con los usuarios habituales y analizando con ellos las distintas cosas que deseen hacer con el sistema.

Para abordar cada caso concreto se le debe dar un nombre, y describirlo brevemente. Durante la elaboración esto es todo lo que se necesita para empezar.

En la identificación de casos de uso existe diferencia entre los objetivos del usuario y sus interacciones con el sistema. Estos casos de uso reflejan las cosas que puede hacer el usuario con el sistema en vez de apoyar lo que realmente quiere y necesita.

3.5.3 – Elementos de los casos de uso.

Actores: en este caso, los actores va a ser los alumnos o los profesores o ambos en tanto interactúen con el sistema. Cuando se identifican los actores conviene enfatizar los papeles que asumen.

Los actores llevan a cabo casos de uso. De hecho un mismo actor puede realizar muchos casos de uso y a la inversa, un caso de uso puede ser atendido por varios actores. Los actores pueden ser seres humanos o sistemas externos que necesiten datos del sistema actual.

3.5.4 – Otras herramientas de la notación UML.

Existen además otras herramientas como:

- Los diagramas de clase, que describen los tipos de objetos que hay en el sistema y las diversas clases de relaciones estáticas que existen entre ellos.
- Los diagramas de interacción, que describen la manera en que interactúan grupos de objetos para lograr cierto comportamiento.
- Los diagramas de paquetes que muestran los paquetes de clases y las dependencias entre ellos
- Los diagramas de estados, que describen el comportamiento del sistema.

- Los diagramas de actividades, útiles en conexión con el flujo de trabajo y para la descripción del comportamiento que tiene gran cantidad de proceso paralelo.
- Los diagramas de componentes, que muestran las relaciones físicas entre los componentes de software y hardware en un sistema.

Estas herramientas junto a otras existentes se desarrollarán en estudios posteriores para el desarrollo e implementación del software.

De acuerdo a los flujos de trabajo o pasos sugeridos por la metodología elegida, se detallan algunos casos de uso que debe requerir el sistema.

3.5.5 – Modelo de casos de uso del sistema.

Que un proyecto de desarrollo está dirigido por los casos de uso significa que progresa a través de una serie de flujos de trabajo que se inician a partir de los casos de uso. Los casos de uso ayudan a los desarrolladores a encontrar las clases. Las clases se recogen en las descripciones de los casos de uso a medida que las leen los desarrolladores, buscando que sean adecuadas para la realización de los casos de uso. Estos además nos ayudan a desarrollar interfaces de usuario que hacen más fácil a los usuarios el desempeño de sus tareas.

El Objetivo principal de ésta primera etapa es describir lo que el software hará y permitir a los desarrolladores y usuarios estar de acuerdo con dicha descripción. Se deben documentar los requerimientos de manera que todo el grupo de proyecto los entienda.

A continuación se muestra un diagrama general de casos de uso, en términos de UML, que incluye los principales casos de uso que el usuario debe poder acceder en el software educativo.

