# Algoritmo para la evaluación de exámenes tipo test en sistemas e-learning avanzados

R. Barchino, J. M. Gutiérrez, J. Macías, S. Otón
Dpto. de Ciencias de la Computación
Universidad de Alcalá
28871 Madrid
e-mail: {roberto.barchino, josem.gutierrez, javier.macias, salvador.oton }@uah.es

**Resumen**

El objeto de este documento es presentar ideas sobre la corrección automática de tests generados por ordenador en el contexto de sistemas de e-learning. Más en detalle, se presentan trabajos realizados entorno a las normas del IMS (Instructional Management Systems) en su apartado QTI (Question & Test Interoperability). Estos trabajos, orientados a la implementación de las normas en un sistema real y su uso para obtener una medida de sus posibilidades, han desembocado en la necesidad de tener en cuenta escenarios más complejos que los planteados por IMS y la necesidad de desarrollar algoritmos adaptados a estos escenarios.

El panorama final estudiado, comprende la realización de pruebas en varias fases donde las preguntas tendrán asociados un nivel y sólo podrán ser seleccionados para una fase de su nivel o superior. Un estudiante accederá a una pregunta de mayor nivel sólo si se ha superado en cierto grado la fase anterior.

También se trata, brevemente, los ajustes realizados en las diversas fases de funcionamiento del sistema para dotar al escenario de un funcionamiento lo más similar posible al que tendrá si no fuese ejecutado por un sistema automático. En particular, se describe la forma de calificar con una cierta relevancia a las preguntas que forman la base de conocimiento y cómo calcular la relevancia de forma sencilla, correcta y representativa del nivel real que representa esa pregunta.

## 1. Motivación

En estos últimos años se han desarrollado en el ámbito universitario y empresarial distintos sistemas de e-learning, que se centran en crear entornos virtuales de aprendizaje mediante una serie de herramientas tecnológicas que originalmente no estaban pensadas para este fin pero se han integrado correctamente en estos sistemas. Un componente fundamental de estos sistemas es el encargado de la evaluación de los avances de los alumnos en la asimilación de contenidos docentes.

Existen distintos estándares internacionales que proponen algoritmos de corrección de exámenes tipo test [1], pero no existe en la actualidad ningún sistema que permita la corrección automática de otro tipo de preguntas más abiertas que las de tipo test, debido a los lógicos problemas de análisis del lenguaje natural.

En el Departamento de Ciencias de la Computación de la Universidad de Alcalá hemos trabajado en estos últimos años con distintos sistemas comerciales de e-learning como LUVIT [2], WebCT [3] y Learning Space [4], pero aun cuando son suficientemente flexibles para trabajar con ellos decidimos crear nuestro propio sistema para experimentar e investigar de una forma más profunda.

En el año 2001 desarrollamos un primer sistema de e-learning para la generación aleatoria y corrección de exámenes tipo test [5], al año siguiente se creó una nueva versión del sistema que cumplía el estándar IMS-QTI para el almacenamiento e intercambio de test y en la actualidad se ha desarrollado una tercera versión del sistema que incorpora los distintos algoritmos

de evaluación de exámenes del estándar QTI de IMS.

En esta última versión de nuestro sistema hemos desarrollado nuevos algoritmos de evaluación que extienden a los del estándar QTI y que siguiendo su especificación almacenaremos en XML integrados en la propia estructura QTI. El motivo de desarrollar estos nuevos algoritmos es la búsqueda de mayor precisión en la evaluación de los conocimientos de los alumnos.

Una vez centrados en la evaluación de contenidos debemos tener presente una serie de cuestiones que nos van a servir como punto de partida en el desarrollo de los nuevos algoritmos de evaluación del aprendizaje.

Estas cuestiones son, en primer lugar que los exámenes deberán abarcar los contenidos fundamentales de la asignatura, que las preguntas de todos los exámenes sean claras y no exista ambigüedad, también que se debe dar a conocer a los usuarios del sistema el criterio o algoritmo de evaluación, y por último la calidad del propio examen tipo test [6] [7]

## 2. Algoritmo Propuesto

Los algoritmos disponibles en IMS [1] se han desarrollado teniendo en cuenta una serie de casos de uso significativos con la finalidad de que sean suficientemente flexibles como para cubrir otros muchos casos.

Algunos de estos casos de uso se enumeran en la tabla 1. No se enumeran todos los casos de uso que propone el estándar, debido fundamentalmente a que algunos de ellos no son de aplicación a nuestro sistema, por ser de propósito general como Inglés, Biología o materias orientadas a la impartición del método científico.

