ISSN: 1692-7257 Volumen 1 - 2006

METIS: DIGITAL DOCUMENTS RECOMMENDATION SYSTEM BASED ON COLLABORATIVE FILTERING

METIS: SISTEMA DE RECOMENDACIÓN DE DOCUMENTOS DIGITALES BASADO EN UN MODELO COLABORATIVO

Lina Maria Osorio Hincapié, Edwing Andrés Mejía Rengifo
Jaime Alberto Guzmán Luna

Universidad Nacional de Colombia, Medellín, Colombia
{lmosori0,eamejia,jaguzman}@unalmed.edu.co

Abstract: This article shows the implementation of the Digital Document Recommendation System: METIS, which is based on the collaborative filtering concept and explicit feedback. The system allows estimating, using the k-means clustering algorithm, the qualification for the documents that have not yet been considered or qualified by a user. This estimation is based generally on the qualifications given by other users to these documents being based on the concept of the collaborative filtrate. Once these documents are qualified, the system recommends to the user the documents with the highest estimated qualifications.

Resumen: Este artículo muestra la implementación del Sistema de Recomendación de Documentos Digitales METIS, el cual se basa en el concepto de filtrado colaborativo y retroalimentación explicita para realizar su tarea. El sistema permite estimar, utilizando el algoritmo k-means, la calificación para los documentos que aún no han sido considerados o calificados por un usuario. Esta estimación se basa generalmente en las calificaciones dadas por otros usuarios a dichos documentos basándose en el concepto del filtrado colaborativo. Una vez que se pueden estimar las calificaciones para los documentos no calificados, el sistema recomienda al usuario los documentos con las calificaciones estimadas más altas.

Keywords: Recommendation systems, Collaborative filtering, Explicit feedback, *k-means* algorithm.

1. INTRODUCCIÓN

Los sistemas de recomendación pretenden simular el proceso social de recomendación, es decir, ese acto habitual que todos practicamos al recurrir a las opiniones de conocidos o expertos cuando tenemos que tomar una decisión para adquirir algo sin tener la suficiente información para ello, generalmente decisiones sencillas en el entorno de la vida cotidiana como qué libro leer, qué película ver o qué música comprar.

2. PROBLEMA

El problema de la recomendación se puede formular como sigue. (Adomavicius, 2003): Sea C el conjunto de todos los usuarios y sea S el conjunto de todos los objetos posibles que se pueden recomendar, por ejemplo libros, documentos, películas, etc. Sea u una función de utilidad que mide la satisfacción que representa el objeto s al usuario c, es decir, $u : CxS \rightarrow R$, donde R es un conjunto totalmente ordenado.

Entonces para cada usuario $c \in C$, debemos elegir un artículo $s' \in S$ que maximiza la función de utilidad del usuario. Más formalmente:

$$\forall c \in C, \qquad s'_c = \arg\max_{s \in S} u(c,s) \quad (1)$$

La utilidad de un objeto es representada generalmente por un valor, que indica cómo un usuario aprecia un objeto particular. Sin embargo, según lo indicado anteriormente, la función de utilidad general puede ser una función arbitraria.

El problema central de los sistemas de recomendación consiste en que la utilidad *u* no está definida para todo el espacio *CxS*, sino solamente en un cierto subconjunto de él. Esto significa que *u* necesita ser extrapolada a todo el espacio *CxS*. Lo anterior se debe a que la utilidad esta representada por una calificación y definida inicialmente por el conjunto de objetos calificados previamente por el usuario, generalmente aquellos objetos que el usuario ya conoce. Por lo tanto, el sistema de recomendación debe poder estimar la calificación que el usuario daría a los objetos que aún no conoce. Una vez que se estimen las calificaciones desconocidas, las recomendaciones de un objeto son hechas seleccionando el valor más alto entre todos los estimados para ese usuario, conforme a la ecuación (1).

2.1. Modelado de las necesidades de información del usuario

En esta etapa debe resolverse como modelar al usuario con sus necesidades de información, que no han de permanecer estáticas, y por tanto habrá que simular ese posible dinamismo. En general el modelo de espacio vectorial, es el preferido por su simplicidad (Montaner, 2003).

Los objetos a recomendar se representan por el conjunto $O = \{ o_1, o_2, ..., o_M \}$, por otro lado los usuarios se representan por el conjunto $U = \{ u_1, u_2, ..., u_N \}$, cada usuario u_i es un vector $u_i = (v_{i1}, v_{i2}, ..., v_{iN})$, donde v_{ij} representa la nota o calificación con la que el usuario u_i ha dado al objeto o_j. Hay que tener presente que el usuario no tiene porque haber calificado todos los objetos del sistema, de no ser así no tendría caso que el sistema hiciera recomendaciones, pues el usuario ya conocería todos los objetos pertenecientes a nuestra base de datos. La tarea del Sistema de Recomendación es precisamente estimar la calificación que un usuario particular daría a sus elementos vacíos.

