

● Alberto Corbí y Daniel Burgos  
Logroño (La Rioja, España)

# LIME: un modelo de recomendación para entornos de aprendizaje online formal/informal

LIME: a model of recommendation for online learning environments  
formal/informal

## RESUMEN

En los modelos e implementaciones sobre eLearning (conocidos habitualmente como sistemas Gestores de Aprendizaje o LMS) se da una aparente ausencia de conexión entre las actividades de índole formal e informal. Además, la metodología online se focaliza en el establecimiento de un set de unidades y objetos de aprendizaje, así como tests y recursos como foros de discusión, blogs personales y mensajería. Ignoran, por tanto, todo el potencial del aprendizaje que surge de la interrelación entre el LMS, redes sociales y otras fuentes externas. Gracias a este comportamiento, a la interacción del usuario y a la labor de seguimiento y consejo personalizado por parte de un tutor, puede mejorar esta experiencia de aprendizaje. Se ha diseñado y desarrollado un modelo de aprendizaje online adaptativo para redes sociales de ámbito restringido, que da relevancia a este enfoque. Además, se ha programado un módulo de software que implementa este modelo conceptual de manera práctica y empleando para ello estándares promulgados por el IMS Global y tecnologías web. Finalmente se presenta el despliegue técnico de este producto entorno a un sistema gestor de contenidos académicos real.

## ABSTRACT

In current eLearning models and implementations (e.g. Learning Management Systems-LMS) there is a lack of engagement between formal and informal activities. Furthermore, the online methodology focuses on a standard set of units of learning and learning objects, along with pre-defined tests, and collateral resources like, i.e. discussion for a and message wall. They miss the huge potential of learning via the interlacement of social networks, LMS and external sources. Thanks to user behavior, user interaction, and personalized counseling by a tutor, learning performance can be improved. We design and develop an adaptation eLearning model for restricted social networks, which supports this approach. In addition, we build a practical eLearning software module, based on standards from IMS Global and web technologies, that implements this conceptual model in a real application case. We present a preliminary deployment status on a modern learning management system..

## PALABRAS CLAVE / KEYWORDS

Aprendizaje potenciado mediante la tecnología, Personalización en aprendizaje a distancia, Redes sociales, Modelo Educativo Conceptual, eLearning, LMS.

Technology-enhanced Learning, eLearning, Personalization, Social Networks, Conceptual Educational Model, LMS.

## SOBRE EL AUTOR/ES

Dr. Alberto Corbí. Universidad Internacional de La Rioja (UNIR) (alberto.corbi@unir.net).

Dr. Daniel Burgos. Universidad Internacional de La Rioja (UNIR) (daniel.burgos@unir.net).

## 1. Introducción

Las redes sociales centradas en un tema o en una comunidad específicos constituyen un vehículo preciso de comunicación y conexión entre usuarios, tengan el rol que tengan. Estos pueden ser alumnos, profesores, empleados, personal de administración, directores académicos, o responsables de finanzas, que muestran una determinada actitud según el contexto y el objetivo de su aportación o estado. Cada usuario puede preguntar, responder, iniciar una actividad, seguir otra, comentar el trabajo de un compañero, evaluar el trabajo de otro usuario, explorar en Internet, realizar un test programado, participar en una video-conferencia con el profesor de un curso, y así una larga lista de actividades. Y en todas ellas el usuario puede ser pro-activo, reactivo, pasivo, consumidor, productor, distribuidor, y mostrar muchas otras actitudes.

