REVISTA COLOMBIANA DE COMPUTACIÓN
Volumen 15, número 2
Págs. 7- 37

Diseño de un Sistema Recomendador Híbrido de Objetos de Aprendizaje

Jacqueline T. Solís[*], Mario A. Chacón-Rivas[*], Cesar Garita[**]

Fecha de recibido: 28/06 /2014 Fecha de Aprobación: 09/09/2014

Resumen

El proceso de planeación o diseño de un curso se basa en gran medida en la selección de las actividades que están sujetas al criterio o conocimiento del docente; del mismo modo que dada la cantidad de recursos disponibles para un tema específico es muy grande y solo puede restringirse por el criterio del experto en la materia.

El Sistema Recomendador Híbrido de Objetos de Aprendizaje busca sugerir recursos de aprendizaje disponibles en distintos repositorios en la nube para un profesor a partir de los contenidos del Diseño Instruccional (DI) de su curso. Para esto, el sistema selecciona los descriptores textuales de los contenidos e intenta extraer las palabras clave que describen a cada actividad del DI de forma automática. Seguidamente, realiza búsquedas parciales con estos descriptores en una serie de repositorios y prioriza los metadatos de los Objetos de Aprendizaje (OA) recuperados de acuerdo con la afinidad que presenten con respecto a la descripción de las actividades de un curso.

De este modo, se le presentan al profesor solo aquellos ítems que cuenten con mayor afinidad en cuanto a los contenidos que describen una actividad mediante una interfaz gráfica unificada. Así, se aprovechan los OA creados por otros profesionales en la materia y se reduce la cantidad de recursos que debe revisar un profesor para elegir entre aquellos que le podrían interesar para utilizarlos en su lección.

Palabras clave: *Sistemas de recomendación en e-learning, Recomendación de objetos de aprendizaje, TEC Digital, SQI, ROA.*

Abstract

The course design is based on the selection of activities that are subject to the discretion or knowledge of the teacher. In the same way, the given amount of

[*] TEC Digital, Instituto Tecnológico de Costa Rica, Cartago, Costa Rica. e-mail: jacsolis@itcr.ac.cr; machacon@itcr.ar.cr
[**] Centro de Investigaciones en Computación, Instituto Tecnológico de Costa Rica, Cartago, Costa Rica. e-mail: cesar@itcr.ac.cr

resources available for a specific topic is very large and can only be restricted by the criterion of skill in the art .

Learning Objects Hybrid Recommender System seeks to suggest resources from Learning Object Repositories in the cloud for a teacher using the contents of an Instructional Design (ID) as input. To do this, the system selects the textual content descriptors and attempts to extract keywords describing each ID's activity automatically. Then, it makes partial matches with these descriptors in a number of repositories and prioritizes the Learning Object's metadata retrieved according to the affinity to the original content.

In this way, only the greater affinity items are presented to the teachers through an unified graphical user interface so the author does not have to spent a lot of time searching for the learning objects. Therefore, the teacher can take only the more relevant learning objects created by other professionals in the field and use them in his/her lesson.

Keywords: *Recommender Systems in e-learning, Learning Objects Recommendation, TEC Digital, SQI, LOR.*

1. Introducción

El proceso de planeación o diseño de un curso se basa en gran medida en la selección de las actividades que están sujetas al criterio o conocimiento del docente. Estas actividades, por lo general, son exposiciones magistrales en cursos tradicionalmente presenciales, mixtos o virtuales y se conforman en la mayoría de las veces de lecturas complementadas con actividades de discusión grupal como foros, blogs o wikis.

La disponibilidad de recursos digitales para un tema específico es muy grande, por lo que solo puede restringirse mediante criterios brindados por el experto en la materia a enseñar. Del mismo modo, determinar el nivel de calidad del contenido del objeto es difícil, lo que hace de la práctica de selección o recomendación de objetos de aprendizaje una actividad poco frecuentada por el docente.

Ante esta problemática se han creado investigaciones y proyectos que buscan adaptar e implementar soluciones tecnológicas en el apoyo de la recomendación de objetos de aprendizaje. En este dominio de investigaciones se debe decidir entre los algoritmos y teorías de recomendación, contemplar los intereses de los usuarios y el dominio de estudio requerido.

1.1 Contexto

Esta investigación se realizó en el contexto de la tesis de maestría en ciencias de la computación, en el Instituto Tecnológico de Costa Rica

(TEC). Específicamente se realiza dentro del TEC Digital, unidad responsable de la plataforma y desarrollos e-learning. Algunos de los desarrollos de apoyo a la docencia implementan soluciones de gestión de diseño instruccional, gestión de atributos, estilos de aprendizaje, entre otros[1]. Estas implementaciones conforman una base de recursos docentes y de apoyo al proceso de aprendizaje que complementan los servicios tradicionales de una plataforma de e-learning.

El proceso de ubicación, selección y recomendación de los contenidos o evaluaciones asociados a las actividades de un curso es largo y se basa en una serie de criterios que son brindados por un profesor experto en la materia. Además, dada la cantidad de recursos digitales disponibles el principal problema ya no es encontrar materiales sino, seleccionar aquellos que tienen algún grado de relevancia para la lección.

Es por este motivo que se propone un modelo para integrar técnicas de minería de datos en este proceso de búsqueda y selección de recursos. Para esto, se realiza un análisis de los recursos disponibles en un repositorio de objetos de aprendizaje (ROA), con el fin de identificar los factores críticos a considerar durante las fases de clasificación y recomendación de objetos de aprendizaje (OA); de modo que sea posible identificar aquellos con mayor afinidad con respecto a las descripciones brindadas en los documentos facilitados por el profesor y así, ofrecerle al profesor, una lista de OA que podrían resultar de interés para el desarrollo de sus lecciones.

En la segunda sección de este artículo se expone la metodología y los trabajos relacionados que dan forma a la propuesta de solución. En esta sección se presenta un resumen general de la teoría de los sistemas recomendadores, sus características y usos en el ámbito educativo. Como subsecciones se comentan algunas técnicas de recomendación empleadas en la industria, técnicas de hibridación empleadas para mejorar las recomendaciones, se enumeran herramientas de recomendación de carácter libre que se pueden utilizar y se presenta en términos generales la metodología utilizada en el diseño de la propuesta de solución.

En la tercera sección se presentan los resultados del diseño del agente recomendador híbrido de objetos de aprendizaje, su arquitectura, datos de entrada y los resultados de su aplicación en un caso de estudio.

Finalmente, se incluyeron una sección de conclusiones obtenidas tras la investigación y otra para las referencias bibliográficas utilizadas en el artículo.

2. Metodología

Esta sección presenta el contexto en el que se desarrolla la propuesta de trabajo; así como la fundamentación teórica y los enfoques que han sido utilizados por otras implementaciones de sistemas recomendadores en diferentes contextos de desarrollo.

2.1 Sistemas Recomendadores

Los sistemas recomendadores, según Ricci [3], son "sistemas de información que sugieren ítems a usuarios basados en su comportamiento y/u otros tipos de datos" y en su mayoría están enfocados a procesos de toma de decisiones donde el usuario no tiene suficiente conocimiento para evaluar las alternativas de ítemes que podrían resultar potencialmente de interés.

Según Manouselis et al. [5], se define recomendación como una forma particular de filtrado de información que explota los comportamientos anteriores de los usuarios y sus similitudes para generar una lista de información a la medida de las preferencias de un usuario final. De modo que son sistemas que deben ser capaces de predecir la utilidad esperada de cada ítem para solo seleccionar los que sean de interés y clasificarlos con base en algún mecanismo de comparación.

En el contexto particular de la educación, la cantidad de recursos potencialmente útiles en el apoyo a los procesos de aprendizaje puede resultar abrumadora; y es por este motivo que los sistemas recomendadores han tomado fuerza en el ámbito educativo. Estos sistemas difieren de sus contrapartes comerciales en que deben contemplar información como los estilos de aprendizaje, profundidad del conocimiento requerido o pretendido, entre otros. Por lo que existen modelos y teorías pedagógicas que tienen como misión enriquecer el aprendizaje y que deben tomarse en consideración antes de iniciar cualquier implementación [3,5].

Algunos ejemplos de sistemas recomendadores en procesos de aprendizaje que han sido mencionados por Manouselis et al. [5,7], son los siguientes:

Altered Vista: fue creado para propagar información sobre la calidad de los recursos de aprendizaje.

RACOFI (*Rule-Applying Collaborative Filtering*), el cual fue presentado de manera informal el 12 de Agosto del 2003 y su objetivo fue realizar recomendaciones a usuarios en línea acerca de audios

correspondientes a OA utilizando filtros basados en reglas y colaborativos .

