SARA DONNELLY GARCES AGREDO

META BUSCADOR WEB BASADO EN LA INFORMACIÓN DEL CONTEXTO Y EL FILTRADO COLABORATIVO

Universidad del Cauca
Facultad de Ingeniería Electrónica y Telecomunicaciones
Maestría en Computación
Popayán
2012

SARA DONNELLY GARCES AGREDO

META BUSCADOR WEB BASADO EN LA INFORMACIÓN DEL CONTEXTO Y EL FILTRADO COLABORATIVO

Tesis presentada a la Facultad de Ingeniería Electrónica y Telecomunicaciones de la Universidad del Cauca para la obtención del Título de Magíster en COMPUTACIÓN

Director
CARLOS ALBERTO COBOS LOZADA, MSc. Ph.D (c)

Popayán
2012

Agradecimientos

A Dios por todas las bendiciones recibidas y por hacer de cada uno de mis sueños una realidad.

A mi director, Ph.D (c) MSc. Carlos Alberto Cobos Lozada por compartir sus conocimientos conmigo y por brindarme su apoyo incondicional durante todo mi proceso de formación.

A mi familia y amigos por ser un soporte en los momentos difíciles.

A mis compañeros y docentes por la oportunidad de conocerlos y aprender de ellos.

Resumen

Este proyecto de investigación se enfoca en el campo de recuperación de información en la Web. Presenta un modelo de meta buscador Web que integra el filtrado colaborativo (basado en ítems) a la propuesta [1] de Massimo Melucci fundamentada en proyectores sobre planos que se originan en la información del contexto del usuario.

El modelo obtenido fue implementado en una aplicación Web, denominada MyBestMetaWebSearch, que usa una arquitectura multi-capa basada en Servicios Web XML que permite re-ordenar y filtrar los resultados entregados por los buscadores tradicionales Google, y Bing como fuente inicial de la búsqueda. Esta aplicación Web contempla los siguientes pasos generales: (1) Registrarse e ingresar al sistema, (2) ingresar la consulta basado en palabras clave, (3) pre-procesar la consulta, (4) expandir la consulta, (5) recuperar los documentos de los buscadores tradicionales, (6) filtrar la Información, (7) visualizar y calificar resultados y (8) Modificar información del contexto e información de feedback para la comunidad. Finalmente se presenta el proceso de evaluación del modelo propuesto con medidas clásicas del área de la recuperación de la información, satisfacción del usuario y relevancia, para lo cual se usó en primera instancia una colección cerrada de textos denominada CACM, posteriormente se calculó la Curva de Precision-Recuerdo, Mean Average Precision (MAP), Precisión en K resultados ordenados y el estadístico Kappa, y se compararon los resultados con los entregados originalmente por los buscadores tradicionales donde se muestra que en algunas ocasiones son mejores que los entregados por los buscadores Web tradicionales más usados hoy en día, Google y Bing, aunque dicha mejora no es significativa.

Palabras Clave. Recuperación de información, Contexto del usuario, filtrado colaborativo, expansión de consulta, meta buscador Web

Abstract

This research project focuses on the field of information retrieval on the Web. A model of a meta search Web engine that integrates collaborative filtering (based on items) is proposed. It builds on the Massimo Melucci's proposal [1] based on projectors on planes that originate from the user context information in order to provide more relevant results to users.

The obtained model was implemented in a Web application called MyBestMetaWebSearch, which uses a multi-layered architecture based on XML Web Services that allows re-sort and filter the results delivered by traditional search engines Google, Yahoo! and Bing as an initial source searching. This Web application provides the following general steps: 1) Registering and logging in, 2) entering the query based on keywords, 3) pre-processing the query, (4) query expansion, (5) retrieving documents traditional search engines, (6) filtering the information, (7) visualizing and describing results and (8) Modifying background information and feedback information to the community. Finally, the evaluation of the proposed model with classical measures in the area of information retrieval, user satisfaction and relevance were presented. For this, a closed collection of texts called CACM was used primarily. Then, the Precision-recall curve, Mean Average Precision (MAP), precision at K and Kappa statistic were calculated. The results were compared with those originally delivered by traditional search engines (Google, Yahoo! and/or Bing) which shows that results reported by the model are slightly more relevant than those reported by the compared systems.

KEYWORDS: Information retrieval, user's context, collaborative filtering, query expansion.

Tabla de Contenido

Lista de Tablas

Lista de Figuras

Parte I – Introducción

La búsqueda de información utilizando buscadores Web, es una actividad cada vez más cotidiana a cualquier ser humano, dado que gran parte de la información se encuentra disponible en Internet [1-3], en volúmenes tan grandes que se miden en términos de millones de petabytes (10^{15} bytes) o exabytes (10^{18} bytes) [4].