Diversas teorías sostienen que el estudiante no es completamente consciente de lo que aprenden. Incluso cuando el estudiante realiza un control de su propio progreso, puede percibir nociones equivocadas acerca de las competencias adquiridas [1], lo que conlleva una valoración errónea de los progresos reales de aprendizaje y los resultados asociados [2, 3]. Por lo tanto, los estudiantes requieren una guía que permita mejorar su rendimiento, por ejemplo a través del uso de experiencia meta-cognitiva [4]. Otros enfoques, sin embargo, utilizan la recomendación al usuario para que este decida sobre su itinerario en función de una tutorización. Este sistema se utiliza en entornos comerciales como venta de libros o discos [5, 6]. En este campo, existen diversos sistemas recomendadores que permiten mejorar la experiencia de aprendizaje [7-10]. En este sentido, la recomendación por sí misma no garantiza nada, sino la interpretación que de ella haga el usuario. Esta recomendación debe ser función de entradas (inputs) que no son otra cosa sino evidencias, momentos de interacción, preferencias, comportamientos, perfiles, y otros sucesos similares. Estos inputs son fruto de la monitorización del usuario [11, 12].

El presente trabajo ahonda en este ecosistema de aprendizaje, marcado por la acotación de la muestra y de la temática, y por la definición e implementación de un modelo de aprendizaje personalizado que tiene en cuenta todos los factores y particularidades descritos arriba, para proporcionar tutorización adaptativa gracias a la utilización de reglas de interacción. Es decir, aprendizaje en una red social temática acotada, donde los usuarios interactúan con el sistema y entre sí y reciben guía académica y recomendaciones de manera individualizada.

## 2. Descripción del modelo LIME.

El modelo desarrollado, L.I.M.E., se basa en cuatro categorías: Learning, Interaction, Mentoring y Evaluation, que son analizados y utilizados como elementos de control y guía en el entorno de aprendizaje. Estas categorías se comportan como vectores que definen una serie de factores que permiten proporcionar una recomendación de itinerario personalizada para cada usuario [13]:

- **L=Learning.** Contribución específica de cada estudiante
- **I=Interaction.** Contribución de cada estudiante, en relación con otros
- **M=Mentoring.** Contribución del profesor/tutor/experto
- **E=Evaluation.** Evaluación aplicada a las tres categorías anteriores

A través de estas cuatro categorías, estudiamos el comportamiento y rendimiento de un estudiante en función también de su contexto, incluyendo reputación, confianza y otros factores. El modelo también define una estructura de conocimiento distribuida entre LE (Learning Environment), LO (Learning Object), UK (Unit of Knowledge), y PLN (Personal Learning Network).

El diseñador instruccional y/o el profesor deben definir sobre esta estructura una estrategia de aprendizaje, que tendrá en cuenta diversos elementos:

- **Setting.** Se corresponde con el escenario formal o informal donde se tienen lugar las acciones (entradas,

inputs) que se tratan a continuación y donde se implementa la metodología educativa.

- **Category.** Son las categorías antes mencionadas (i.e. Learning, Interaction, Mentoring, Evaluation). Cada una de ellas requiere un peso porcentual que definirá la importancia concreta para un escenario de aprendizaje. La categoría de Evaluación se define y aplica de manera transversal.

- **Input.** Factores específicos (i.e. acciones del estudiante, comportamientos, calificaciones, etc.) que alimentan las categorías y el modelo de aprendizaje (entradas), también con peso relativo asignado.

La siguiente tabla (Tabla 1) muestra una distribución ficticia de Settings, Categories e Inputs, todos ellos con pesos asociados, así como la configuración del modelo para una unidad de aprendizaje (e.g. clase, curso). Nótese que la categoría Evaluation se encuentra resaltada de manera transversal a lo largo de las otras tres.

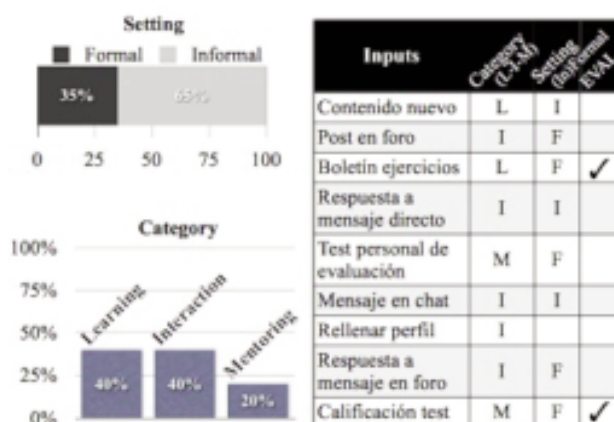


Tabla 1. Ejemplo de definición del modelo LIME.