2.2 Mecanismo de Filtrado de la Información

Un sistema de recomendación toma información como entrada y entrega como salida recomendaciones, a menudo en forma de estimaciones, teniendo en cuenta para esto los preferencias manifestadas por el usuario a través de los usos anteriores del sistema (Viappiani, 2002). Para obtener dichas preferencias el sistema puede valerse de alguno de los mecanismos existentes: pregunta y respuesta, filtrado basado en contenido, filtrado colaborativo, filtrado demográfico.

Pregunta y respuesta. Bajo este esquema, se le pide al usuario que llene un cuestionario que contiene un juego de preguntas, las repuestas dadas por el usuario son guardadas y tomadas en cuenta para generar su perfil. Este método presenta el inconveniente de que el sistema no posee la forma de realimentar el perfil creado, a menos que el usuario actualice sus datos cosa que podría resultar tediosa pues generalmente son cuestionarios muy largos; es posible además que se inserten datos falsos o muy ambiguos y además que el cuestionario no sea suficiente para generar un perfil adecuado.

Filtrado basado en contenido. En este esquema se escogen los objetos basándose en las características de sus contenidos. El sistema busca por objetos similares a aquellos que el usuario prefiere basado en una comparación del contenido. Esta propuesta tiene algunas desventajas principalmente en la captura de diferentes aspectos del contenido, por ejemplo, música, videos e imágenes. Pero, también para el caso de textos, las representaciones que se hacen del documento, capturan únicamente ciertos aspectos del contenido, que resultan en un pobre desempeño de los sistemas. Además de los problemas de representación, estos sistemas tienden a aprender de forma tal que ellos recomiendan objetos similares a los objetos ya vistos en usos anteriores del sistema (García, 2002).

Filtrado colaborativo. En este enfoque el sistema busca usuarios con intereses similares a los de un usuario dado y le recomienda los objetos preferidos por dichos usuarios. En lugar de calcular la semejanza entre los objetos, el sistema

calcula la semejanza entre los usuarios. En la visión colaborativa, no hay análisis del contenido de los objetos. A cada objeto se le asigna un único identificador y una calificación definida por el usuario. La semejanza entre los usuarios está basada en la comparación de las calificaciones que los usuarios dan a los mismos objetos (García, 2002).

Para cualquier objeto de la base de datos, el sistema debe recolectar información de diferentes usuarios para ser capaz de recomendarlo. Usuarios semejantes no son relacionados mientras no hayan calificado un número suficiente de objetos similares. Uno de los inconvenientes que presenta este enfoque es que para un usuario con gustos inusuales comparado con el resto de usuarios el sistema tendrá un desempeño muy pobre, pues no sabrá en donde agruparlo (García, 2002).

Filtrado demográfico. El filtrado demográfico agrupa a los usuarios en clases dependiendo de sus características demográficas y se sume que los usuarios pertenecientes a un grupo tienen preferencias similares.

El filtrado demográfico generaliza los intereses de los usuarios, así que el sistema recomienda los mismos objetos a usuarios con perfil demográfico similar, además no posee mecanismos para adaptarse a los cambios en las preferencias individuales de los usuarios. Sin embargo el filtrado demográfico puede ser muy poderosos combinado con otra técnica (Montaner, 2003).

2.3 Mecanismo de retroalimentación

Retroalimentación implícita. Con esta técnica se obtiene información acerca del usuario de forma discreta; observando su interacción natural con el sistema. Algunos de los comportamientos que han sido investigados más extensivamente son: el tiempo de lectura, la selección de texto, el guardado y la impresión de información. La principal ventaja de usar la retroalimentación implícita es que esta evita al usuario el tener que retroalimentar al sistema.

Retroalimentación explicita. La acción explícita del usuario es el método más familiar y más directo de obtener la retroalimentación para construir un modelo relevante del usuario. Bajo este esquema el sistema anima a los usuarios a calificar los objetos recuperados por el, para mejorar la calidad de los resultados futuros.

El problema de este acercamiento explícito es de sentido práctico: los usuarios no realizarán simplemente tal retroalimentación explícita constantemente y con frecuencia. El incitar persistentemente para que califiquen los resultados es molesto, y los usuarios en un cierto plazo reducirán o eliminarán el uso de tales mecanismos de retroalimentación. Esto reduce la cantidad y la exactitud de la retroalimentación obtenida, que alternadamente reduce la calidad de los resultados obtenidos.