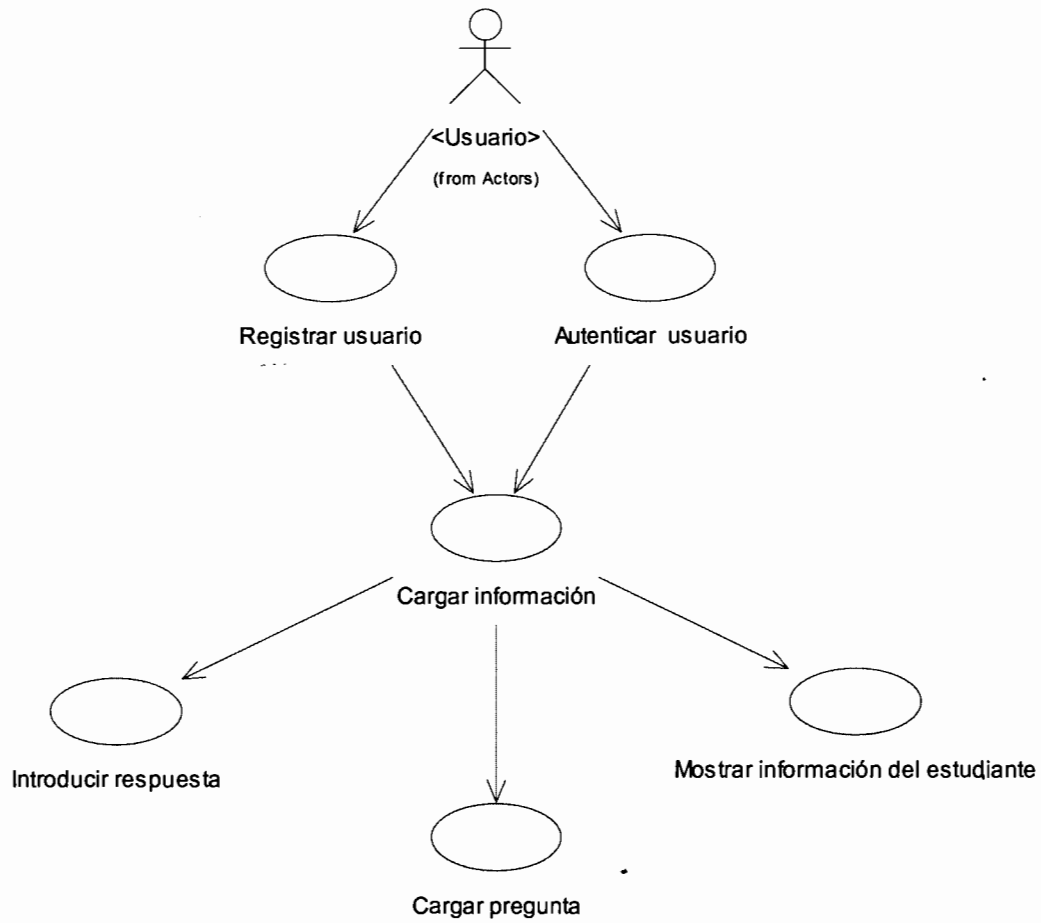


Figura 3.13: Diagrama de casos de uso del sistema.

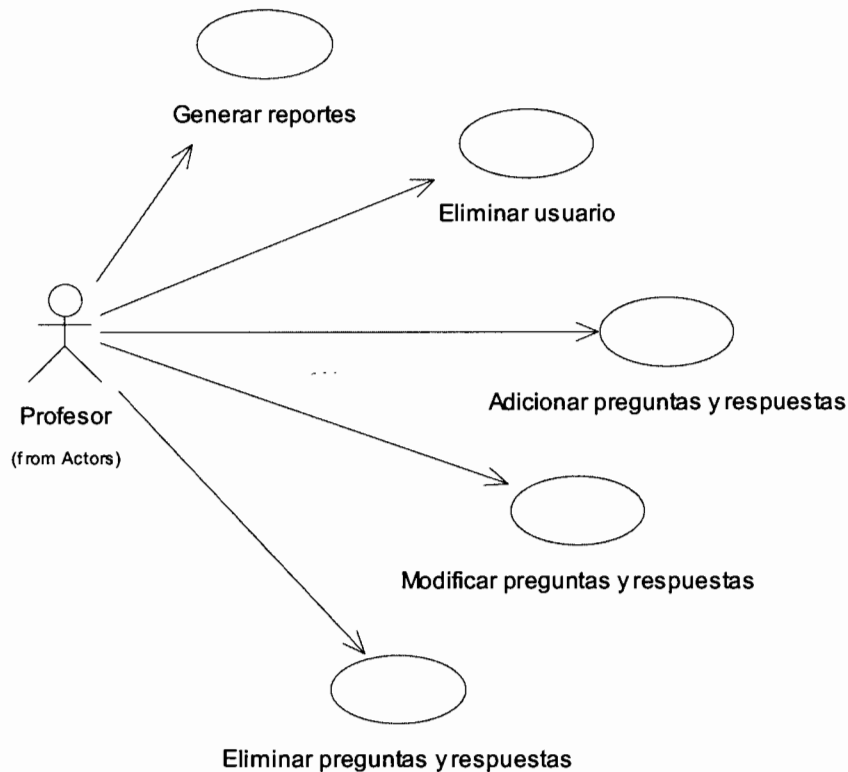


Figura 3.14: *Diagrama de casos de uso del profesor.*

El modelo de casos de uso ayuda al cliente, a los usuarios y a los desarrolladores a llegar a un acuerdo sobre como utilizar el sistema. La mayoría de los sistemas tiene muchos tipos de usuarios. Cada tipo de usuario se representa mediante un actor. En este caso interactúan dos actores con el sistema:

Actores	Justificación
Profesor	Es la persona encargada de administrar el sistema, para ello puede insertar, modificar o eliminar preguntas y respuestas del banco de preguntas.
Estudiante	Es la persona que mide su nivel de conocimiento con ayuda del sistema.

3.5.6 – Expansión de los Casos de Uso.

Mediante los casos de uso expandidos se describe la secuencia de eventos que los actores deben utilizar para interactuar con el del sistema. A continuación se muestra la explicación de cada caso de uso.