### 3. LIME como un servicio independiente.

La **implementación práctica** en producto de software del modelo LIME recibe el nombre de **iLIME**. Uno de los principales objetivos marcados con respecto a la arquitectura de diseño de iLIME ha consistido en la independencia de plataforma. Es decir, iLIME es capaz de interactuar con la gran mayoría de sistemas gestores de contenidos académicos (LMS) disponibles tanto en escenarios comerciales como libres (FOSS). Para alcanzar este hito, iLIME ha sido diseñado como una herramienta LTI, tecnología que se detalla a continuación.

La **Interoperabilidad entre Herramientas de Aprendizaje** (Learning Tool Interoperability - LTI) forma parte del cuerpo de estándares consorcio IMS Global. Uno de los beneficios clave de LTI es que propone un protocolo de mensajería independiente de plataforma que permite la adición de herramientas y servicios externos a cualquier LMS que implemente el mencionado estándar [14, 15]. La especificación LTI define dos tipos de integración: completa y básica. En la implementación de iLIME ha sido suficiente con esta última y por otro lado ha ayudado a asegurar una compatibilidad todavía mayor con los LMS disponibles en el mercado. LTI designa a su vez dos tipos de actores principales: **consumidores** de herramientas (tool consumer), normalmente representado por el propio LMS (Sakai, Moodle, Blackboard, etc.) y los **proveedores** de herramientas (tool provider) que consisten en todos aquellos servicios externos que proveen de contenidos o experiencias educativas complementarias al sistema consumidor.

En la implementación prototipo de iLIME que se expone aquí, se ha hecho uso del LMS Apereo Sakai CLE (en su versión 2.9.3) como software consumidor. Sin embargo, dado el gran apoyo que está recibiendo la tecnología LTI, iLIME es fácilmente transplantable a cualquier otro entorno académico desplegado de manera habitual en y por instituciones/entidades educativas.

La gran mayoría de versiones modernas de sistemas LMS como ATutor, Moodle, Sakai, Blackboard,

LAMS, Instructure Canvas, etc. son ya capaces de operar como consumidores LTI. De igual manera, muchos servicios, empresas y proyectos de escala mundial (Wikipedia, YouTube, Wordpress, etc.) así como otros más específicos y humildes (como por ejemplo Noteflight) han adaptado sus interfaces para funcionar como proveedores de contenidos LTI (Fig. 1) y conectarse transparentemente a cualquiera de los LMS citados anteriormente.



Fig 1. Algunos servicios de internet conocidos que son capaces de funcionar como proveedores LTI.

El protocolo LTI Básico, permite a las herramientas proveedoras, entre otras acciones:

- **Autenticación** mediante tokens con el consumidor
- Presentación de contenidos externos de manera **transparente** en la interfaz del consumidor (LMS).
- Acceder a la **lista** de alumnos de un sitio.
- Remitir **calificaciones** obtenidas por el alumno para su inclusión en el boletín de notas.
- **Registrar información** propia en el espacio de datos del LMS.

Más explícitamente, los aspectos del LTI básico que iLIME explota con mayor relevancia son:

#### A. *Lienzo integrado (canvas) en LMS*

Consiste en un elemento HTML5 de tipo iframe, pop-up o similar que ofrece la posibilidad de **presentar información** y datos en la misma ventana del LMS (en una área específica o en una ventana emergente), potenciando la sensación de **inmersión** por parte de la herramienta proveedora en el la ventana del LMS(consumidor).