Nuestro sistema en su versión actual permite la realización de test como los incluidos en los tres primeros casos de uso y prepara, aunque no implementa aún, el cuarto tipo, esto es, permite almacenar pesos para preguntas pero no utilizarlos en la corrección. Todos estos casos de uso están pensados para la realización de la prueba de evaluación una vez terminado el aprendizaje y que se presenten los resultados de la misma de forma inmediata.

| Caso de Uso | Caracterísitcas |
|---|---|
| Test de elección múltiple | Múltiples opciones<br>Una sola correcta<br>Todas las preguntas valen lo mismo |
| Test verdadero/falso | Dos opciones (verdadero/falso)<br>Todas las preguntas valen lo mismo |
| Test de respuesta múltiple | Múltiples opciones<br>Más de una correcta<br>Todas las preguntas valen lo mismo |
| Test fin de temario | Múltiples opciones<br>Una sola correcta<br>Cada pregunta tiene diferente valor |

Tabla 1. Casos de uso del estándar QTI

Los algoritmos de evaluación contenidos en la especificación QTI [1] se enumeran en la tabla 2.

| Algoritmo |
|---|
| Number Correct |
| Number Correct (Attempted) |
| Weighted Number Correct |
| Weighted Number Correct (Attempted) |
| Parameter Weighted Number Correct |
| Parameter Weighted Number Correct (Attempted) |
| Sum Of Scores |
| Sum Of Scores (Attempted) |
| Weighted Sum Of Scores |
| Weighted Sum Of Scores (Attempted) |
| Parameter Weighted Sum Of Scores |
| Parameter Weighted Sum Of Scores (Attempted) |
| Best K from N |
| Guessing Penalty |
| Weighted Guessing Penalty |

Tabla 2. Algoritmos del estándar QTI

En nuestro caso, queremos plantear un caso de uso diferente, de mayor flexibilidad y complejidad que puede ser usado para la evaluación final o para auto evaluaciones periódicas por parte del alumno. Este caso de uso planteará preguntas de

distinto peso en función del grado de acierto anterior del alumno y la corrección del examen deberá tener en cuenta esta situación. Por este motivo, los algoritmos existentes en QTI no serán suficientes.

La utilización de pesos para valorar las preguntas será una herramienta fundamental a la hora de permitir a los educadores obtener del sistema los exámenes de calidad buscados. En los siguientes apartados analizaremos la generación y significado de estos pesos y el caso de uso y algoritmo planteados.

### 2.1. Preguntas con pesos

El peso asociado a una pregunta va a representar la relevancia de esa pregunta y del acierto o fallo en su respuesta. Por este motivo, es muy importante y difícil elegir correctamente este valor. La única fuente de información para obtener este valor a priori es la experiencia y conocimientos del docente que genera la base de preguntas a partir de la que se generan los tests.

Una forma de simplificar el establecimiento de la relevancia de una pregunta es la de atender a varios aspectos sobre la misma de forma independiente para luego reunir todos los valores en uno mediante un operador de agregación.

En nuestro caso, hemos considerado dos características de relevancia de cada cuestión, la importancia y la dificultad. Para cada una de ellas hemos establecido un rango de valores enteros de 1 a 5 que luego se combinan dando una matriz que se puede reducir por varios operadores de agregación. En la Figura 1A podemos observar la matriz de posibles valores, en la Figura 1B el resultado del operador de agregación media aritmética.

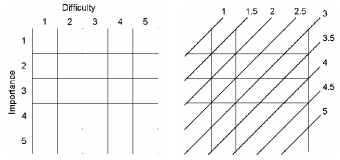


Figura 1. Relevancia de las Preguntas

En la actualidad, utilizamos el operador media aritmética como operador de agregación porque posee muy buenas características para nuestros fines no siendo la menor de ellas la facilidad de implementación.

### 2.2. Algoritmo de Evaluación

A continuación vamos a describir en detalle el caso de uso sobre el que hemos trabajado, así como ejemplos del mismo que nos den pie a presentar el algoritmo de evaluación utilizado.

El objetivo es que el sistema pueda realizar exámenes tipo test y evaluar a un alumno sin intervención del profesor. Pero, ¿qué le interesa al profesor?, que el alumno conteste de forma correcta a las preguntas de una determinada materia en varias fases con preguntas de relevancia creciente para dilucidar así su verdadero conocimiento en la materia no sólo en cantidad sino también en nivel del mismo. Para conseguir estos objetivos, el examen tipo test se realizará dividido en un número de fases concreto, en nuestro caso ese número será de cinco. Los exámenes, por tanto, estarán divididos en cinco fases con cuatro preguntas por fase. El funcionamiento será el siguiente: en la primera fase se harán preguntas de nivel 1, si el alumno responde correctamente una pregunta, en la siguiente fase se la harán preguntas de nivel 2, es decir, un nivel superior a la pregunta realizada con anterioridad y si el alumno no contesta o contesta incorrectamente una pregunta, en el nivel siguiente se le volverá a realizar una pregunta del mismo nivel de la pregunta que falló o no contestó. En cada nueva fase, el número de preguntas que avanzan de nivel será igual al número de preguntas que contestó correctamente y el número de preguntas que serán del mismo nivel será igual al número de preguntas que no superó. Esto puede dar lugar a una fase que incluya preguntas de cuatro niveles distintos en el caso de un alumno de conocimientos dispersos y de profundidad irregular en las distintas áreas de la materia.