QSIA (*Question Sharing and Interactive Assignments*): Es un sistema enfocado en compartir recursos que le permite al usuario determinar quiénes van a ser parte de su insumo en el filtro colaborativo .

CYCLADES: Es un sistema que propone la evaluación de distintos recursos digitales disponibles en repositorios através del OAI-PMH. Este sistema utilizó filtros colaborativos para determinar las posibles recomendaciones útiles a sus usuarios, enfocándose en el intercambio de documentos.

CoFind (*Collaborative Filter in N Dimensions*): Es un prototipo capaz de aprovechar recursos libres disponibles en internet y centró su enfoque en el uso de etiquetas para exponer los metadatos del contexto educativo.

ISIS: Utiliza un enfoque híbrido para recomendar rutas de navegación sobre recursos de aprendizaje utilizando información social relacionada con otros usuarios y metadatos de los estudiantes y actividades de aprendizaje.

TORMES (*Tutor-Oriented Recommendations Management for Educational Systems)*: Es un sistema recomendador semántico basado en conocimientos diseñado para reflejar las necesidades del estudiante al estar bajo distintos escenarios y al mismo tiempo, ofrecerle al educador control al seleccionar lo que se le va a estar presentando a sus alumnos. Fue diseñado por el grupo de investigación aDeNu de UNED España [11].

Willow: Es un sistema de evaluaciones adaptativo asistido por computadora mediante la aplicación de técnicas de minería de datos. Este sistema ya ofrecía soporte para modelos de usuario y procesamiento de lenguaje natural como su antecesor Atenea, y se le creó una extensión para ofrecer las recomendaciones en los distintos escenarios enfocándose en el proceso de selección de las preguntas [25].

SERS (*Semantic Educational Recommender Systems*) presentado en [6], aprovecha aspectos semánticos que dejan por fuera los sistemas recomendadores tradicionales, incluso los referentes a la usabilidad, accesibilidad e interoperabilidad de los recursos en distintos escenarios. Para esto los autores plantearon tres requerimientos mínimos al diseñar este tipo de sistemas: el uso de un *modelo de recomendaciones*, el uso de una *arquitectura orientada a servicios abierta* y una *interfaz de usuario* para desplegar los resultados.

2.2 Técnicas Empleadas en Recomendación

Dentro del contexto de los sistemas recomendadores hay tres componentes principales que conforman un sistema: se le llama "ítem" a los elementos que se van a recomendar, "modelos de usuario" a la codificación de las necesidades y preferencias de un usuario dentro del sistema y ¨transacciones¨ a las interacciones entre el usuario y el sistema [3,4].

Una práctica común dentro de los sistemas recomendadores, consiste en la combinación de dos o más técnicas con el fin de mejorar el desempeño de la posible recomendación para un modelo de usuario en particular. De este modo, se busca compensar las deficiencias de una técnica, con las fortalezas de otra. Sin embargo, para decidir cuáles técnicas se pueden combinar, es necesario conocer las principales características de cada uno de los métodos utilizados para su construcción.

Se han realizado numerosos trabajos en los cuales se recopilan las características principales de las técnicas utilizadas para generar recomendaciones [5,8]. Las principales técnicas para generar recomendaciones que mencionan estos autores son las siguientes:

Colaborativos: son aquellos sistemas que intentan generar las recomendaciones más acertadas basándose en las recomendaciones de otras personas que han tenido gustos o preferencias similares. Un perfil de usuario de este tipo, consiste típicamente de un vector de ítems y una clasificación o "*rating"* que aumenta continuamente con las interacciones de usuarios a través del tiempo.

Demográficos: se refiere a aquellos que usan la información demográfica de los atributos del perfil de un usuario para realizar su clasificación y asignarle las posibles recomendaciones de su interés.

Basados en utilidad: realiza las sugerencias tomando como base la utilidad de cada objeto para el usuario, por lo que el problema central consiste en cómo crear la función de utilidad para cada uno de ellos .

Basados en conocimientos: realizan inferencias lógicas mediante reglas para determinar las preferencias del usuario. Es decir, posee información sobre cómo puede un ítem específico, ajustarse a las necesidades de un usuario.

Basados en contenidos: utilizan una serie de descriptores iniciales tanto para los usuarios como para los ítemes que se van a recomendar; de modo que se puede usar esa información para predecir los intereses del usuario, es decir, entrena el perfil del usuario basándose en las características de los ítemes con los que ha interactuado.

Basados en preferencias: aquellos sistemas que intentan generar los posibles valores que darían los usuarios al evaluar aquellos ítemes que aún no han visto.

Una de las restricciones que se han señalado a los sistemas recomendadores es que sus resultados sugeridos no cuentan con precisión o certeza. Debido a esto se propone trabajar con técnicas de hibridación en las recomendaciones, como se describe en la siguiente subsección.

2.2.1 Técnicas de Hibridación

Las técnicas de hibridación se utilizan para compensar las debilidades de una técnica de recomendación con las fortalezas de otra; con el fin de mejorar la calidad de la recomendación que se va a presentar al usuario o solucionar algún problema como el "cold start", que se origina cuando ingresan nuevos usuarios al sistema y no se cuenta con información suficiente para generarle una recomendación.

Existen numerosas técnicas, algunas de ellas sensibles al orden y otras que no lo son. Por ejemplo, las técnicas de pesos, mixtas, intercambio y mezcla de características no son sensibles al orden; por otro lado, las demás sí pueden verse afectadas por el orden de las entradas [3]. Algunas de las técnicas comunes usadas para lograr la hibridación en dichos sistemas son:

Pesos: los resultados de varias recomendaciones se combinan en una sola respuesta.

Intercambios: el sistema cambia de técnica de recomendación de acuerdo con la situación.

Mixtos: se dan los resultados de diferentes recomendadores al mismo tiempo.

Combinación de características: las recomendaciones de diferentes orígenes de datos se combinan en un único algoritmo de recomendación.

Cascada: un recomendador refina la salida de otro.

Aumentar la características: la salida de un recomendador, se usa como característica de entrada de otro.

Meta-nivel: el modelo aprendido por un recomendador, se usa como entrada para otro.

La selección de la técnica apropiada para la hibridación de métodos depende del tipo de información con el que se va a trabajar y el objetivo final de realizar la combinación. Algunos autores, como R. Burke en [8], proponen que algunas combinaciones pueden resultar en un análisis redundante de los datos, o bien, pueden dar como resultado combinaciones que no son posibles o útiles para el usuario final. Es por este motivo que el análisis del contexto y los datos disponibles para el recomendador, juegan un papel crucial antes de iniciar cualquier diseño.

2.2.2 Herramientas de Recomendación

Al igual que en otros ámbitos del desarrollo de software, la reutilización de componentes busca reducir el tiempo invertido en realizar implementaciones de algoritmos bien conocidos y estudiados; de modo que no se le reste tiempo a la implementación y refinamiento de las nuevas propuestas de algoritmos [9].

Existen múltiples librerías y API de recomendación que se encuentran implementadas bajo licencias *OpenSource* y están disponibles para trabajos académicos y comerciales [9, 16,17,18,19,20,21,22]. En estas implementaciones se pueden encontrar algoritmos asociados a recuperación de información, aprendizaje automático y generación de diversos tipos de recomendaciones que permiten a quienes las utilizan, reutilizar códigos funcionales y optimizados. De modo que agilizan los procesos de implementación de la tecnología en diferentes contextos de operación y agilizan los procesos de validación para nuevas implementaciones.

2.3 Metodología Propuesta

La herramienta propuesta en este artículo busca sugerir OA que podrían ser de utilidad para la lección descrita por un profesor; por lo que como entradas se propone la utilización de los contenidos del DI y una clasificación de acuerdo con los tipos de documento para asociar la teoría de estilos de aprendizaje (TEA) [2] a los recursos sugeridos; ya que aportan una descripción detallada de los elementos que deben ser cubiertos durante la lección.

El diseño instruccional (DI) es tanto la herramienta como el proceso donde "se analizan, organizan y presentan objetivos, información, actividades, métodos, medios y el proceso de evaluación, que al conjugarse entre sí conforman el contenido de un curso con miras a generar experiencias satisfactorias de aprendizaje" [23, 24]. En el contexto del recomendador, el contenido del DI es tomado del Generador de Diseño Instruccional, que es una herramienta que busca "guiar al profesor en la planificación didáctica de sus cursos mediante la creación del DI, con el fin de generar experiencias satisfactorias de

aprendizaje en la educación superior" [23]; esta herramienta aporta descripciones textuales sin estructura y dadas en lenguaje natural sobre los contenidos de la lección que impartirá un profesor.