#### B. *Registro unificado en el LMS*

Todos los datos de la herramienta proveedora pueden salvaguardarse de manera segura en el almacén de datos del LMS (mediante el protocolo Settings definido en LTI 1.1), permitiendo a la primera el poder **funcionar sin un almacén** de datos dedicado o base de datos ad-hoc.

#### C. *Autenticación moderna*

El sistema de autenticación entre iLIME y el LMS está basado en el robusto estándar de facto OAuth 2.0 (RFC 6749) implementado por las conocidas empresas y servicios de Internet como Google y Twitter. Todas las comunicaciones y mecanismos de persistencia del estado de la sesión entre proveedor y consumidor tienen lugar en un entorno seguro protegido por pares clave-valor negociados (Fig. 2).

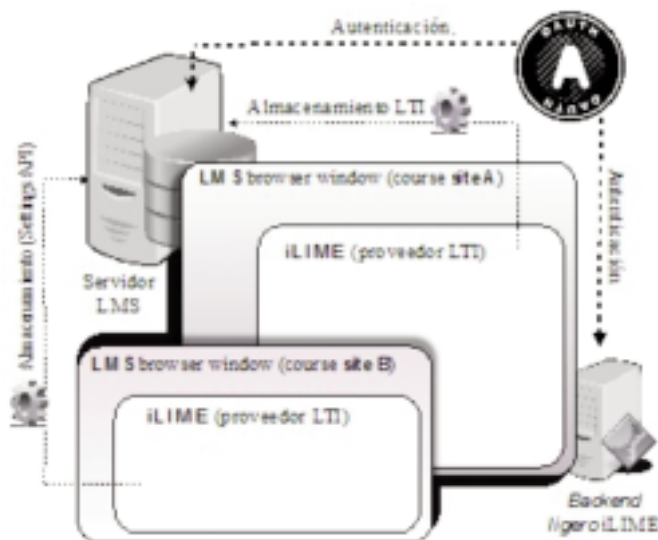


Fig. 2. Principales tecnologías web e IMS utilizadas.

#### 4. Consola de administración del modelo.

La consola de administración de iLIME es el componente destinado a la configuración del modelo. Cada curso o sitio cuenta con su consola y parámetros aislados e independientes del resto. Básicamente, esta configuración consiste en los siguientes apartados:

##### A. *Parámetros fundamentales del modelo.*

En esta sección es donde se da el peso específico a cada una de las categorías (aprendizaje, interacción y tutoría) y a los escenarios formal e informal. El establecimiento de estos parámetros se realiza mediante una interfaz de usuario sencilla. Esta configuración es específica a cada sitio (clase, curso, etc.) dentro del LMS.

##### B. *Sección de Inputs*

Aquí es donde pueden añadirse aquellas actividades realizadas por los estudiantes que servirán para alimentar el modelo (entradas), así como configurar su peso específico, su categoría, su escenario (setting) y si son evaluables (forman parte de las calificaciones) o no.

##### C. *Editor de reglas.*

En este apartado, el profesor, tutor o gestor puede crear y editar reglas que tomarán a su vez los inputs y la configuración general del modelo como parámetros. El resultado de la interpretación de estas reglas será la recomendación al alumno.

La consola de configuración ha sido diseñada de tal manera que siempre que acontezca cualquier modificación (en los parámetros del modelo, inputs, reglas, etc.) será registrada automáticamente en el LMS, gracias a la API de configuración del LTI Básico que se presenta a continuación. El framework HTML5 escogido para el diseño de iLIME permite realizar operaciones de deshacer/rehacer con comodidad.