### 2.3. Realizando un Test

El alumno que realice un test recibirá una nota numérica, con valor entre 0 y 10, que el sistema generará mediante una serie de operaciones aritméticas. Este valor será generado por el algoritmo de evaluación que

hemos desarrollado. Antes de presentar el algoritmo, vamos a introducir una serie de ejemplos de posibles situaciones que nos han servido, entre otros, para crear el algoritmo. El algoritmo, estará compuesto por varias fases. En una primera fase, se calcula un valor para el conjunto de respuestas de un alumno en el que se utiliza el peso asignado a cada una de las preguntas presentadas en la prueba. La fórmula para obtener estos valores se presenta en la figura 2.

$$\sum_{i}^{i} \text{item}_i \left( \frac{\text{relevancia}_i + 1}{2} \right)$$

Figura 2. Fórmula del Algoritmo de Evaluación

Como se puede apreciar en la fórmula, los pesos cuyo valor estaban entre 1 y 5, se modifican para que queden entre 1 y 3, suavizando el peso de los elementos de mayor nivel. La asignación de pesos se puede ver en la tabla 3.

| Importancia | Peso |
|---|---|
| 1 | 1 |
| 2 | 1.5 |
| 3 | 2 |
| 4 | 2.5 |
| 5 | 3 |

Tabla 3. Normalización Pesos/Importancia

Empezaremos por presentar una serie de ejemplos de situaciones que se pueden producir en el sistema tras la realización de un test por parte de un alumno. En primer lugar presentamos las situaciones más sencillas que marcan los tres límites principales en toda evaluación, el máximo, el mínimo y el valor frontera entre superado y no superado.

| | | | | |
|---|---|---|---|---|
| 1 ✗ | 1 ✗ | 1 ✗ | 1 ✗ | 1 ✗ |
| 1 ✗ | 1 ✗ | 1 ✗ | 1 ✗ | 1 ✗ |
| 1 ✗ | 1 ✗ | 1 ✗ | 1 ✗ | 1 ✗ |
| 1 ✗ | 1 ✗ | 1 ✗ | 1 ✗ | 1 ✗ |

Figura 3. Resultado con 0 puntos.

| | | | | |
|---|---|---|---|---|
| 1 ✓ | 2 ✓ | 3 ✓ | 4 ✗ | 4 ✗ |
| 1 ✓ | 2 ✓ | 3 ✓ | 4 ✗ | 4 ✗ |
| 1 ✓ | 2 ✓ | 3 ✓ | 4 ✗ | 4 ✗ |
| 1 ✓ | 2 ✓ | 3 ✓ | 4 ✗ | 4 ✗ |

Figura 4. Resultado con 18 puntos.

| | | | | |
|---|---|---|---|---|
| 1 ✓ | 2 ✓ | 3 ✓ | 4 ✓ | 5 ✓ |
| 1 ✓ | 2 ✓ | 3 ✓ | 4 ✓ | 5 ✓ |
| 1 ✓ | 2 ✓ | 3 ✓ | 4 ✓ | 5 ✓ |
| 1 ✓ | 2 ✓ | 3 ✓ | 4 ✓ | 5 ✓ |

Figura 5. Resultado con 40 puntos.

Los ejemplos de la figuras 3, 4 y 5 muestran los resultados de 0 puntos (mínimo), 18 puntos (valor frontera) y 40 puntos (máximo) del algoritmo para un determinado test.