Nombre del caso de uso		Autenticar usuario
Actores	Estudiante, Profesor	
Propósito	Verificar que el usuario esté registrado en el sistema para posibilitar su entrada en el sistema y acceder a las opciones disponibles según su rol.	
Resumen: El caso de uso se inicia cuando el usuario entra su login y password, el sistema verifica que el usuario está registrado, de ser correcto lo autentica. De esta manera realiza las funciones definidas de acuerdo al rol asignado en el sistema.		
Referencias	R1	
Precondiciones	Que el usuario haya accedido al sistema	
Poscondiciones	Ha quedado registrado un usuario del sistema	
Requisitos especiales		

Nombre del caso de uso		Registrar usuario
Actores	Estudiante, Profesor	
Propósito	Registrar al usuario en la base de datos para que posteriormente el sistema posibilite su entrada en el sistema.	
Resumen: El caso de uso se inicia cuando el usuario entra su login y password, el sistema registra al nuevo usuario en la base de datos, de esta manera puede autenticarse posteriormente y acceder a los servicios del sistema.		
Referencias	R2	
Precondiciones	Que el usuario no esté registrado	
Poscondiciones	Se ha adicionado un nuevo usuario al sistema	
Requisitos especiales		

Nombre del caso de uso	Generar reportes
Actores	Profesor
Propósito	Emitir criterio sobre un alumno o varios a partir de los resultados alcanzados en el sistema.
Resumen: El caso de uso se inicia cuando el profesor desea ver los resultados de los alumnos en un determinado nivel, accediendo a los reportes almacenados.	
Referencias	R4
Precondiciones	Que el actor se haya autenticado en el sistema. Tiene que haberse evaluado el o los alumnos solicitados en el sistema.
Poscondiciones	Se ha mostrado los reportes de los estudiantes que han sido evaluados en el sistema.
Requisitos especiales	

Nombre del caso de uso	Adicionar preguntas y respuestas
Actores	Profesor
Propósito	Actualizar el banco de preguntas y respuestas.
Resumen: El caso de uso se inicia con el acceso al banco de preguntas para actualizarlo a través de la adición de preguntas y respuestas.	
Referencias	R3, R 3.1
Precondiciones	Que el actor se halla autenticado en el sistema y posea permiso de administración.
Poscondiciones	Se han adicionado al banco de preguntas y respuestas las nuevas preguntas y respuestas.
Requisitos especiales	

Nombre del caso de uso	Eliminar preguntas y respuestas
Actores	Profesor
Propósito	Actualizar el banco de preguntas y respuestas
Resumen: El caso de uso se inicia con el acceso al banco de preguntas para actualizarlo a través de la eliminación de preguntas y respuestas ya que pueden haber quedado obsoletas.	
Referencias	R3, R 3.2
Precondiciones	Que el actor se halla autenticado en el sistema y posea permiso de administración.
Poscondiciones	Las preguntas y respuestas seleccionadas se han eliminado y queda actualizado el banco de preguntas y respuestas.
Requisitos especiales	

Nombre del caso de uso	Modificar preguntas y respuestas
Actores	Profesor
Propósito	Actualizar el banco de preguntas y respuestas
Resumen: El caso de uso se inicia con el acceso al banco de preguntas para actualizarlo a través de la modificación de preguntas y respuestas para mejorar así el nivel de rigurosidad de las mismas.	
Referencias	R3, R 3.3
Precondiciones	Que el actor se halla autenticado en el sistema y posea permiso de administración.
Poscondiciones	Se ha actualizado el banco de preguntas y respuestas.
Requisitos especiales	

Nombre del caso de uso		Eliminar usuario
Actores	Profesor	
Propósito	Eliminar un usuario registrado en el sistema	
Resumen: El caso de uso se inicia con la entrada del profesor al registro de usuarios donde selecciona y elimina al usuario deseado.		
Referencias	R 3, R 3.4	
Precondiciones	Que el actor se halla autenticado en el sistema y posea permiso de administración.	
Poscondiciones	Se ha eliminado el usuario seleccionado.	
Requisitos especiales		

Nombre del caso de uso		Mostrar información del estudiante
Actores	Estudiante	
Propósito	Mostrar el nivel que posee el estudiante	
Resumen: El caso de uso se inicia cuando el estudiante solicita saber su nivel actual de conocimiento al sistema.		
Referencias	R9	
Precondiciones	Que el actor se halla evaluado en el sistema.	
Poscondiciones	Se ha mostrado la información.	
Requisitos especiales		

Nombre del caso de uso		Cargar pregunta.
Actores	Estudiante	
Propósito	Mostrar la pregunta a resolver por el estudiante.	
Resumen: El caso de uso se inicia cuando el estudiante entra al sistema e inicia la evaluación o continúa la misma, el sistema le muestra la pregunta a resolver.		
Referencias	R 6.1	