Las características del diseño de la solución involucran distintas áreas de conocimiento entre las que se destacan la cosecha de metadatos mediante agentes HTTP usando SQI, procesamiento de lenguaje natural (extracción de "*tokens"*, identificación de siglas, reducción de conjuntos de palabras, entre otras), sistemas recomendadores (identificación de perfiles y determinar posibles valores de utilidad o interés sobre ítemes desconocidos por el usuario) y minería de datos (preprocesamiento de datos semiestructurados, análisis de datos, identificación de las reglas del negocio, entre otras).

Dado que la herramienta requiere una serie de mecanismos que responden a distintas metodologías de desarrollo, se utilizó como modelo el planteamiento sugerido para trabajar los sistemas recomendadores al separar las fases de desarrollo de una solución en dos subproblemas: *Construcción del perfil del usuario* y *Generar la recomendación.* De este modo, se segmentó el problema en tareas específicas e internamente se adaptó el proceso de desarrollo a las características de los datos y subproblemas encontrados en cada fase en particular.

Otro elemento de importancia es que la solución debe responder a un modelo de usuario que represente las características del curso o de un estudiante para ser usadas en los procesos de búsqueda y filtrado de información. Según [10], la forma en que se construye un modelo de usuario puede responder a distintos propósitos:

- Identificar los procesos cognitivos que ocultan las acciones del usuario.
- Determinar las diferencias entre las habilidades de un usuario normal y un experto.
- Determinar los patrones o preferencias ocultas en el comportamiento del usuario, o
- Determinar las características del usuario.

Para nuestro contexto específico, la "adquisición del modelo del usuario" estará representando las actividades del curso para el que fue definido el DI, y se construye con el fin de determinar las principales características y necesidades que se deben satisfacer. Esto principalmente se debe a que no se cuenta en la actualidad con un modelo de usuario, ya sea profesor o estudiante, que nos permita recuperar estas características y necesidades a partir de la información disponible en la plataforma educativa. Además, un mismo curso puede ser impartido por múltiples profesores, o bien, puede ser impartido por

un mismo profesor en muchas ocasiones con variantes leves en sus contenidos; por lo que las búsquedas permitirían a largo plazo, acotar las búsquedas en microcontextos.

El proceso para "generar la recomendación" se centrará en el análisis de los metadatos de los OA, con el fin de determinar la similitud entre ellos y la descripción de cada actividad del DI. De este modo, se espera seleccionar el banco de recursos más afín a la descripción de la actividad antes de presentarlos al profesor.

3. Resultados

En esta sección se describe la arquitectura de la propuesta y los pasos seguidos para construir los perfiles y generar las recomendaciones de OA. Seguidamente, se presenta el proceso de desarrollo del prototipo, así como los resultados obtenidos en su aplicación sobre un caso de estudio específico para la herramienta.

3.1 Agente Recomendador Híbrido de Objetos de Aprendizaje

El agente ARHOA (Agente Recomendador Híbrido de Objetos de Aprendizaje) descrito en este artículo, se ubica en la categoría de híbrido, ya que utiliza distintas técnicas de recomendación para generar sus resultados mediante un recomendador basado en contenidos enlazado en cascada con un recomendador colaborativo; finalmente se desplegará la información mediante una representación mixta con la recomendación por estilos de aprendizaje.

Este agente toma la información textual del DI del curso para realizar un preprocesamiento con el fin de reducir las ambigüedades textuales. El trabajar solamente con la información de un curso puede ser poco representativa y deja un grado de confiabilidad y de soporte muy bajo en las reglas de asociación que se pueden identificar dentro de un conjunto de datos, por lo que fue necesario ampliar este conjunto tomando información de los cursos equivalentes en otros planes de estudio o carreras.

Los recursos de aprendizaje o items, provienen de ROA externos al servidor donde se encuentra la aplicación y no le pertenecen al TEC. Para realizar las conexiones con ellos se utiliza "Simple Query Interface" (SQI); de modo, que la información disponible sobre cada item viene dada en datos semiestructurados que en muchas ocasiones están incompletos o bien, son ambiguos con respecto a su contenido o etiquetado.

3.1.1 Arquitectura de ARHOA

La arquitectura se basa en un modelo de integración de fuentes de datos empleando un bus de datos XML y descomponiendo el problema en subfases de búsqueda y análisis.

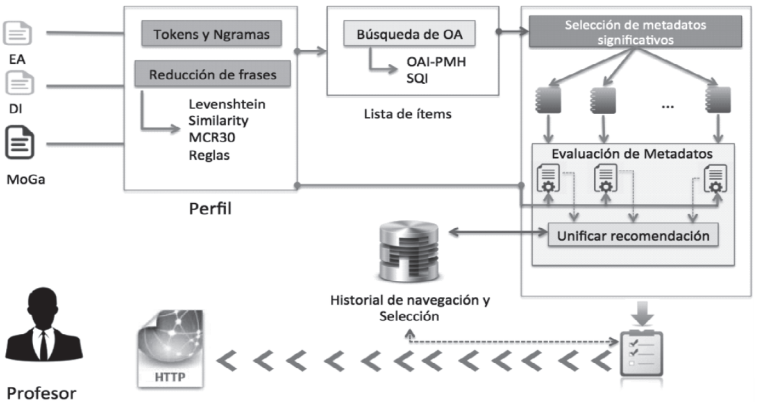

Fig. 1. Arquitectura General del ARHOA.

La Fig. 1 muestra la arquitectura general del ARHOA. La construcción del perfil de un curso usa las descripciones dadas en un diseño instruccional para extraer las palabras y frases que describen cada actividad.

La selección de frases y palabras que se conservan después de la construcción del perfil del curso conforman una bolsa de términos que describen una aproximación de los contenidos que debería cubrir una actividad. Esto es útil porque el SQI permite realizar búsquedas sobre uno o más ROA en la nube para recuperar solo los metadatos de los OA que contengan ciertos textos o bien, satisfagan reglas de búsqueda definidas por el cliente al consumir el servicio y son retornados en XML, como se desarrollará en la sección 3.2.

Los metadatos de los OA candidatos contienen al menos una de las frases que fue recuperada del perfil del curso y se utiliza la información que aportan sus descripciones para analizar los contenidos textuales. De estos metadatos de los OA que responden a las consultas y vienen dados en LOM-IEEE, se procede a extraer los descriptores que fueron seleccionados como significativos. En nuestro caso, se seleccionaron los que se encuentran asociados a las descripciones del dominio educacional, general y técnico de cada objeto.

Una vez extraídos los contenidos textuales y construídos los modelos que describen a cada OA, se realiza una comparativa con respecto a la descripción dada para una actividad y se unifica la recomendación a

partir de la aplicación de dos medidas de similitud antes de presentarle los resultados al usuario final.

Dado el volumen de datos que deben ser consumidos en cada consulta, se definió que el recorrido sobre los resultados de la búsqueda se realizará de forma progresiva. Para esto fue definido un tope de metadatos a evaluar en cada iteración y la siguiente vez que inicie la búsqueda, se iniciará a partir del último recurso recuperado hasta recorrer la totalidad de la cardinalidad de cada respuesta. El punto en el cual termina de recorrerse una respuesta del repositorio se puede determinar a partir de un comando del SQI que retorna la cardinalidad esperada para una consulta en particular.

Una vez que se analizan los metadatos de un OA se registran en una base de datos sus identificadores y los textos que fueron extraídos, así como los resultados obtenidos tras evaluarlo para una actividad específica; de modo que no es necesario recalcular los datos tras la siguiente iteración y los ordenamientos se realizan sobre la totalidad de items asociados a una actividad específica. Este modelo de datos es provisional, ya que se están implementando modificaciones para trabajar directamente sobre modelos vectoriales para aprovechar las nuevas funcionalidades del motor de bases de datos.

Para consumir recursos de un ROA es necesario construir o utilizar un cliente que ofrezca soporte para la cosecha de metadatos respetando las normativas establecidas para los agentes robóticos HTTP.

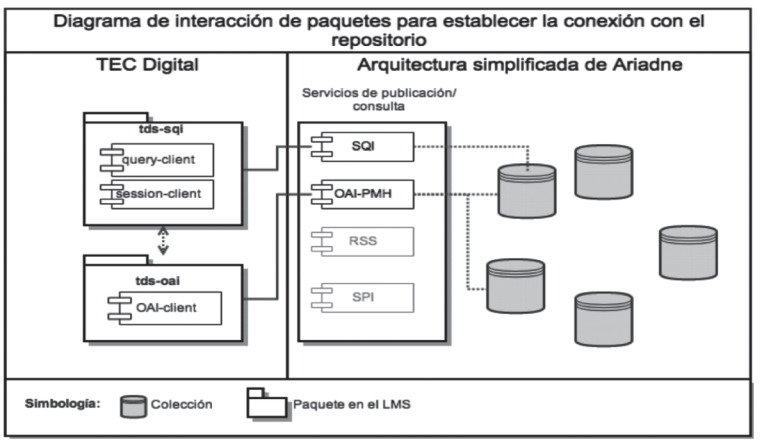

Fig. 2. Diagrama simplificado de paquetes para conexión con el ROA.