Como se ha dicho hasta ahora un sistema de recomendación debe estar provisto de al menos un mecanismo de filtrado y al menos un mecanismo de retroalimentación, para poder generar los perfiles con los cuales generará las estimaciones de calificación de los objetos desconocidos por un usuario, para generar dichos perfiles se deben agrupar a los usuarios según sus afinidades en las evaluaciones dadas a los objetos, para lograr esto se requiere de un algoritmo de agrupamiento que, en nuestro caso utilizamos el algoritmo de agrupamiento k_means, que se describe a continuación.

2.4 Agrupamiento

Una técnica de agrupamiento se puede definir como una técnica diseñada para realizar una clasificación asignando patrones a grupos de tal forma que cada grupo sea más o menos homogéneo y distinto de los demás. El criterio de homogeneidad más simple está basado en la distancia: se espera que la distancia entre los patrones de un mismo agrupamiento sea significativamente menor que la distancia entre patrones de agrupamientos diferentes.

Se dispone de un conjunto de vectores $\{x_1,...,x_p\}$, que representan a los objetos y a partir de él se desea obtener el conjunto de grupos $\{1,...,n\}$ que los engloban. El problema es que a priori no se sabe cómo se distribuyen los vectores en las clases, ni siquiera cuántas clases habrá. A partir del conjunto de vectores de características dado se trata de realizar agrupaciones de estos vectores en clases, de acuerdo con las similitudes encontradas (Moreiro, 2002).

Todos los grupos poseen un centroide, que es un objeto representante de los objetos del grupo o cluster al cual este pertenece. La similitud a los objetos del cluster respecto a su centroide se mide por una función de similitud, por ejemplo la

distancia euclidiana, correlación vectorial, entre otros.

Agrupamiento Particional. En el agrupamiento particional el objetivo es obtener una partición de los objetos en grupos o clusters de tal forma que todos los objetos pertenezcan a alguno de los *k* clusters posibles y que por otra parte los clusters sean disjuntos.

Si denotamos por $\{X_1, X_2,..., X_M\}$ al conjunto de *M* patrones, se desea buscar *k* grupos de patrones o clusters, $C_1, C_2,..., C_K$ de tal forma que:

$$C_1 \bigcup C_2 \bigcup \bigcup C_k = \{X_1, X_2, \ldots, X_M\} \quad (2)$$

$$C_i \bigcap C_j = 0 \quad \forall_i \neq j \quad (3)$$

Paso 1: *Considerar k elementos como k clusters con un único elemento*
Paso 2: *Asignar cada vector del espacio al centroide más cercano.* *Después de cada asignación se recalculará el nuevo centroide.*
Paso 3: *Después de que todos los objetos hayan sido asignados en el paso anterior, calcular los centroides de los clusters obtenidos, y reasignar cada objeto al centroide más cercano*
Paso 4: *Repetir los pasos 2 y 3 hasta que se alcance un determinado criterio de parada*

Fig. 1. Algoritmo k-medias, (McQueen, 1967)

Método de k-medias de McQueen. El método que McQueen propuso en el año 1967 es conocido como k-medias y es el método de agrupamiento particional más utilizado. En la Fig. 1 se relacionan los principales pasos de este algoritmo.

3. SISTEMA DE RECOMENDACIÓN METIS

El sistema desarrollado recomienda artículos en formato electrónico. La aplicación permite al usuario buscar, leer, descargar y evaluar artículos en formato electrónico, los documentos se encuentran divididos por temas de interés. El prototipo combina las técnicas de filtrado colaborativo y mecanismo de aprendizaje explicito para realizar las recomendaciones, además utiliza procedimientos que se encuentran almacenados en la base de datos, estos procedimientos se encargan de generar las clases o clusters de usuarios, utilizando el algoritmo de agrupamiento k-means entre otros.

Los procedimientos almacenados en la base de datos se encargan de seleccionar aquellos documentos que han sido evaluados un mínimo de veces, este valor es un parámetro configurable por el administrador del sistema; una vez seleccionados aquellos documentos que cumplen esta condición se procede a seleccionar los usuarios que han evaluado al menos una vez, alguno de estos documentos; cuando se han escogido los usuarios se genera el modelo vectorial explicado anteriormente, en este espacio vectorial cada vector de usuario posee las evaluaciones dadas por este a los documentos que fueron seleccionados; una vez se tiene el espacio vectorial, se ejecuta el algoritmo k-means propiamente, el cual termina su ejecución ya sea por que alcanza un número máximo de iteraciones (parámetro que se puede configurar) o encuentra que la diferencia entre los clusters hallados en dos pasos del ciclo es mínima. Dada la cantidad de pasos y cálculos que se realizan para obtener los clusters, llevar a cabo este procedimiento en línea provocaría un tiempo de respuesta del sistema muy alto, esto ocasionaría descontento en los usuarios por lo tanto estos procesos son llevados a cabo fuera de línea.