Precondiciones	Que el usuario se haya autenticado en el sistema.
Poscondiciones	Se ha mostrado la pregunta correspondiente al estudiante.
Requisitos especiales	

Nombre del caso de uso	Introducir respuesta.
Actores	Estudiante
Propósito	Introducir respuesta del estudiante.
Resumen: El caso de uso se inicia cuando el estudiante selecciona la respuesta que estima correcta, el sistema chequea si es correcta o no.	
Referencias	R 7.1
Precondiciones	Que el usuario se haya autenticado en el sistema.
Poscondiciones	Se ha introducido la respuesta del estudiante.
Requisitos especiales	

Nombre del caso de uso	Cargar información.
Actores	Estudiante, Profesor
Propósito	Mostrar la pregunta a resolver, introducir respuesta o mostrar información del usuario.
Resumen: El caso de uso se inicia cuando el actor se autentica en el sistema y solicita un determinado servicio del sistema.	
Referencias	R6, R7, R9
Precondiciones	Que el usuario se haya autenticado en el sistema.
Poscondiciones	Se ha mostrado la información solicitada.
Requisitos especiales	

En estudios posteriores se debe desarrollar el diseño e implementación de la propuesta teniendo en cuenta lo antes expuesto, para lograr que se cumplan los objetivos del algoritmo inteligente que se propone en este trabajo investigativo.

3.6 – Conclusiones.

En este capítulo se ha definido un modelo estructural que permite utilizar las redes bayesianas para la formulación y evaluación de test adaptativos computerizados además, cuáles son las variables que deben formar parte de dicho modelo y los tipos de relaciones que han de establecerse entre ellas. También se ha abordado lo que se considera uno de los problemas principales en el uso de redes bayesianas: la especificación de los parámetros. En efecto, la gran dificultad de obtener una buena estimación de los parámetros es la principal causa esgrimida por muchos investigadores para no hacer uso en sus aplicaciones de este sólido y bien fundamentado modelo. Se cree que con esta aproximación es posible simplificar este problema y, de este modo, hacer más accesible la utilización de las redes bayesianas en el contexto de los test adaptativos computerizados y lograr así un algoritmo que propicie la individualización del aprendizaje de los alumnos.

CONCLUSIONES.

En las páginas precedentes se ha expuesto una visión panorámica de la teoría con la finalidad de proporcionar herramientas de modelado robustas y bien fundamentadas en el campo de los STI. Con esto se ha mostrado que tanto la teoría clásica de la TRI como las más recientes redes bayesianas son un punto de partida adecuado para ello.

Existen tipos de softwares educativos que se emplean solo para evaluar al estudiante, aunque la tendencia actual es construir softwares educativos que funcionen como tutoriales, simuladores y evaluadores, es decir sistemas que presenten el contenido, que simulen si es necesario y que evalúen el desempeño del estudiante. Por otro lado se puede plantear que el módulo de evaluación de estos sistemas constituye una herramienta de vital importancia debido a que al evaluar el desempeño del estudiante se pueden trazar estrategias pedagógicas, claras y objetivas, encaminadas a satisfacer las necesidades docentes de los estudiantes.

Las redes bayesianas proveen una serie de importantes ventajas a partir de su característica de constituir una mezcla de técnicas estadísticas y de modelos gráficos. El hecho de que las redes guarden información sobre las dependencias e independencias existentes entre las variables involucradas permite manejar situaciones donde exista incertidumbre. Por otro lado, su presentación gráfica de la red facilita la interpretación y obtención de conclusiones sobre el tema en estudio por parte de sus lectores. Además, debido a que estas redes combinan relaciones causales con lógica probabilística, permite combinar conocimiento experto con datos. Además permiten definir modelos y utilizarlos tanto para hacer razonamiento de diagnóstico, como para hacer razonamiento predictivo.

Por último, cabe destacar de nuevo que con este modelo se abre la posibilidad de obtener una evaluación más completa de los alumnos que la que se obtenía con los test basados en la TRI, lo cual hace que este tipo de evaluación mediante test fuese muy adecuada para integrarla con un Sistema Tutor Inteligente y por ende a los software educativos que ya se están produciendo en nuestras jóvenes casas de software en el país; pero que presentan varios inconvenientes a la hora de evaluar al estudiante de manera diferenciada, que pueden ser solucionados con la solución planteada, en aras de lograr un mejor aprendizaje de los estudiantes en general.

RECOMENDACIONES.

Se recomienda:

- Continuar con la investigación para mejorar la clasificación del aprendizaje del alumno de manera personalizada.
- Realizar la ingeniería del software educativo que utilizaría el algoritmo.
- Implementación del algoritmo en los softwares educativos existentes en el país.
- Extender este algoritmo a otras ramas donde se necesite un sistema de clasificación probabilístico fuera del campo de la educación.