En esta primera versión de la arquitectura de integración, se utilizó el modelo de conexión de ARIADNE como marco de referencia. Este ROA dispone de una serie de servicios y tecnologías para exponer sus

recursos. Entre las tecnologías que se utilizan, destacan el uso de OAI-PMH para realizar cosecha entre repositorios, SQI para realizar consultas, SPI para realizar publicaciones y RSS para notificar a los interesados sobre los nuevos OA agregados a las colecciones disponibles [15]. La selección de uno u otro mecanismo para consumir los recursos depende de las necesidades del usuario final; en nuestro caso particular, se intentó realizar la cosecha mediante dos mecanismos: el OAI-PMH y el SQI como se puede apreciar en la Fig. 2.

3.2 Proceso de Recomendación

A continuación se presenta el procedimiento utilizado parar construir los perfiles de un curso y generar las recomendaciones para el usuario final.

Actualmente se trabaja con colecciones textuales reducidas para construir los perfiles que describen a cada curso y se enfocó en una solución de propósito general para los cursos de distintos contextos. Por esta razón no se están utilizando diccionarios de dominio específico en este momento, pero no se descarta su posterior integración a la herramienta.

3.2.1 Construcción del Perfil del Curso

Dado que el DI viene dado en lenguaje natural, se debe buscar un mecanismo que simplifique los procesos de búsqueda y procesamiento de textos. Por este motivo, se utilizaron técnicas para reducir los bloques de texto expresados en lenguaje natural a un subconjunto de palabras que sean significativas para describir el curso y más útiles a nivel computacional; específicamente mediante el uso de palabras de parada en español y reglas para reducir las expresiones.

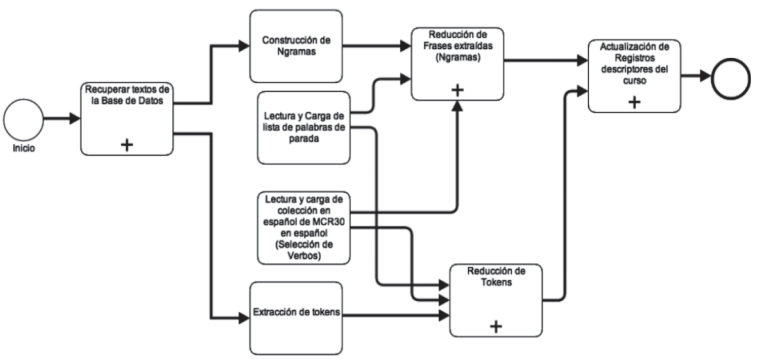

Fig 3. Diagrama simplificado de preprocesamiento para perfiles de curso.

La Fig. 3 muestra una simplificación del proceso que se siguió para extraer, reducir y corregir los textos introducidos por un usuario en el DI, con el fin de iniciar la construcción de un perfil para un curso y cada una de sus actividades. En este proceso se realizó una selección de palabras y frases para búsquedas a partir de los textos brindados en lenguaje natural por el profesor.

La colección de wordnet español 3.0 (MCR30) ofrece información sobre frases y palabras en español previamente clasificadas; de modo que nos ofrece un marco de referencia para identificar las categorías gramaticales de los textos introducidos en el DI. Por otro lado, el uso de listas de palabras de parada también aporta otro marco de referencia para acotar de la cardinalidad del modelo vectorial que describe una actividad.

Para realizar la evaluación de la relevancia de cada uno de los términos extraídos se aplicaron los siguientes pasos:

1. Separar *tokens* y *Ngramas* a partir de los contenidos textuales. En este caso, los *Ngramas* son grupos de 2-N palabras, la extracción se realizó utilizando WEKA.
2. Conteo de apariciones de las palabras y cálculo del TF-IDF de las palabras que están asociadas a las distintas actividades del curso.

La reducción de los elementos a considerar en la construcción de los vectores que describe a cada actividad se realizó utilizando reglas sobre las elementos extraídos con los tokens y frases como se expone a continuación:

a. Comparación de palabras y frases para reducir los conjuntos que describen a una actividad: Para esto se utilizó la distancia de Levenshtein y la medida de Similarity provista por pg_trgm en PostgreSQL 8.4, con el fin de agrupar aquellas que podrían estar relacionadas, pero con errores ortográficos.

La distancia de Levenshtein indica la cantidad de transformaciones necesarias para convertir una palabra en otra. Esta distancia aplica para textos de distintas longitudes, a diferencia de la distancia de Hamming tiene como condición que sean de una misma longitud. Por otro lado, la medida provista por Similarity, indica cuál es el porcentaje de similitud que existe entre frases a partir de los trigramas que componen dos palabras.

Para establecer las cotas de relación que permitieran restringir los pares de palabras con posibles correcciones ortográficas, se utilizó

un conjunto de entrenamiento asociado al caso de estudio que fue clasificado de forma semiautomática para buscar una relación entre estas dos variables. Dado que uno de los objetivos iniciales no fue hacer un procesador de lenguaje natural completo, se realizó una aproximación de corrección ortográfica con estos valores y se redujo significativamente la cardinalidad del conjunto de atributos que describen a cada actividad.

b. *Descartar verbos compuestos que aparezcan como Ngramas, o bien, aquellas frases que terminen con un verbo*: se utilizó la colección de verbos disponibles dentro de MCR30 para identificar verbos compuestos y descartarlos del conjunto de frases que describen una actividad cuando aparezcan solos.

c. *Reglas generales para descartar textos basadas en el idioma español (stripping)*: Se descartaron números, monosílabos y frases que inicien o terminen con palabras de parada.

Los textos que se conservaron serán utilizados para realizar consultas contra el repositorio de objetos de Aprendizaje mediante SQI o bien, OAI-PMH. De este modo, se pueden adquirir una serie de OA que serían los items candidatos a ser sugeridos para cada actividad en particular

.

Una vez identificadas las palabras y frases a tomar en consideración para cada actividad, se construyeron reglas para realizar una preselección de OA. Esto se debe a que una misma palabra puede pertenecer a más de un idioma, y la cardinalidad de los conjuntos de items candidatos a ser una respuesta excede la capacidad computacional del equipo en el que se ejecuta actualmente.

Según la especificación de LOM se puede identificar el idioma de un OA utilizando los metadatos correspondientes a general.Language, meta-metadata.language y educational.language [12]. Sin embargo, los metadatos incompletos o mal etiquetados dentro de los ROA dificultan el proceso de preselección de los recursos candidatos y agregan fases de preprocesamiento y aproximaciones necesarias para delimitar los elementos preseleccionados.

Con el objetivo de aprovechar algunas características del idioma español y la sintaxis de consultas ofrecida por el SQI [13], se optó por la aplicación de reglas que toma en cuenta los conectores gramaticales como las partículas “y”, “e”, “u” y “o”. Con esto se restringió la cantidad de OA que responden a una expresión cuando esta excede la cardinalidad de respuesta esperada como tope para la transacción.

3.2.2 Generar la Recomendación

En términos generales, determinar si un OA es relevante o no para una actividad específica, puede ser una decisión ambigua; ya que depende del contexto y del criterio de quien esté eligiendo el conjunto de actividades. Además, utilizar únicamente el soporte brindado por los documentos originales del DI y del MoGa del curso, sería restringir el conjunto de datos disponibles en otros contextos que pueden tener relación directa con las actividades; por lo que se reducirían los niveles de confiabilidad debido a la escasez de información.

Cómo parte de la información obtenida de la base de datos, se pueden identificar al menos cinco categorías o grupos de pertenencia para las actividades recuperadas en el momento de intentar generar una recomendación para un curso:

- La actividad en evaluación (C1).
- Actividades pertenecientes al mismo curso (C2), pero que no son la que estamos evaluando en el momento.
- Actividades descritas en cursos equivalentes que tienen similitudes con la actividad en evaluación (C3).
- Actividades descritas en cursos equivalentes que tienen similitudes con otras actividades del DI, pero con poca similitud con la actividad que está en evaluación (C4).
- Pertenecientes a cursos equivalentes, pero que no tienen similitud con ninguno de los descriptores del DI (C5). En este caso, se consideraría que la actividad descrita es específica del dominio del curso equivalente; por lo que no se tomará en cuenta para evaluar los descriptores.