La configuración de cada curso reside, gracias a la API de configuración (Settings API) de LTI en un área discreta y accesible de la base de datos del LMS. iLIME puede consultar con permisos de lectura y escritura siempre que sea necesario. Cada sitio (curso, clase, etc.) dentro del LMS cuenta con su propio compartimento para la API de configuración. Aunque internamente, los LMS suelen codificar esta área de información con formato simple VARCHAR, se ha decidido codificar la configuración de iLIME (para cada curso) como una cadena de texto JSON, traducida a su vez en una secuencia de caracteres en base 64. El objetivo de esta última traducción no es otro que el de asegurarnos el registro adecuado de caracteres no ASCII que puedan pre-

sentar un problema con ciertas bases de datos e implementaciones LTI algo desfasadas. La próxima revisión de LTI (LTI 2.0) añade soporte completo para JSON, Unicode y tamaño de configuración generosamente extendido. La versión actual del LTI Básico (1.1) sólo permite salvaguardar 4K de información. Para contrarrestar esta limitación y que iLIME pueda persistir los inputs, configuración y reglas de cada sitio, se aplica un algoritmo de condensación JSON (cuyo resultado es, a su vez, una cadena JSON válida) seguido de la habitual compresión LZMA. Finalmente tiene lugar la traducción a base 64. Esta última cadena es la enviada al LMS para su escritura. La lectura de la misma implica el proceso inverso.

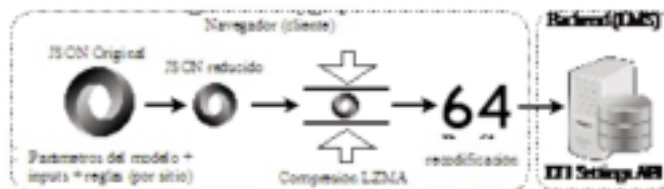


Fig. 3 Fases de compresión de la información antes de su registro en el almacén de configuración LTI (Settings API).

## 5. Generación y presentación de recomendaciones.

El proceso de generación de recomendaciones conlleva dos pasos consecutivos y diferenciados. Por un lado es necesario llevar a cabo una tarea de minería para obtener **valores medios** (o máximos) de inputs y por otro han de aplicarse **reglas** que tengan en cuenta estos estadísticos y los valores concretos de cada usuario/alumno que solicita una recomendación.

### A. Monitoreo de valores estadísticos sobre sitios e inputs.

Uno de los aspectos fundamentales de la implementación del modelo LIME es la necesidad de compilar valores estadísticos sobre cada input. Más tarde estos valores intervendrán como parámetros en las reglas diseñadas por el tutor/profesor. En una primera iteración, se han tomado en cuenta valores medios y máximos por clase y por input. La comparación con los **valores máximos y medios de la clase/curso** son ciertamente buenas referencias que los docentes suelen usar a la hora de determinar si un alumno está realizando progresos. Estos referentes estadísticos pueden obtenerse de varias maneras. Resaltaremos dos tecnologías que quedan resumidas de manera esquemática en la fig. 5.

La primera y más inmediata manera es sin duda mediante el uso de **servicios web específicos** de cada LMS. Casi todos los LMS modernos disponen de interfaces simples, elegantes, y estándares de acceso a su información interna [19], siempre y cuando, claro está, una autenticación exitosa haya tenido lugar previamente. Estos protocolos de acceso suelen ser RPC, SOAP, REST, etc. Para el caso de iLIME se ha programado un conjunto de pequeños scripts, como prueba de concepto, para el LMS Sakai CLE que cumplen precisamente esta función. Estos scripts son ejecutados de forma asíncrona en momentos específicos de relativa presión computacional con el objetivo de importunar lo menos posible al sistema de e-learning. De hecho, la carga provocada por iLIME suele ser normalmente despreciable. Los resultados de estos pequeños comandos son parejas clave-valor que consisten en el nombre del input y el dato máximo (y/o medio) para ese input. Estos resultados son guardados (o actualizados) en el compartimiento del *Settings API* para poder ser consultado y tenido en cuenta ulteriormente.