REFERENCIAS BIBLIOGRÁFICAS.

- [1] BECK; STERN, *et al.* *Using the Student Model to Control Problem Difficulty*, 1997. [2005]. Disponible en: <http://polaris.lcc.uma.es/~eva/investigacion/SBMA.pdf>
- [2] BEINLICH, I. A., SUERMONDT, H.J., CHAVEZ, R.M+, COOPER, G.F. *The Alarm monitoring system: A case study with two probabilistic inference techniques for belief networks*. 2nd European Artificial Intelligence in Medicine, 1989. p.
- [3] BERRIOS, G. *Definición de Informática Educativa*, 2001. [2005]. Disponible en: <http://mipagina.cantv.net/gersonberrios/>
- [4] BIRNBAUM, A. *Some latent trait models and their use in inferring an examinee's mental ability*. In F. M. Lord, & M. R. Novick (eds), *Statistical theories of mental test scores*. Reading, MA: Addison-Wesley (1968).
- [5] BREESE, J. S., BLAKE, RUSS. *Automating Computer Bottleneck Detection with Belief Nets*. Conference on Uncertainty in Artificial Intelligence,, Morgan Kaufmann, San Francisco,, 1995. 45 p.
- [6] CONATI, C. and K. V. LENH. *POLA: A student modeling framework for probabilistic on-line assessment of problem solving performance*. 5th International Conference on User Modeling UM'96, 1996. 75-82 p.
- [7] CUBINO, R. L. *La evaluación en el área de tecnología*. Amaru. Salamanca, 1998. 194 p. 8481960926.
- [8] DIAZ, R. P. and L. F. NIÑO. *Curso de Ingeniería*, UNIVIRTUAL, 2004. [2005]. Disponible en: <http://www.virtual.unal.edu.co/cursos/ingenieria/2001832/index.html>

- [9] DUSSAILLANT, F. *Técnicas de medición en pruebas de admisión a las universidades*. Chile, 2003. 2005.
- [10] EZAWA, K. J., SCHUERMANN, *Fraud/Uncollectible Debt Detection Using a Bayesian Network Based Learning System: A Rare Binary Outcome with Mixed Data Structures*. Conference on Uncertainty in Artificial Intelligence, Morgan Kaufmann, San Francisco, 1995. 157-166 p.
- [11] FELGAER, P. E. *Optimización de redes bayesianas basado en técnicas de aprendizaje por inducción*. Argentina, Universidad de Buenos Aires, 2005. p.
- [12] FLAUGHER, R. Item Pools. In H. Wainer (ed.), *Computerized Adaptive Testing: A Primer*. Hillsdale, NJ: Lawrence Erlbaum Associates Publishers (1990).
- [13] GONZ, W. *Inteligencia Artificial. Algunas Definiciones*, Answer Math, 2004. [2005]. Disponible en: <http://www.answermath.com/inteligencia-artificial.htm>
- [14] GÜRER; DESJARDINS, et al. *Representing a Student's Learning States and Transitions.*, 1995. [2005]. Disponible en: <http://polaris.lcc.uma.es/~eva/investigacion/SBMA.pdf>
- [15] HAMBLETON, R. K. *Principles and selected applications of Item Response Theory*. In R. L. Linn (ed.), *Educational Measurement*. New York: MacMillan (1989).
- [16] JIMENEZ, J. A. *Introducción a la Inteligencia Artificial*, Dpto. Ciencias de la Computación de Inteligencia Artificial, 2004. [2005].
Disponible en: <http://www.cs.us.es/cursos/iia-2004/temas/introduccion.pdf>
- [17] KATZ and LESGOLD. *Modelling the student in SHERLOCK II.*, 1994. 150 p.
- [18] MARQUÉS, P. *El software educativo*, 1996. [2005].
Disponible en: http://www.filos.unam.mx/POSGRADO/seminarios/pag_robertp/paginas/soft_edu.html