Los cursos equivalentes son determinados por el departamento de admisión y registro; en la programación, el recomendador utiliza los servicios web disponibles para consultar los códigos de cursos equivalentes al curso asociado a un DI y con esta información, recupera los contenidos de otros DI que podrían contener descriptores que podrían resultar de interés. Tras normalizar los vectores con descriptores que describen tanto a la actividad que estamos analizando, como a las candidatas a ser equivalentes; se les aplica similitud de cosenos y se consideran similares aquellas que cumplan con al menos un 60% de similitud entre sus contenidos.

Para cada una de estas posibles categorías se desea determinar el grado de influencia que será utilizado al calcular la función de peso de cada una de las palabras clave; sin embargo para esta primera fase el trabajo se concentró en las categorías C1 y C2.

Al generar la recomendación es necesario realizar una comparación de los metadatos contenidos en los OA preseleccionados, que en nuestro caso representan los items a recomendar, con respecto a la descripción dada en el perfil de cada actividad del curso.

		Términos							
		t1	t2	...	t_{n-1}	t_n	t_{n+1}	...	t_k
Act. Del curso	$Act._1$	0.003	1.03	...	0	0	0.6	...	...
	$Act._2$	4.000	0.83	...	0	0	0	...	...
	...	...	...	...	...	...	...	...	...
	$Act._N$	...	...	...	...	...	...	...	...

Fig. 4. Perfil de curso.

Como se muestra en la Fig. 4, el perfil de un curso está compuesto por modelos vectoriales que representan a cada actividad. Cada uno de estos vectores contiene la cantidad de apariciones, o bien, la frecuencia de aparición de los términos que sobrevivieron al proceso de reducción de textos y consulta en los ROA.

Al preprocesar los metadatos de los OA candidatos recuperados de los ROA, se extraen los metadatos que aportan descripciones (*general.description*, *educational.description* y *technical.description*) y se les aplican dos medidas para determinar su proximidad con respecto al modelo vectorial que representa al perfil:

1. Similitud de cosenos: Se calcula la distancia del vector que representa a cada OA con respecto al vector que describe la actividad. Para aplicar este algoritmo se utiliza aritmética vectorial para determinar el ángulo que representa la similitud entre un documento d_1 representado en un vector v_1 y el documento d_2, representado mediante un vector v_2. En nuestro caso en particular, uno de los documentos sería la descripción de la actividad; mientras que los restantes d_n son las representaciones de los OA en evaluación.
2. TF-IDF: Calcula la relevancia que tiene cada término en un documento, con respecto a una colección general de documentos. Es decir, para cada aparición de un término asigna un valor que refleja la importancia que tiene ese término sobre el documento que lo contiene tomando en consideración el total de veces que aparece en todos los documentos.

Estos dos algoritmos se aplicaron en cascada, es decir, se aplicó primero la similitud de cosenos y se seleccionaron aquellos top-N descripciones de OA que poseen mayor similitud con la descripción de la actividad.

Posteriormente, se refinó el ordenamiento de los recursos recomendados con el TF-IDF.

Por ejemplo, suponiendo que el tope de metadatos para analizar en una iteración esté definido en 60, se estarían analizando 60 OA nuevos recuperados del ROA por cada palabra o frase que describa una actividad. De estos descriptores, se construirán vectores normalizados con la descripción de la actividad, con el fin de determinar cuáles son los que poseen la similitud de cosenos más cercana a la actividad y se registra en la base de datos. Seguidamente, se calcula el TF-IDF de los metadatos recuperados en relación con actividad incluyendo los N-gramas, de modo, que se toman en consideración tanto los tokens como las frases para considerar la similitud de los recursos y establecer el orden en que serán desplegados en pantalla incluyendo aquellos metadatos de otros OA que fueron analizados previamente.

Para la extracción de los tokens y N-gramas tanto en la construcción de los perfiles como en el análisis de los metadatos se utilizaron las funciones de manipulación de textos disponibles en WEKA y posteriormente, se refinó la reducción y se aplicó la corrección ortográfica antes de calcular las medidas de similitud entre los vectores. Además, es importante rescatar que las descripciones de las actividades dadas por los profesores suelen ser textos cortos que no superan las 80 palabras en lenguaje natural sin preprocesar; de modo que se cuenta con palabras o textos muy cortos para usar como referencias una vez preprocesados los contenidos.

Adicional a este mecanismo de recomendación, se construyó un clasificador utilizando una serie de reglas a partir de datos recopilados con un experto en educación. Los datos utilizados corresponden a distintos tipos de OA y el aprovechamiento que se esperaría al aplicarlos a estudiantes con preferencias por distintos tipos de Estilos de Aprendizaje.

Este clasificador requiere de un proceso de validación formal y de un conjunto de datos de entrenamiento mayor para darle valor estadístico. Además, los estilos de aprendizaje están vinculados a individuos más que a documentos; de modo, que se requiere de la existencia de modelos de usuario (*User Model*) para los estudiantes que utilicen los recursos recomendados con el fin de determinar su utilidad práctica.

ARHOA emplea de características de los sistemas colaborativos (aplicación de la similitud de coseno y balanceo de pesos con las evaluaciones de los usuarios), demográficos (delimitación de vecindarios de acuerdo con las características del curso) y de los basados en contenidos (aplicación del TF-IDF) combinadas en los

procesos asociados al entrenamiento y generación de la recomendación. Además, la construcción de la hibridación al generar las recomendaciones se da mediante el uso de técnicas como el procesamiento en cascada y manejo de pesos (ajustes de peso mediante metadatos y aumento de características) y la utilización de representaciones mixtas (al incluir información sobre los estilos de aprendizaje en la clasificación del OA).

3.3 Diseño Visual del Prototipo

El prototipo está pensado para operar en dos formas: como un buscador de OA y como un recomendador; de modo que el usuario final no está restringido por la existencia o no de un diseño instruccional para aprovechar la conectividad brindada por la aplicación.

El diseño de interfaces de usuario accesibles y eficaces ha tomado fuerza en los últimos años; de modo que ya no solo se busca generar recomendaciones; sino que el despliegue de la información sea útil para quien la reciba.

El TEC Digital cuenta con un equipo de profesionales encargados de velar por esta misión, y se encargan tanto de los procesos de diseño, como los de validación una vez que ha sido puesto a prueba una implementación.

La interface de usuario sugerida por el equipo de comunicación visual, busca conservar la armonía con los diseños de las interfaces previamente diseñadas para la plataforma, como se puede apreciar en la Fig. 5; que corresponde a la pantalla de selección de OA para el usuario de la aplicación.

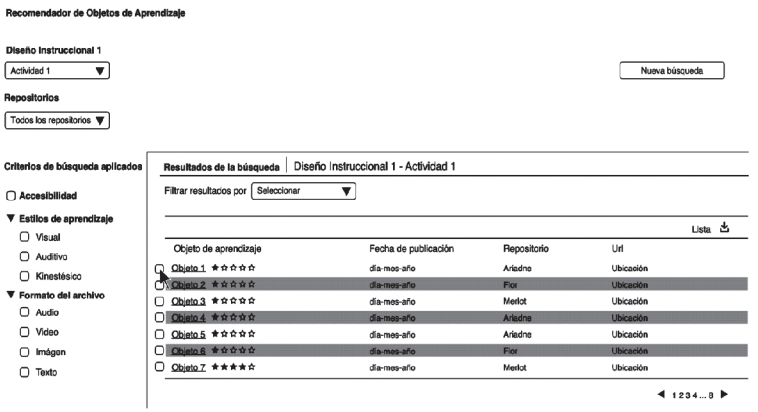

Fig. 5. Vista del menú para selección de actividades.

Posteriormente, se le ofrecerá al usuario final una serie de OA ordenados de acuerdo con la afinidad que tiene con la actividad seleccionada.

En esta pantalla se incluyó un mecanismo de valoración para que el profesor tenga, de manera opcional, la oportunidad de indicar la utilidad que encontró en los OA que le fueron desplegados. A partir de esta información, se puede ajustar el ordenamiento de los ítems a desplegar, ya que la información disponible dentro de los metadatos del OA permite reajustar el perfil del curso.

Como se puede apreciar en la Fig. 6, esta pantalla incluye elementos de navegación y está pensada para que las preferencias del usuario final se almacenen mediante un clic sobre el OA en pantalla.

Los detalles de un OA estarán disponibles mediante un clic sobre la lista que se despliega en la pantalla de inicio y será posible visualizar algunos de los metadatos relevantes disponibles a través de la descripción dada en LOM.

En el detalle se incluye el hipervínculo de descarga para el OA, de modo que si un usuario desea, puede ir directamente al URL origen del recurso y evaluarlo por sí mismo, más allá de la información disponible en la descripción general.