La segunda aproximación a esta tarea es mediante el uso de la tecnología **TinCan**. Este proyecto va ganando cada vez más y más apoyos y captando la atención de la comunidad de Learning Analytics. En pocas palabras, TinCan consiste en un serie de herramientas, servicios y protocolos para automatizar **el registro de la actividad de los estudiantes** a lo largo de su recorrido en el sistema académico. Los registros obtenidos son cronológica indexados en el llamado **Almacén de Registros de Aprendizaje** (Learning Record Store), servicio que normalmente correrá en el mismo sistema o infraestructura informática que da vida al LMS. A este LRS

podrán acceder otras herramientas de *data mining*, *reporting* o *analytics* con el objetivo de realizar tareas de procesamiento, análisis y/o presentación de datos y resultados sobre la evolución académica de los alumnos y grupos de alumnos.



Fig. 4. Comunicación entre LMS y LRS (TinCan) para almacenamiento de registros de aprendizaje. Posterior uso por parte de, por ejemplo, herramienta de generación de informes.

A pesar de que aún queda trabajo por hacer, muchos sistemas de elearning están incorporando ya esta funcionalidad. TinCan representa una **enfoque atractivo y estandarizado para capturar eventos** y acciones de los usuarios e inventariarlos en un repositorio central de manera ordenada.

En nuestra primera iteración se ha optado por la primera opción, creándose por tanto un conjunto de scripts encargados de solicitar mediante tecnologías web al LMS Sakai, los valores máximos y medios de cada input.



Fig. 5: Inputs y monitorización de la actividad de los estudiantes mediante APIs dependientes de LMS (izquierda) y mediante el LRS de TinCan (derecha).

#### B. *Generación de recomendaciones y presentación a los estudiantes.*

iLIME no requiere el uso de procesos en segundo plano a la hora de generar las recomendaciones para un alumno específico. Todo ocurre en tiempo real (cuando el alumno accede a la herramienta) dentro de la ventana del navegador (Fig. 6).

Para conseguir esto, se han utilizado tecnologías de filtrado de colecciones [16, 17, 18] que son ejecutadas cuando un estudiante accede a iLIME desde dentro del propio entorno de aprendizaje. Estos filtros (reglas) se aplican sobre la colección de inputs mediante tecnologías HTML5 en el propio navegador que no interrumpen en absoluto al LMS ni al propio backend de iLIME. Por esta razón iLIME entra dentro del conjunto de paquetes de software de servidor conocidos como ligeros, dado los pocos recursos computacionales que consume y los esporádicos accesos a base de datos que contempla.

Si la operación de filtrado no devuelve ningún input (conjunto vacío  $\emptyset$ ), esto significa que no se cumplen las condiciones y por tanto no se emite la recomendación que acompaña a cada regla. Si por el contrario algunos de los inputs sobreviven a este proceso de tamizado, implica necesariamente que la regla programada se cumple y se procede a presentar la recomendación. Como ya se ha comentado, las operaciones de filtrado y recomendación (si procede) tienen lugar siempre que un alumno acceda al sistema y a la sección de recomen-

dación iLIME.