- [19] MARTIN, J. and K. VANLEHN. *Student assessment using Bayesian nets*, 1995. [2005]. Disponible en: <http://www.pitt.edu/~vanlehn/distrib/journal/HCS95.pdf>
- [20] MILLAN, E., JOSE L, PEREZ. , CONEJO, R. *Modelado del alumno*. en. España, Málaga, 2004. 9.p.
- [21] MILLAN, E., JOSE L, PEREZ., CONEJO, R. *Uso de redes bayesianas en test adaptativos informatizados*. en. España, Málaga, 2003. 9.p.
- [22] MILLAN, E. *Sistema bayesiano para el modelado del alumno*. Málaga. Universidad de Málaga, 2000. 195 p.
- [23] MISLEVY, R. and D. GITOMER. *The Role of Probability-Based Inference in an Intelligent Tutoring System*, 1996. [2005]. Disponible en: <http://polaris.lcc.uma.es/~eva/investigacion/SBMA.pdf>
- [24] PÉREZ, V., EIRIZ, O., CORREDERAS, G., LÓPEZ, M. *Folleto del curso Informática educativa. Maestría en pedagogía profesional*.: Instituto Superior Pedagógico para la Educación Técnica y Profesional. La Habana., 1997. 28. p.
- [25] PORTO, E. R. *Las computadoras en la Educación*, Horizonte, 2001. [2005]. Disponible en: <http://www.horizonteweb.com/sitios.htm>
- [26] RASCH, G. *Probabilistic models for some intelligence and attainment test*. Copenhagen: Danish Institute for Educational Research (1960).
- [27] SALINAS, G. K. *Red Escolar de Informática Educativa. Revista Red Escolar, 2000*. vol 2. número 7. Disponible en: <http://redescolar.ilce.edu.mx/redescolar/Revista/07/index.html>
- [28] SUCAR, L. E. *Razonamiento con incertidumbre*, ITESM, 2003. [2005]. Disponible en: <http://dns1.mor.itrsm.mx/esucar/incertidumbre.html>

- [29] THISSSEN, D, & MISLEVY, R. *Testing Algorithms*. In H. Wainer (ed.), *Computerized Adaptive Testing: A Primer*. Hillsdale, NJ: Lawrence Erlbaum Associates Publishers (1990).
- [30] WAINER, H., & MISLEVY, R. *Item Response Theory, Item Calibration and Proficiency Estimation*. En H. Wainer (ed.), *Computerized Adaptive Testing: A Primer* Hillsdale, NJ: Lawrence Erlbaum Associates Publishers.1990. 65-102 p.

BIBLIOGRAFÍA.

1. BECK; STERN, *et al.* *Using the Student Model to Control Problem Difficulty*, 1997. [2005]. Disponible en: <http://polaris.lcc.uma.es/~eva/investigacion/SBMA.pdf>
2. BEINLICH, I. A., SUERMONDT, H.J., CHAVEZ, R.M., COOPER, G.F. *The ALARM monitoring system: A case study with two probabilistic inference techniques for belief networks*. 2nd European Artificial Intelligence in Medicine, 1989: p.
3. BERRIOS, G. *Definición de Informática Educativa*, 2001. [2005]. Disponible en: <http://mipagina.cantv.net/gersonberrios/>
4. BIRNBAUM, A. *Some latent trait models and their use in inferring an examinee's mental ability*. In F. M. Lord, & M. R. Novick (eds), *Statistical theories of mental test scores*. Reading, MA: Addison-Wesley (1968).
5. BREESE, J. S., BLAKE, RUSS. *Automating Computer Bottleneck Detection with Belief Nets*. Conference on Uncertainty in Artificial Intelligence, Morgan Kaufmann, San Francisco, 1995. 45 p.
6. CHELAPA, JAIN. *Markov Random Fields: Theory and Models*. Academic Press.
7. CONATI, C. and K. V. LENH. *POLA: A student modeling framework for probabilistic on-line assessment of problem solving performance*. 5th International Conference on User Modeling UM'96, 1996. 75-82 p.
8. CONEJO, R., MILLÁN, E., PÉREZ-DE-LA-CRUZ, J. L., & Trelia, M. An Empirical Approach to On-Line Learning in SIETTE. In *Proceedings of 3rd International Conference on Intelligent Tutoring Systems ITS'2000*. LNCS 1839, Springer Verlag (2000).