En las fases de desarrollo del sistema se utilizo la metodología planteada por la Escuela de Sistemas de la Universidad Nacional Sede Medellín, que incluye los elementos necesarios para obtener un producto en operación y de alta calidad, a través de la integración del método CDM (Custom Development Method) de Oracle Corporation y algunos modelos del método RUP (Rational Unified Process).

3.1 Fase de Definición

El objetivo de la fase de definición es analizar el problema y la solución que propone el sistema que se pretende desarrollar. Está compuesta entre otros por la identificación de los actores principales del sistema, el diagrama jerárquico de funciones.

Agentes o Unidades Organizacionales Involucradas.

Usuario: Es aquel que puede registrarse en el sistema, buscar documentos, leer documentos, evaluar documentos y solicitar recomendaciones de artículos que no ha leído entre otros.

Administrador: Es el encargado de ingresar y clasificar los artículos por materia, administrar los permisos de los usuarios y algunos parámetros internos del sistema que influyen en su desempeño.

Diagrama Jerárquico de Funciones. Muestra las funciones principales que ofrece el sistema de recomendación METIS. El sistema inicia en la interfaz principal o de Inicio donde los usuarios o el administrador ingresan a la interfaz de bienvenida correspondiente y escoge sus entre las opciones listadas, pero si se trata de un usuario nuevo podrá listar los documentos digitales por tema e inspeccionar si la información contenida en el sistema es de su interés, si es así, es necesario llenar un formulario de inscripción e inscribirse.

Los usuarios registrados pueden evaluar los documentos digitales, listar los documentos evaluados, solicitar pronósticos y recomendaciones, pueden además actualizar los parámetros de usuario, actualizar sus datos o cancelar definitivamente su registro en el sistema.

El administrador por otra parte podrá realizar las funciones de administración del sistema, tales como, administración de usuarios, de ocumentos digitales, procesos internos, variables y temas. (Ver Fig. 2).

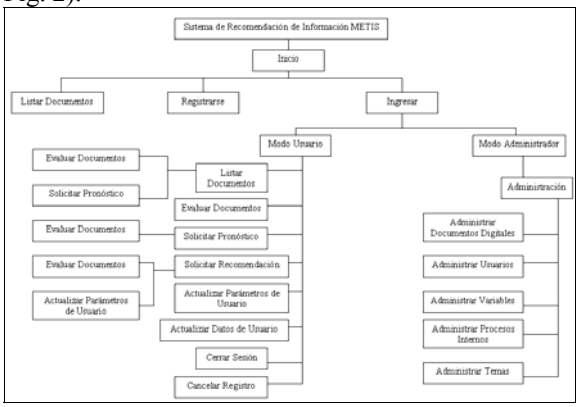

Fig. 2. Diagrama Jerárquico de Funciones

3.2 Fase de Análisis

El objetivo de la fase de análisis es entender el sistema y su estructura. La fase de análisis está compuesta entre otros por el modelo de datos y los modelos funcionales.

Modelo de Datos. El modelo de datos representa la estructura de los objetos del sistema, sobre los cuales se debe guardar y mantener información, y las relaciones entre ellos. Ver Fig. 3.

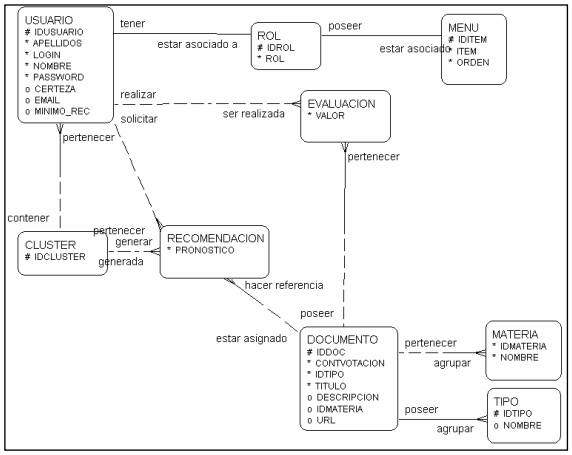

Fig. 3. Diagrama Entidad – Relación

Modelos Funcionales

Las funciones que ofrece el sistema se representan a través de los diagramas de casos de uso, los cuales juegan un papel muy importante en el desarrollo de productos de software, ya que, aunque sólo describen la forma en que un sistema funciona y no el cómo, sirven de guía para el desarrollo de las interfaces de usuario. Ver Fig. 4.