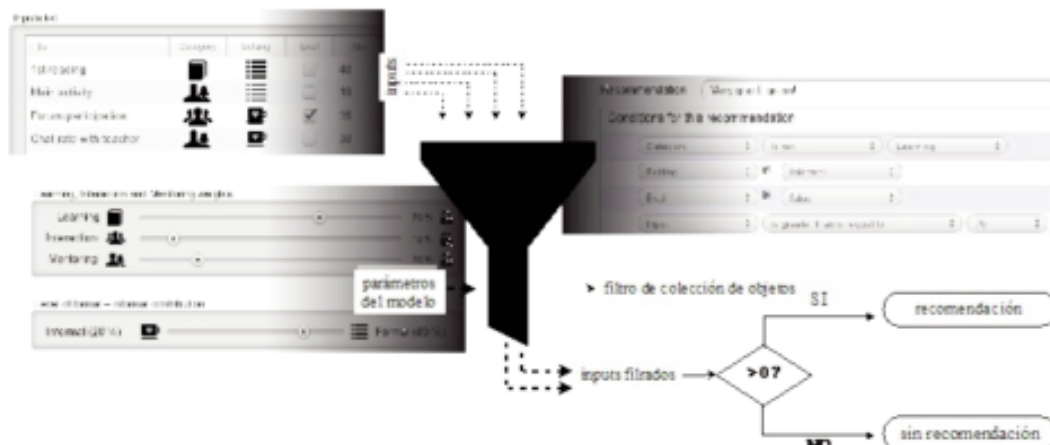


Fig. 6: Consola de administración de iLIME y aplicación de reglas a través de filtrado de colecciones de objetos.

## 6. Discusión.

El modelo LIME incorpora una metodología que busca encontrar el balance entre todas las actividades y acciones (inputs) en las que puede verse involucrado el alumno a lo largo de su recorrido educativo en un entorno de aprendizaje online. El modelo trabaja con los pesos específicos para cada uno de estos inputs, que juegan el papel de entradas en LIME. El modelo está basado en el comportamiento, capacidades y relación entre el usuario final y el resto de participantes. Además, propone cuatro vectores o pilares principales: Aprendizaje, Interacción, Tutoría y Evaluación (L.I.M.E). Cada uno de ellos aporta un factor decisivo para definir la relación del usuario final con su entorno, que es a su vez traducido en un conjunto de reglas interconectadas. Basándose en las acciones del usuario en el sistema y cómo se mueve de manera informal por el mismo, LIME devuelve al usuario recomendaciones personalizadas, guías, consejos, etc. preestablecidos por el tutor. La meta no es otra que hacer real y factible una labor de tutoría y soporte consistente a lo largo del proceso educativo.

Con el objetivo de evaluar el modelo LIME, se ha desarrollado iLIME como producto de software. Actualmente opera como herramienta LTI (parte de los estándares promulgados por el IMS Global) junto al gestor de contenidos académicos Sakai CLE. iLIME permite tanto la configuración sencilla del modelo por parte de administradores, profesores y tutores como la presentación y entrega final de recomendaciones a los alumnos. Todas estas operaciones tienen lugar de manera integrada y transparente en la propia instancia del LMS. A lo largo de este documento se han expuesto las peculiaridades que, a nivel técnico, han sido experimentadas en el transcurso del desarrollo de esta herramienta.

## Reconocimientos.

Esta actividad ha sido parcialmente financiada por UNIR Research (<http://research.unir.net>), Universidad Internacional de la Rioja (UNIR, <http://www.unir.net>), dentro del Plan Propio de Investigación, Desarrollo e Innovación [2013–2015]. Asimismo, los autores agradecen sinceramente el soporte, ayuda y base de conocimientos aportados por la comunidad de usuarios hispanohablantes del LMS Sakai CLE (S2U).

## Referencias.

- [1] R. A. Bjork, (1999). "Assessing our own competence: Heuristics and illusions," in *Attention and performance XVII. Cognitive regulation of performance: Interaction of theory and application*, D. Gopher and A. Koriari, Eds. Cambridge, MA: MIT Press, pp. 435-



459.