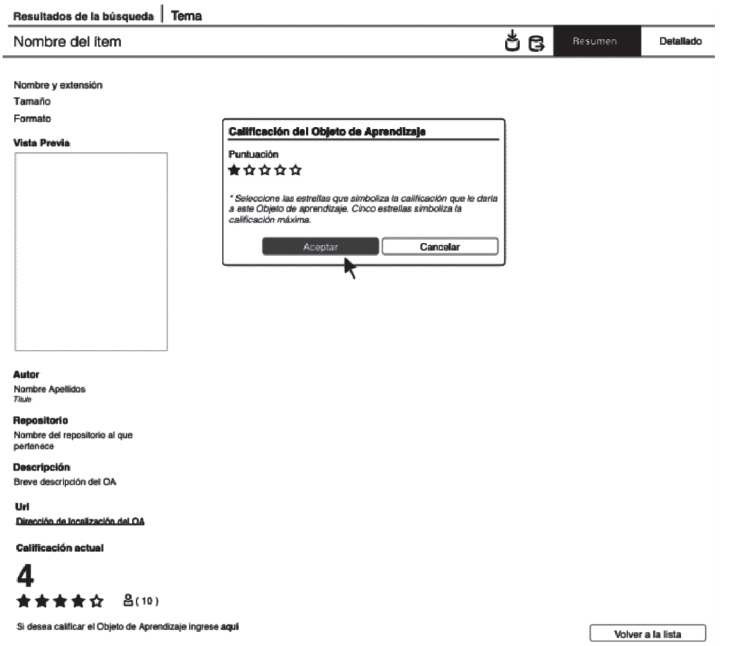

Fig. 6. Detalle del OA seleccionado por un usuario.

Como mecanismo adicional, se incorporó un elemento para capturar de forma opcional, la evaluación de los recursos sugeridos por parte del usuario final y se estará incorporando una herramienta para capturar la opinión del usuario sobre el funcionamiento general de la aplicación con el fin de detectar posibles mejoras tras su puesta en producción.

3.4 Caso de Estudio

Como caso de estudio se trabajó sobre un DI incompleto para el curso TI-3600: Bases de Datos para la Administración de Tecnologías de Información. El documento utilizado para el entrenamiento consta de ocho actividades brindadas en lenguaje natural que fueron insertadas en la aplicación del DI a partir del documento de referencia facilitado por el profesor.

Como se muestra en la Fig. 7, a nivel estructural, la descripción brindada para cada actividad en el documento incluye viñetas, errores ortográficos, siglas, elementos numéricos, entre otros. Estos elementos son típicos de la estructuración utilizada al redactar los contenidos de las unidades o sesiones y tanto la longitud de los textos, como la calidad de los contenidos dependen de quién introduzca los datos al DI.

```
Evaluar lecturas asignadas la semana anterior.
Presentación de conceptos de introducción a la teoría de SABD: tablas, registros, indices y llaves.
Proceso de construcción de una base de datos:
        Etapas actividades
        1.      Proceso práctico de instalación del SQL Server
                a.      Instalación del Sql Server 2000, a partir de 0
                b.      Presentación de la arquitectura física.
                        i.      Rutas de archivos de programas, datos, respaldos.
                        ii.     Nombre de archivos físicos y sus extensiones (.mdf, .ldf)
```

Fig. 7. Muestra de datos para la actividad 4 del DI del curso TI3600.

Como parte del preprocesamiento de datos, se realizaron pruebas con distintos mecanismos para extraer las palabras y frases relevantes. Para el caso particular de la extracción de tokens derivados de los textos se realizaron las pruebas con WEKA utilizando el WordTokenizer con radicación explícita para español, sin radicación y de las herramientas de análisis provistas por ts_vector se realizaron pruebas con radicación en español, sin radicación y sin específicar el idioma (en cuyo caso, se toma inglés por omisión).

Tras la aplicación de los distintos mecanismos para extracción de tokens sobre las descripciones brindadas para el DI, se obtuvieron cinco colecciones de palabras con distintas cardinalidades para cada actividad. En la Fig. 8 se muestra un resumen de la cardinalidad de los resultados por cada actividad. En términos generales, el método que retornó una menor cantidad de palabras fue la extracción de tokens utilizando los

ts_vector con el proceso de radicación para el idioma español activo; sin embargo, la utilización de las raíces de las palabras recuperadas para realizar las búsquedas en un ROA no retornó elementos coherentes para ser considerados en un proceso de recomendación.

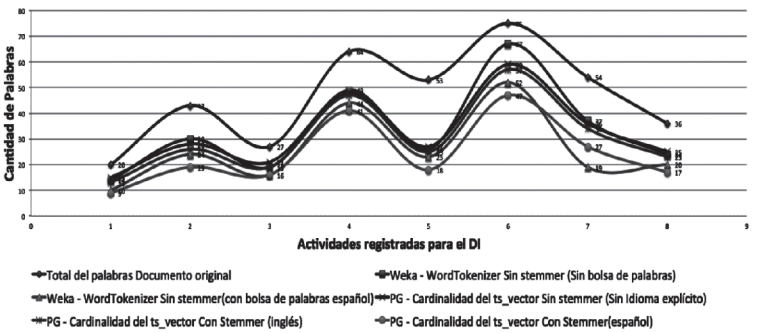

Fig. 8. Cardinalidad de los procesos de extracción de tokens para el caso de estudio.

El segundo método que generó una menor cardinalidad fue el generado por el WordTokenizer sin radicación y con una bolsa de palabras de parada en español como elementos para descartar. La lista de palabras de parada fue tomada de los diccionarios de PostgreSQL 9.1 en español y se aplicó como diccionario en el proceso de reducción de los tokens extraídos por la funcionalidad básica de la herramienta. En este caso, los resultados de las búsquedas en un repositorio fueron más coherentes y aunque no fue posible capturar la condición donde una palabra puede pertenecer a más de un idioma; se seleccionó este método como mecanismo para extraer los tokens a ser considerados para un perfil de curso.

Los otros métodos utilizados arrojaron conjuntos ligeramente mayores en cuanto a la cardinalidad de la respuesta obtenida; sin embargo, no se descarta que para otros DI o colecciones de texto mayores puedan arrojar mejores resultados que las dos primeras opciones tomadas en consideración.

Como ROA fuente para realizar las consultas se utilizó el repositorio Ariadne, el cual, permite realizar búsquedas cruzadas con otros repositorios que pertenecen a GLOBE como LORNET, MERLOT y FLOR; además, cuenta con miles de OA disponibles que pueden descargarse en múltiples formatos.

Este repositorio da soporte principalmente a IEEE-LTSC LOM, pero también soporta DublinCore y el ISO/IECMLR y puede realizar la conversión entre tipos de metadatos de forma automática. Otra de sus

ventajas, es que cuenta con una herramienta de búsqueda denominada ”ARIADNE finder” que permite realizar búsquedas sobre el repositorio y esconde los protocolos y estándares de la vista del usuario.

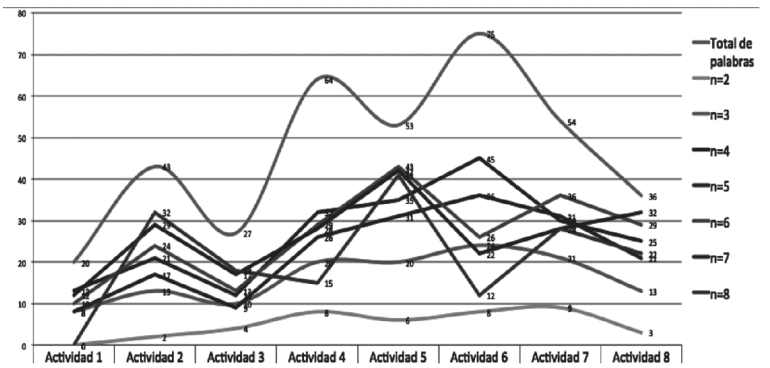

Fig. 9. Cardinalidad de los procesos de extracción de N-gramas para el caso de estudio.

Para considerar un N-grama (el equivalente a una frase) como candidata, se marcó como condición que aparezca más de una vez alguna sección del documento y se ignoraron los signos de puntuación como elementos de parada ver Fig. 9. Las pruebas con el caso de estudio se realizaron para frases con una longitud entre dos y ocho palabras. En cuanto a las cantidades de respuestas obtenidas, se hizo evidente que a mayor longitud en los textos originales, mayor cantidad de combinaciones y sin embargo esto no garantiza que el texto generado corresponda a una frase real; tampoco fue posible identificar algún factor de correlación entre la longitud de los Ngramas y la posibilidad de obtener una respuesta válida por parte del ROA.