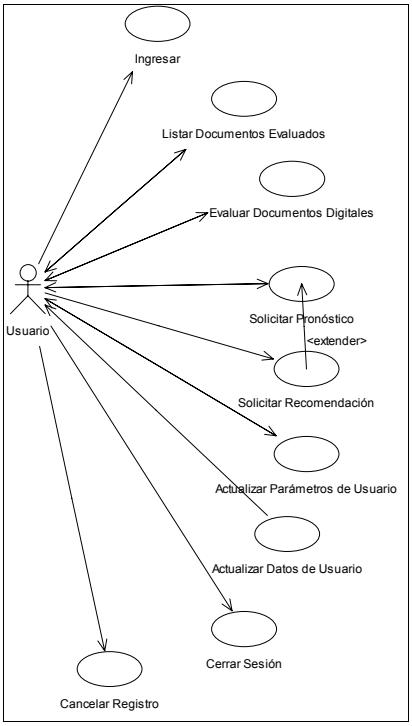

Fig. 4. Casos de Usos para el Administrador

3.3 Implementación

Para la implementación del prototipo se opto por una arquitectura multinivel, propia de aplicaciones Web.

Esta arquitectura permite operar bajo cualquier ambiente y desde cualquier lugar, con la utilización de un navegador de Internet, y con la implementación de páginas Web dinámicas.

Además se ha hecho uso del patrón de diseño Modelo Vista Control (MVC), en el cual el flujo de la aplicación está dirigido por un controlador central. El controlador delega solicitudes - en nuestro caso, solicitudes HTTP - a un manejador apropiado. Los manejadores están unidos a un modelo, y cada manejador actúa como un adaptador entre la solicitud y el modelo. El modelo representa, o encapsula, un estado o lógica de negocio de la aplicación. Luego el control normalmente es devuelto a través del controlador hacia la vista apropiada. El reenvío puede determinarse consultando los conjuntos de mapeos, normalmente cargados desde una base de datos o un archivo de configuración. Esto proporciona un acoplamiento cercano entre la vista y el modelo, que puede hacer las aplicaciones significativamente más fáciles de crear y de mantener la Fig. 5 ilustra esta arquitectura.

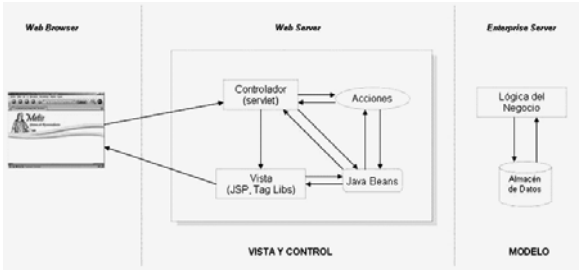

Fig. 5. Arquitectura Técnica

4. CONCLUSIONES

En el presente artículo se ha descrito el proceso llevado a cabo en la implementación del sistema de recomendación de documentos electrónicos METIS. Este sistema permite a los usuarios acceder a documentos en formato electrónico, y solicitar recomendaciones sobre aquellos que el aún no ha leído accediendo de una manera mas efectiva a la información que necesita dependiendo del tema de interés. Por el momento se tienen temas relacionados con la ingeniería de sistemas, pero eventualmente se podría ampliar la gama de temas. Es importante resaltar que un sistema de recomendación es independiente de los objetos que este recomienda pues la información utilizada en sus estimaciones son las calificaciones dadas a los objetos de interés, así que en un futuro el sistema podría recomendar objetos diferentes a documentos digitales.

REFERENCIAS

Adomavicius, G., Tuzhilin, A. (2003), "Recommendation Technologies: Survey of Current Methods and Possible Extensions", pp.2-3.

García, J., Pérez, J., Arenas, A. (2002), "Aplicación de una Metodología de Desarrollo de Sistemas Multiagente en la Diseminación Selectiva de Información en la Web".

McQueen, J.B. (1967)," Some methods for classification and analysis of multivariate observations", Fifth Berkeley Symposium on Mathematical Statistics and Probability 1, pp. 281-297

Montaner, M. (2003), "Collaborative Recommender Agents Based On Case-Based Reasoning and Trust", pp. 7-50.

Moreiro, J. (2002), "Aplicaciones Al Análisis Automático Del Contenido Provenientes De La Teoría Matemática de la Información", pp. 282.