A continuación, se presentan una serie de ejemplos que ilustran situaciones potencialmente conflictivas, por su alineamiento con el valor frontera ya sea por exceso o por defecto.

| | | | | | |
|---|---|---|---|---|---|
| 1 ✓ | 2 ✓ | 3 ✗ | 3 ✗ | 3 ✗ | |
| 1 ✓ | 2 ✓ | 3 ✗ | 3 ✗ | 3 ✗ | |
| 1 ✓ | 2 ✓ | 3 ✗ | 3 ✗ | 3 ✗ | |
| 1 ✓ | 2 ✓ | 3 ✗ | 3 ✗ | 3 ✗ | |

Figura 6. Resultado con 10 puntos.

| | | | | |
|---|---|---|---|---|
| 1 ✗ | 1 ✗ | 1 ✓ | 2 ✓ | 3 ✓ |
| 1 ✗ | 1 ✗ | 1 ✓ | 2 ✓ | 3 ✓ |
| 1 ✗ | 1 ✗ | 1 ✓ | 2 ✓ | 3 ✓ |
| 1 ✗ | 1 ✗ | 1 ✓ | 2 ✓ | 3 ✓ |

Figura 7. Resultado con 18 puntos.

| | | | | |
|---|---|---|---|---|
| 1 ✓ | 2 ✗ | 2 ✓ | 3 ✓ | 4 ✗ |
| 1 ✓ | 2 ✓ | 3 ✗ | 3 ✓ | 4 ✗ |
| 1 ✓ | 2 ✓ | 3 ✓ | 4 ✗ | 4 ✓ |
| 1 ✗ | 1 ✓ | 2 ✓ | 3 ✗ | 3 ✓ |

Figura 8. Resultado con 20,5puntos.

En los ejemplos de la figuras 6, 7 y 8 se muestran los resultados numéricos de 10 puntos, 18 puntos y 20,5 puntos del algoritmo para un determinado test. No se incluyen otras situaciones no cercanas a la frontera porque su evaluación no ha aportado datos significativos en el análisis que permitió crear el algoritmo.

### 2.4. Evaluación de un test.

En algunos de los ejemplos nos hemos encontrado con un problema a la hora de dar la nota final al alumno, en la Figura 7, la puntuación obtenida es de 18 puntos, en teoría, una puntuación final de cinco. Pero vemos, que aún cuando el alumno ha contestado los tres niveles, éste ha necesitado más preguntas del nivel mínimo, fallando hasta 8 de estas, en este tipo de casos se desea penalizar la puntuación del ejercicio porque se aprecia que el alumno no posee conocimientos de base suficientes para superar la prueba. Esta penalización se realizará mediante la siguiente norma: si el alumno comete 8 o más fallos en los niveles 1,2 y 3 e independientemente de la nota que la fórmula matemática le otorgue, esa prueba será calificada como no superada.

Con la penalización incorporada al algoritmo, se ha conseguido plasmar dos ideas subyacentes en la mente del docente cuando corrige un test, la puntuación depende del número de respuestas correctas, pero marcadas con un cierto peso, y la necesidad de superar un mínimo del conocimiento de base en la materia, también representado por los pesos.

Una vez obtenida la valoración intermedia de una prueba, se procederá a su normalización para obtener la nota final entre 0 y 10. Esta normalización se realizará mediante el ajuste de los dos tramos de pendiente desigual de valores intermedios. Los dos tramos y el ajuste a realizar se presentan en las tablas 4 y 5.

| Evaluación intermedia | Evaluación final |
|---|---|
| 0 | 0 |
| 18 | 5 |

Tabla 4. Normalización, primer rango.

| Evaluación intermedia | Evaluación final |
|---|---|
| 18 | 5 |
| 40 | 10 |

Tabla 4. Normalización, segundo rango.

## 3. Conclusión

Las iniciativas de establecimiento de normativas de desarrollo de los distintos componentes de sistemas e-learning son muy importantes y el trabajo en los límites de las mismas más aún porque establece solidez a lo ya desarrollado. El escenario planteado y el algoritmo de evaluación aplicado, trabajan en esta zona del conocimiento para abrir nuevas vías de desarrollo a la normativa existente. Los resultados obtenidos hasta la fecha y las ideas presentadas aparecen claramente como uno de los caminos a seguir por las nuevas versiones de la normativa y por nuestros propios sistemas.

## 4. Agradecimientos.

Nos gustaría dar las gracias a nuestros compañeros del departamento de Ciencias de la Computación por la ayuda recibida para el desarrollo de este trabajo, especialmente al grupo de trabajo de e-learning y al de operadores de agregación.

## Referencias

[1] IMS Global Learning Consortium. *IMS Question & Test Interoperability,* 2002
[2] LUVIT, 2002. *http://www.luvit.com*
[3] WebCT, 2002. *http://www.webct.com*
[4] Learning Space, 2002 *http://www.lotus.com/learningspace*
[5] Barchino, R et al. *EDVI: Un sistema de apoyo a la enseñanza presencial basado en Internet*. VII Jornadas de Enseñanza Universitaria de la Informática. Mallorca, España, 2001
[6] Aiken, L.R. *Test psicologógicos y evolución*. Prentice Hall Hispanoamericana, México, 1996
[7] Croket, L. and Algina J. *Introduction to Classical and Modern Test Theory.* Holt, Rinehart and Winston, 1986