9. Ramírez, C, Martínez, N. *Un Algoritmo para generar Redes Bayesianas a partir de datos estadísticos*. Universidad Tecnológica de la Mixteca, Oaxaca, 2000.
10. CUBINO, R. L. *La evaluación en el área de tecnología*. Amaru. Salamanca, 1998. 194 p. 8481960926
11. DIAZ, R. P. and L. F. NIÑO. *Curso de Ingeniería*, UNIVIRTUAL, 2004. [2005]. Disponible en: <http://www.virtual.unal.edu.co/cursos/ingenieria/2001832/index.html>
12. DUSSAILLANT, F. *Técnicas de medición en pruebas de admisión a las universidades*. Chile, 2003.
13. EZAWA, K. J., SCHUERMAN, . *Fraud/Uncollectible Debt Detection Using a Bayesian Network Based Learning System: A Rare Binary Outcome with Mixed Data Structures*. Conference on Uncertainty in Artificial Intelligence, Morgan Kaufmann, San Francisco, 1995. 157-166 p.
14. FELGAER, P., BRITOS, P., SICRE, J., SERVETTO, A., GARCÍA-MARTÍNEZ, R., PERICHINSKY, G. (2003). *Optimización de redes bayesianas basado en técnicas de aprendizaje por inducción*. IX Congreso Argentino de Ciencias de la Computación. La Plata. Octubre 6 al 10.
15. FELGAER, P. E. *Optimización de redes bayesianas basado en técnicas de aprendizaje por inducción*. Argentina, Universidad de Buenos Aires, 2005. p.
16. FLAUGHER, R. Item Pools. In H. Wainer (ed.), *Computerized Adaptive Testing: A Primer*. Hillsdale, NJ: Lawrence Erlbaum Associates Publishers (1990).
17. GARCÍA-MARTÍNEZ, R. *Sistemas Autónomos. Aprendizaje Automático*. Buenos Aires. Editorial Nueva Librería. (1997). 170 p. ISBN 950-9088-84-6.
18. GARCÍA-MARTÍNEZ, R., SERVENTE, M., PASQUINI, D. *Sistemas Inteligentes*. Buenos Aires. Editorial Nueva Librería. (2003). 347 p. ISBN 987-1104-05-7.

19. GÜRER; DESJARDINS, *et al.* *Representing a Student's Learning States and Transitions.*, 1995. [2005]. Disponible en: <http://polaris.lcc.uma.es/~eva/investigacion/SBMA.pdf>
20. HAMBLETON, R. K. *Principles and selected applications of Item Response Theory.* In R. L. Linn (ed.), *Educational Measurement.* New York: MacMillan (1989).
21. HERNÁNDEZ ,ORALLO, J. *Extracción automática de conocimiento de bases de datos e ingeniería de software.* Programación declarativa e ingeniería de la programación. (2000).
22. JIMENEZ, J. A. *Introducción a la Inteligencia Artificial,* Dpto. Ciencias de la Computación de Inteligencia Artificial, 2004. [2005].
Disponible en: <http://www.cs.us.es/cursos/ia-2004/temas/introduccion.pdf>
23. KATZ and LESGOLD. *Modelling the student in SHERLOCK II.*, 1994. 150 p.
24. LORD, F. M. & N. M. R. *Statistical theories of mental test scores.* Reading, MA: Addison- Wesley (1968).
25. LORD, F. M. *Applications of item response theory to practical testing problems.* Hillsdale, NJ: Lawrence Erlbaum Associates (1980).
26. MARQUÉS, P. *El software educativo,* 1996. [2005]. Disponible en:
http://www.filos.unam.mx/POSGRADO/seminarios/pag_robertp/paginas/soft_edu.html
27. MARTIN, J. and K. VANLEHN. *Student assessment using Bayesian nets,* 1995. [2005]. Disponible en: <http://www.pitt.edu/~vanlehn/distrib/journal/HCS95.pdf>
28. MILLÁN, E, M. T., PÉREZ DE LA CRUZ, JOSÉ LUIS Y CONEJO, RICARDO. *Modelado del alumno.* en. España, Málaga, 2004. 9.p.

29. MILLÁN, E. M. T., PÉREZ DE LA CRUZ, JOSÉ LUIS Y CONEJO, RICARDO. Uso de redes bayesianas en test adaptativos informatizados. en. España, Málaga, 2003. 9.p.
30. MILLÁN, E. *Sistema bayesiano para el modelado del alumno*. Málaga, Universidad de Málaga, 2000. 195. p.
31. MISLEVY, R. and D. GITOMER. *The Role of Probability-Based Inference in an Intelligent Tutoring System.*, 1996. [2005]. Disponible en: <http://polaris.lcc.uma.es/~eva/investigacion/SBMA.pdf>.
32. PÉREZ, V., OSANA EIRIZ, CORREDERAS, G., LÓPEZ, M. *Folleto del curso Informática educativa. Maestría en pedagogía profesional.*: Instituto Superior Pedagógico para la Educación Técnica y Profesional. La Habana., 1997. 28. p.
33. PORTO, E. R. *Las computadoras en la Educación*, Horizonte, 2001. [2005]. Disponible en: <http://www.horizonteweb.com/sitios.htm>
34. RASCH, G. *Probabilistic models for some intelligence and attainment test*. Copenhagen: Danish Institute for Educational Research (1960).
35. RÍOS, A., MILLÁN, E., TRELLA, M., PÉREZ-DE-LA-CRUZ, J. L., & Conejo, R. *Internet Based Evaluation System*. In *Proceedings of the 9th World Conference of Artificial Intelligence and Education AIED'99*. IOS Press(1999).
36. SALINAS, G. K. *Red Escolar de Informática Educativa*. *Revista Red Escolar*, 2000. vol 2. número 7. Disponible en: <http://redescolar.ilce.edu.mx/redescolar/Revista/07/index.html>
37. SCHMULLER, J. *Aprendiendo UML en 24 horas*. primera. México, Pearson Educación, 2000. 448 p. ISBN: 968-444-463-X.
38. SUCAR, L. E. *Razonamiento con incertidumbre*, ITESM, 2003. [2005]. Disponible en: <http://dns1.mor.itesm.mx/esucar/incertidumbre.html>