Una vez unificados los descriptores textuales candidatos para el DI, se procedió a construir un modelo en XML que represente al curso y las posibles variantes de consultas válidas para aplicar de acuerdo a la sintaxis del SQI. Este modelo incluye categorías para los tipos de frases construídas, de modo que a nivel de aplicación se pueden aplicar reglas a las consultas para intentar reducir la cardinalidad de las respuestas obtenidas por parte del ROA. En la Fig. 10 se muestran algunos ejemplos de distintas clases asignadas a los textos, donde se dió prioridad a las frases candidatas generadas y a los tokens con mayor frecuencia de aparición. Por ejemplo, los elementos de “clase 3” de la figura incluyen conectores gramaticales del tipo “y”, de modo que se pueden descomponer en reglas de tipo “and” al realizar las consultas con la sintaxis PLQLv1 del SQI y de este modo, se restringen los metadatos que cumplen la condición solicitada en la consulta.

```
<curso id_di="15051">
   <actividad id="actividad_1">
      <termino clase="2">bases</termino>
      <termino clase="1">conceptos de datos</termino>
      <termino clase="3">conceptos de datos e información</termino>
      <termino clase="3">datos e información</termino>
      <termino clase="3">entrega y presentación</termino>
      <termino clase="3">entrega y presentación del programa</termino>
      <termino clase="2">introducción</termino>
      <termino clase="1">presentación de conceptos</termino>
      <termino clase="1">presentación de conceptos de proceso</termino>
      <termino clase="1">presentación del programa</termino>
      <termino clase="1">presentación del programa del curso</termino>
      <termino clase="1">programa del curso</termino>
      <termino clase="2">programas</termino>
</actividad>
<actividad id="actividad_2">
      ...
</actividad>
<actividad id="actividad_3">
      ...
</actividad>
</curso>
```

Fig. 10. Fragmento de un modelo de XML para representar los descriptores textuales de un DI.

Este perfil de cada curso es la representación de la lista de frases que podrían describir a cada una de sus actividades. Con la información que aporta, se valida la cardinalidad de las respuestas que tiene registradas el ROA a consultar. De este modo, se pueden descartar de la lista de items de búsquedas aquellas frases que no poseen ningún OA con metadatos que coincidan con la consulta.

En cuanto a los resultados obtenidos, tras aplicar los algoritmos fue posible recuperar OA con temas relacionados a los textos; sin embargo las ambigüedades del idioma dificultan la tarea de asignar valores precisos de afinidad. Por lo que se está trabajando en selección de un algoritmo de radicación para mejorar la calidad de las aproximaciones una vez finalizado el proceso de preselección de OA y está pendiente una validación con usuarios expertos en los cursos que aportan el DI para determinar la validez y precisión de las sugerencias de recursos.

A partir del análisis de los contenidos dados por un DI, se hizo evidente la necesidad de mejorar las descripciones brindadas por los profesores al describir los cursos que se impartirán. Las descripciones pobres dificultan la búsqueda de recursos y su posterior sugerencia para las actividades.

En la preselección de los OA y durante el proceso de sugerencia de recursos, se encontró que la calidad de los contenidos de los metadatos no es la más adecuada para establecer parámetros de selección eficientes. Existen problemas con el etiquetado y las funciones del SQI implementadas en algunos ROA están incompletas; sin embargo, falta realizar pruebas con otros repositorios que podrían brindar mejores resultados para el contexto operativo de la herramienta. Además, para aprovechar mejor los metadatados de los OA es necesario incorporar

tareas de preprocesamiento que no fueron contempladas en el diseño inicial del proceso de recomendación; de modo que sea posible identificar las asociaciones por radicación que genera de forma automática el ROA al realizar la búsqueda de metadatos
.
En cuanto al OAI-PMH, su principal dificultad radica en que no siempre es posible conocer la cardinalidad de la respuesta que se obtendrá; de modo que a nivel computacional es complejo administrarlo en memoria y no es funcional recuperar todos los recursos que satisfacen alguna condición de búsqueda con cada iteración; sin embargo, la disponibilidad de sus funciones parece ser mayor que la que ofrece el SQI.

La validación del proceso para generar recomendaciones a partir de información incompleta requiere de la incorporación de personal especializado en procesos de enseñanza y aprendizaje. De este modo, se podría aprovechar su conocimiento para validar los resultados y colaborar con el diseño de las adaptaciones del algoritmo de recomendación para contrarrestar los efectos de los datos faltantes; además, existen oportunidades de investigación en otras áreas de interés que van más allá de solo minería de datos, procesamiento de lenguaje natural y entornos educativos.

4. Conclusiones

A modo general, la propuesta de un proceso de automatización para la recomendación de OA a partir de los documentos del DI y la teoría de estilos de aprendizaje es viable. Sin embargo, falta madurar los procesos de diseño y representación de los metaelementos que son necesarios para llevar a cabo esta tarea.

ARHOA es un sistema recomendador que se ubica en la categoría de híbrido por el uso de características de los sistemas colaborativos (aplicación de la similitud de cosenos), demográficos (delimitación de vecindarios de acuerdo a las características del curso) y de los basados en contenidos(aplicación del TF-IDF) para generar la recomendación en cascada y mediante representaciones mixtas.

Al generar el modelo para clasificar los estilos de aprendizaje se encontró una serie de elementos sin métricas numéricas definidas dentro de la documentación analizada. Estos elementos describen características que son relevantes para la toma de decisiones con respecto a cuáles recursos seleccionar y su interpretación es un reto para los sistemas recomendadores. Es necesario trabajar y realizar propuestas en la especificación de estándares para especificar los estilos

de aprendizaje en los OA; en la actualidad esto se debe realizar en los metadatos empleando lenguaje natural, lo que trae problemas claros de idioma, instrumentos de medición empleados, entre otros.

La disponibilidad de textos sin calidad en cuanto a la redacción, contexto, nivel educativo esperado, entre otros; dificulta las tareas de preprocesamiento y validación de las hipótesis que se pueden plantear a partir de ellos.

La validación del proceso para generar recomendaciones a partir de información incompleta requiere de la incorporación de personal especializado en procesos de enseñanza y aprendizaje. De este modo, se podría aprovechar su conocimiento para validar los resultados y colaborar con el diseño de las adaptaciones del algoritmo de recomendación para contrarrestar los efectos de los datos faltantes.

La especificación de LOM-IEEE está pensada para compartir recursos educativos pero presenta importantes deficiencias si se desea modelar las características de los procesos educativos que se aplican en la actualidad. Se debería agregar una dimensión semántica a la especificación o bien, como se sugiere en [14], utilizando otra especificación como el IMS- RCDEO para representar los modelos de atributos y los objetivos de aprendizaje.

El tiempo de respuesta requerido para evaluar los metadatos de un OA aumenta en función del tamaño de las descripciones que contiene y la longitud de dicha descripción no es un indicador relevante para determinar la afinidad de un recurso con respecto a una actividad.

Al analizar la información asociada a la disponibilidad de los metadatos se evidenció que a pesar de los esfuerzos por mejorar su calidad, los elementos con mayor soporte desde el 2003 son los identificadores, la ubicación del recurso y el idioma. Esta información es suficiente para diferenciar los OA, pero no para realizar una recomendación de recursos para las actividades de un curso.

Se identificó que para realizar una recomendación de OA es necesario enfocarse en metadatos que describen aspectos académicos y de contenido, lo que implica utilizar metadatos con menor soporte en el repositorio pero que aportan un valor agregado a las descripciones de los recursos.

En cuanto al SQI, se encontraron diferencias importantes asociadas a la cardinalidad de una respuesta obtenida al especificar la búsqueda para un metadato específico y cuando no se especifica donde buscar. Esto toma importancia cuando se desea aplicar filtros a los OA, por ejemplo, el

idioma que puede venir dado en múltiples formatos y estar en distintas secciones de la especificación en LOM.

El proceso de reducción de los Ngramas mediante "*stripping*" no es suficiente para formar frases coherentes y relevantes dentro del idioma español, del mismo modo que la aplicación de algoritmos como el TF-IDF o similitudes de coseno de forma independiente no son indicadores reales de la relevancia práctica de los términos para un tema en particular.

Además, las ambigüedades semánticas y sintácticas del español dificultan la construcción de un analizador de expresiones que tome en consideración los tiempos verbales, sujetos, signos de puntuación y errores ortográficos, entre otras propiedades del idioma que requieren la implementación de mecanismos de desambiguación estructural del idioma.

La aplicación de mecanismos de desambiguación estructural propios de los sistemas de procesamiento de lenguaje natural, representan por sí solos un proyecto de investigación y desarrollo complejo, pero su aplicación para identificar grupos nominales puede mejorar la calidad de los descriptores utilizados para construir el perfil de curso.