- [2] V. Romero and D. Burgos, (2010). "Meta-Mender: A meta-rule based recommendation system for educational applications," presented at Proceedings of the Workshop on Recommender Systems for Technology Enhanced Learning, RecsysTEL, Barcelona, Spain.
- [3] V. Romero, D. Burgos, and A. Pardo, (2011). "Meta-rule based Recommender Systems for Educational Applications," in Educational Recommender Systems and Technologies: Practices and Challenges, O. Santos and J. Boticario, Eds.: Information Science-Idea Group.
- [4] B. White and J. Frederiksen, (2005). "A theoretical framework and approach for fostering metacognitive development," Educational Psychologist, vol. 40, pp. 211-223.
- [5] G. Linden, B. Smith, and Y. J., (2003). "Amazon.com recommendations: Item-to-item collaborative filtering," Internet Computing IEEE, vol. 7, pp. 76-80.
- [6] B. Marlin, (2003). "Modeling user rating profiles for collaborative filtering," in Advances in neural information processing systems, S. Thrun, L. K. Saul, and B. Schölkopf, Eds. Cambridge, MA: MIT Press.
- [7] S. Y. Chen and G. D. Magoulas, (2005). Adaptable and Adaptive Hypermedia Systems. Hershey, PA: IRM Press.
- [8] K. I. Ghauth and N. A. Abdullah, "Learning materials recommendation using good learners' ratings and content-based filtering," Education Technology Research and Development.
- [9] T. Kerkiri, A. Manitsaris, and A. Mavridou, (2007). Reputation metadata for recommending personalized e-learning resources. Uxbridge: IEEE Computer Society.
- [10] C. Romero, S. Ventura, P. D. De Bra, and C. D. Castro, (2003). "Discovering prediction rules in AHA! courses.," presented at 9th International User Modeling Conferenc..
- [11] D. Burgos, C. Tattersall, and R. Koper, (2007). "How to represent adaptation in eLearning with IMS Learning Design" Interactive Learning Environments, vol. 15, pp. 161-170..
- [12] J. J. Rocchio, (1971). Relevance feedback in information retrieval, in the SMART Retrieval System. Experiments in Automatic Document Processing. Englewood Cliffs, NJ: Prentice Hall, Inc..
- [13] D. Burgos. (2013). "L.I.M.E. A recommendation model for informal and formal learning, engaged". International Journal of Interactive Multimedia and Artificial Intelligence - IJIMAI, 2, 79-86. DOI: 10.9781/ijimai.2013.2211.
- [14] Forment M., Casan-Guerrero M.J., Conde M., García F.J., and Severance C. (2011). "Interoperability for LMS: the missing piece to become the common place for e-learning innovation". International Journal of Knowledge and Learning. Vol. 6, pp. 130-141.
- [15] Paulo J. and Queirós R., (2011). "Using the Learning Tools Interoperability Framework for LMS Integration in Service Oriented Architectures". Conference Proceedings in Technology Enhanced Learning, TECH-EDUCATION '11. Springer Verlag.
- [16] S. Saigaonkar and M. Rao. (2010). "XML filtering system based on ontology". A2CWic '10: Proc. of the 1st Amrita ACM-W Celebration on Women in Computing in India, pages 1-6.
- [17] J. Cheney, S. Lindley, and P. Wadler. (2013). "A practical theory of language-integrated query". In Proceedings of the 18th ACM SIGPLAN international conference on Functional programming (ICFP '13). ACM, New York, NY, USA, 403-416.
- [18] Bosch, H, Heinrich, J., Muller, C., Hoferlin, B., Reina, G., Hoferlin, M., Worner, M. and Koch S., (2009). "Innovative filtering techniques and customized analytics tools". Visual Analytics Science and Technology. VAST 2009. IEEE Symposium.
- [19] Gonzalez, M.A.C., Penalvo, F.J.G., Guerrero, M.J.C.; Forment, M.A., (2009). "Adapting LMS Architecture to the SOA: An Architectural Approach". Internet and Web Applications and Services, 2009. ICIW '09. pp. 322-327.
- [20] Kelly D. and Thorn K. (2013). "Should Instructional Designers care about the Tin Can API?". eLearn Magazine, Issue 3.

Cómo citar este artículo / How to cite this paper

Corbí, A. & Burgos, D. (2014). LIME: un modelo de recomendación para entornos de aprendizaje online formal/informal. Campus virtuales, 3(1), 12-20.

Corbí, A. & Burgos, D. (2014). LIME: a model of recommendation for online learning environments formal/informal. Campus virtuales, 3(1), 12-20.