39. SHUTE, V. J. *Intelligent Tutoring Systems: Past, Present and Future*. In D. Jonassen (ed), *Handbook of Research on Educational Communications and Technology*. Scholastic Publications (1995).
40. THISSEN, D, & MISLEVY, R. *Testing Algorithms*. In H. Wainer (ed.), *Computerized Adaptive Testing: A Primer*. Hillsdale, NJ: Lawrence Erlbaum Associates Publishers (1990).
41. WAINER, H., & MISLEVY, R. *Item Response Theory, Item Calibration and Proficiency Estimation*. En H. Wainer (ed.), *Computerized Adaptive Testing: A Primer* Hillsdale, NJ: Lawrence Erlbaum Associates Publishers.1990. 65-102 p.

GLOSARIO DE TÉRMINOS.

- *Algoritmo*

Es un conjunto finito de instrucciones o pasos que sirven para ejecutar una tarea o resolver un problema. De un modo más formal, un algoritmo es una secuencia finita de operaciones realizables, no ambiguas, cuya ejecución da una solución de un problema.

- *Antepasado*

X es un *antepasado* de Z si y sólo si existe (al menos) un nodo Y tal que X es padre de Y e Y es antepasado de Z .

- *Arco*

Es la unión entre dos nodos y representa la dependencia entre dos variables del modelo. Un arco queda definido por un par ordenado de nodos (X, Y) .

Esta definición de arco corresponde a lo que en otros lugares se denomina *arco dirigido*. En la representación gráfica, un arco $(X; Y)$ viene dado por una flecha desde X hasta Y .

- *Cliques*

Subconjunto de nodos que es conjunto completo y máximo (no hay un conjunto completo que lo contenga).

- *Conjunto completo*

Subconjunto W de G que induce un subgrafo completo de G .

- *Descendiente*

Z es un *descendiente* de X si y sólo si X es un antepasado de Z .

- *Evidencia*

Es el conjunto de observaciones, $e = \{X = x, Y = y, \dots, Z = z\}$, en un momento dado.

- *Familia*

Es el conjunto formado por X y los padres de X , $pa(X)$.

- *Grafo completo*

Cada par de nodos distintos son adyacentes.

- *Grafo dirigido*

Es un par $G = (N;A)$ donde N es un conjunto de nodos y A un conjunto de arcos definidos sobre los nodos.

- *Hijo*

Y es un *hijo* de X si y sólo si existe un arco $(X; Y)$.

- *Independencia*

Dos variables X e Y son independientes, si la ocurrencia de una no tiene que ver con la ocurrencia de la otra.

- *Nodo*

Un nodo X es una variable aleatoria que puede tener varios estados x_i . La probabilidad de que el nodo X este en el estado x se denotará como $P(x) = P(X = x)$.

- *Nodo Terminal*

Es el nodo que no tiene hijos.

- *Observación*

Es la determinación del estado de un nodo ($x \text{ } X =$), a partir de un dato obtenido en el exterior del modelo.

- *Padre*

X es un *padre* de Y si y sólo si existe un arco $(X; Y)$. Los padres de X se representan como $pa(X)$. Por semejanza con el convenio utilizado para variables y sus valores, $pa(x)$ representará el vector formado al asignar un valor a cada nodo del conjunto $pa(X)$.

- *Probabilidad a posteriori*

Es la probabilidad de una variable condicionada a la existencia de una determinada evidencia; la probabilidad a posteriori de X cuando se dispone de la evidencia e se calcula como $P(X | e)$.

- *Probabilidad a priori*

Es la probabilidad de una variable en ausencia de evidencia.

- *Probabilidad condicional*

Dadas dos variables X e Y , la probabilidad de que ocurra j y dado que ocurrió el evento i x , es la probabilidad condicional de Y dado X , y se denota como $P(y_j | x_i)$.

- *Probabilidad conjunta*

Dado un conjunto de variables la probabilidad conjunta especifica la probabilidad de cada combinación posible de estados de cada variable.

- *Test*

Conjunto de preguntas para evaluar el conocimiento sobre un tema específico.