Por otro lado, cualquier herramienta que desee realizar un análisis de los contenidos de los metadatos debe contemplar al menos:

- Soporte para múltiples idiomas: O bien, definir estrategia para filtrar OA de acuerdo al idioma real que contienen.
- Soporte para estándares técnicos específicos: Puesto que las descripciones pueden venir codificadas en formatos específicos que introducen ruido a los textos, por ejemplo las codificaciones para URL en UTF8.
- Tolerancia a errores de redacción: Debido a que muchos de los contenidos de los metadatos son introducidos manualmente por usuarios del repositorio.
- Tolerancia a fallos en el etiquetado de los metadatos: Puesto que muchos de los OA no cumplen con la especificación de LOM completa, de modo que se pueden combinar etiquetas sobre el campo padre de la estructura; o bien, no existir del todo.

Las especificaciones de las interfaces establecidas para comunicaciones entre los repositorios ofrecen modelos robustos, que permiten personalizar y mejorar la calidad de las respuestas esperadas ante una consulta. Estas consultas pueden ser realizadas en distintos modelos de sintaxis; sin embargo su utilidad real está limitada por la implementación de los métodos disponibles para el repositorio que se está referenciando.

A partir de las pruebas realizadas con SQI para recuperar metadatos desde distintos ROA, se encontró que los métodos relacionados con la cardinalidad de las respuestas no están disponibles en todos los repositorios. Esto representa una dificultad técnica al realizar búsquedas de recursos y recuperación de información fuera de las interfases de navegación web que ofrecen los ROA.

Se encontró que el acceso mediante servicios web produce una latencia que podría mejorarse si los algoritmos estuvieran directamente integrados al repositorio o bien, si las implementaciones del SQI y OAI-PMH estuvieran completas para reducir la cantidad de consultas necesarias para obtener una respuesta.

En la descripción general de IEEE-LOM no se incluyen rubros para identificar el mecanismo de encapsulación de los recursos digitales; de modo que las lecciones con objetivos de aprendizaje definidos y otros elementos de carácter educativo quedan reducidos a los detalles que aporte quien redacte los metadatos del OA. Además, dentro de los OA recuperados para las pruebas no se encontró ninguno que incluya descripciones asociadas a los estándares técnicos de su encapsulación.

Al ejecutar el prototipo con el DI de pruebas se identificaron 1108 posibles asociaciones hacia 701 recursos distintos que podrían ser útiles de acuerdo con los parámetros de la búsqueda definidos. Al evaluar estos recursos, se identificaron 335 elementos correspondientes a lecturas (archivos en formato .doc, .pdf y .html), 89 fueron presentaciones (archivos en formato .ppt y .pptx) y sólo 22 recursos de imágenes (archivos en formato .jpg y .tiff), de modo que al menos para la muestra obtenida, se ha mantenido vigente el patrón de disponibilidad de recursos desde el 2003, donde la mayoría de los recursos compartidos corresponden a elementos textuales y presentaciones.

4.1 Trabajo Futuro

En cuanto a los trabajos futuros, se espera trabajar en cuatro áreas principales: extracción de descriptores, diseño de algoritmos aplicados, análisis sobre los mecanismos de etiquetado automático y soporte a ontologías.

En cuanto al *proceso de extracción de descriptores*, se buscarán mejoras en el análisis de los textos con el fin de refinar la recuperación de palabras y frases con valor agregado para las búsquedas.

Con las mejoras en el *diseño de algoritmos*, se pretende realizar estudios formales comparando los resultados teórico-prácticos de los algoritmos

aplicados en esta propuesta contra otros algoritmos que se puedan aplicar a modelos vectoriales; o bien, agregar nuevos modelos de representación para los perfiles. Además, es necesario mejorar la precisión de los algoritmos utilizados actualmente en el recomendador para solventar los desfaces identificados en el proceso de recomendación y que serían valiosos para cualquier trabajo que se realice con recursos en el idioma español.

En la revisión de literatura, se encontraron referencias sobre un modelo de *etiquetado automatizado* utilizado por LACLO, pero aún no hemos encontrado referencias puntuales sobre su implementación. El análisis de este modelo utilizado para generar metadatos de forma automática puede contribuir a mejorar el proceso de desambiguación de metadatos y con esto, a mejorar la precisión de las recomendaciones.

Finalmente, con la adaptación de *soporte a ontologías* se espera brindarle al recomendador la capacidad de expandirse y adaptarse de una forma más flexible a las necesidades del usuario final. Con la implementación de este soporte, será posible modificar de forma dinámica los procesos a ejecutar y abrirán una nuevas posibilidades de escalabilidad y adaptabilidad para el sistema.

Referencias

[1] C. Garita and M. Chacón-Rivas, "TEC Digital: A case study of an e-learning environment for higher education in Costa Rica," in *Information Technology Based Higher Education and Training (ITHET), 2012 International Conference on*, Istanbul, Turkey, 2012, pp. 1–6.

[2] R. M. Felder and B. A. Soloman, "Learning styles and strategies," *URL Httpwww Engr Ncsu Edulearningstylesilsweb Html*, 2000.

[3] F. Ricci, L. Rockach, B. Shapira, and P. B. Kantor, *Recommender Systems Handbook*. Springer, 2011.

[4] "Recommender Systems," *Recommender Systems*. [Online]. Available: http://recommender-systems.org/. [Accessed: 06-Jul-2014].

[5] N. Manouselis, H. Drachsler, R. Vuorikari, H. Hummel, and R. Koper, "Recommender Systems in Technology Enhanced Learning," in *Recommender Systems Handbook*, F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds. Springer US, 2011, pp. 387–415.

[6] O. C. Santos and J. G. Boticario, "Requirements for Semantic Educational Recommender Systems in Formal E-Learning Scenarios," *Algorithms*, vol. 4, no. 2, p. 154, 2011.

[7] N. Manouselis, H. Drachsler, K. Verbert, and E. Duval, *Handbook_Recommender_Systems_for_Learning.pdf*. Springer, 2012.

[8] R. Burke, "Hybrid Recommender Systems," *Kluwer Academic Publishers Hingham, MA, USA*, vol. 12, no. 4, pp. 331–370, Nov-2002.

[9] M. Ekstrand, M. Ludwing, J. Konstan, and J. . Riedl, "Rethinking the recommender research ecosystem: reproducibility, openness, and lenskit," in *Proceeding of the fifth ACM Conference on Recommender Systems*, USA, 2011, pp. 133–140.

[10] G. I. Webb, M. J. Pazzani, and D. Billsus, "Machine Learning for User Modeling," *User Model. User-Adapt. Interact.*, vol. 11, no. 1–2, pp. 19–29, Mar. 2001

.

[11] O. C. Santos and J. G. Boticario, "Modeling recommendations for the educational domain", Procedia Computer Science, Volume 1, Issue 2, 2010, Pages 2793-2800, ISSN 1877-0509, http://dx.doi.org/10.1016/j.procs.2010.08.004

[12] IEEE Learning Technology Standardization Committee, Draft Standard for Learning Object Metadata, 18 April 2001.

[13] European Committee for standardization – Cen Workshop Agregament. A simple Query Interface Specification for Learning Repositories. Ref. No.: CWA 15454:2005 E (2005)

[14] Baldiris, S., Santos, O. C., Barrera, C., Boticario, J., Velez, J., y Fabregat, R. (2008). Integration of educational specifications and standards to support adaptive learning scenarios in adaptaplan. IJCSA, 5(1), 88–107.

[15] Odriozola, S., Luis, J., Ochoa, X., Parra Chico, G. A., y Duval, E. (2011). La experiencia de ariadne: Creando una red de reutilización de objetos de aprendizaje a través de estándares y especificaciones. IEEE-RITA, 6(3), 112–117

[16] Lemur Project, [Online]. Available: http://www.lemurproject.org/

[17] SLIM, [Online]. Available: http://glaros.dtc.umn.edu/gkhome/slim/overview

[18] Apache Mahout, [Online]. Available: http://mahout.apache.org/?

[19] LensKit, [Online]. Available: http://lenskit.grouplens.org/

[20] MyCBR, [Online]. Available: http://www.mycbr-project.net/?

[21] Duine, [Online]. Available: http://www.duineframework.org/

[22] PREA, [Online]. Available: http://prea.gatech.edu

[23] Alfaro, Agustín Francesa, Julia Espinoza Guzmán, y Mario Chacón Rivas. "Hacia una Herramienta para el Diseño Instruccional en Educación Superior."

[24] M. E. Gordillo y M. Chávez Blando, Diseño Instruccional, elemento clave en el desarrollo de cursos para Ambientes Virtuales de Aprendizaje.

[25] Pascual-Nieto, Ismael, et al. "Extending computer assisted assessment systems with natural language processing, user modeling, and recommendations based on human computer interaction and data mining."IJCAI Proceedings-International Joint Conference on Artificial Intelligence. Vol. 22. No. 3. 2011.