Cuando un usuario interactúa con un Sistema de Recuperación de Información (SRI), como en un buscador Web, lo hace expresando características de la información que desea encontrar mediante una consulta. Dicha consulta se expresa por medio de términos de búsqueda, un conjunto de palabras clave o también conocidos como criterios de búsqueda [2]. Los buscadores Web tradicionales (Google, Yahoo!, Bing entre otros) usan diferentes enfoques para procesar la consulta del usuario y le presenta los documentos recuperados en forma de lista. Los resultados entregados por el buscador Web pueden presentar resultados que no son relevantes a las necesidades de información del usuario, y el orden en el cual se presentan los resultados en la lista en muchos casos no es el más adecuado [5]. Con el fin de afrontar estos dos inconvenientes se proponen mejoras a los buscadores, usando por ejemplo conceptos en lugar de términos, mejores esquemas para la definición de las consultas, mejores esquemas de representación de los documentos y de las características de búsqueda de cada usuario que usa el sistema, y el concepto de meta búsqueda [6].

1. Planteamiento del Problema

Día a día la búsqueda Web adquiere más importancia para la gran mayoría de los usuarios de Internet. Buscadores como Google, Yahoo!, Bing y Ask cada vez son más populares [2-4] y de gran utilidad. Desafortunadamente, estos buscadores “clásicos” o “tradicionales” muestran resultados que en muchas ocasiones no corresponden con los que el usuario necesita realmente, lo anterior afecta a los

usuarios de Internet, tal como se corrobora en el estudio realizado por Dogpile.com en colaboración con investigadores del estado de Pensilvania y de Queensland en 2007, donde muestran que los resultados de las búsquedas de los cuatro principales motores de búsqueda (Google, Yahoo!, MSN Search y Ask) son en general distintos o únicos en un alto porcentaje (88,3%) de las veces, compartiendo además un bajo porcentaje de los resultados entre unos y otros.

Los problemas con la búsqueda Web pueden ser divididos en dos clases: de datos y de usuario. Entre los problemas relacionados con los datos están: el volumen, la velocidad de los cambios, la naturaleza contradictoria de la Web (spamming de metadatos, contenido y enlaces), la diversidad de lenguajes y de contenidos, el esquema de cooperación de servidores Web para los buscadores, la recuperación de datos multimedia, entre otros. Desde el lado del usuario se necesitan mejores lenguajes de consulta, interfaces de usuario y visualización de resultados. Además un problema de gran importancia está relacionado con la evaluación de los resultados que se le presentan a los usuarios [5].

Aunque existen muchas circunstancias que motivan la insatisfacción del usuario por la baja precisión de los resultados entregados por los buscadores tradicionales [5, 6], en este proyecto se trabajaron explícitamente dos estrategias: la gestión del contexto del usuario y la retroalimentación que el usuario puede registrar explícitamente al sistema de recuperación de información (o búsqueda Web) con el fin de mejorar la relevancia de los resultados recuperados.

A pesar de que se han realizado varias propuestas de gestión del perfil del usuario, en la presente investigación se tomó como base la reciente propuesta de Massimo [1], quien propone un modelo de manejo de la información del contexto, tomando cada propiedad o característica del contexto como una base no orto normal de un espacio vectorial que luego es usado para establecer una función probabilística denominada la “probabilidad de la relevancia”, con la que se reordena la presentación de los resultados al usuario. Una ventaja, es que dicho modelo es

general, independiente del medio y aplicable a varias tareas. Adicionalmente, usa múltiples fuentes de evidencia presentes en una descripción de contexto (por ejemplo, tiempo de visualización, retención de documentos), funciones de rastreo, matrices de densidad (que incorporan información acerca de la ocurrencia de algunos factores contextuales en términos de preguntas cuyas respuestas están sujetas a medidas de probabilidad) y los proyectores (proyección de cualquier punto *x* del espacio vectorial a un punto del subespacio imagen de la transformación), para mejorar las estructuras de información de feedback implícito personalizada para cada usuario y para cada tarea de búsqueda.

Además, la retroalimentación (feedback) del usuario es planteado en la literatura de recuperación de información como una de las estrategias más populares de reformulación de la consulta del usuario [7]. El proceso de retroalimentación permite que un usuario tome provecho de la información propia (perfil del usuario) así como de información de otros usuarios (comunidad o grupo de usuarios con un perfil similar), en lo que se conoce como técnicas de filtrado colaborativo. De este modo los usuarios pueden obtener mejores resultados futuros con un menor esfuerzo. Esta retroalimentación se ha usado normalmente para expandir las consultas en el modelo vectorial y darle mayor peso a los términos dispuestos en un modelo probabilístico.

Teniendo en cuenta que no se encontró en la literatura recopilada una propuesta que integre el contexto del usuario propuesto por Massimo con técnicas de filtrado colaborativo, en este proyecto resultó de interés tomar como pregunta investigación la siguiente: ¿Es posible encontrar resultados más relevantes a las necesidades de los usuarios a través de la creación de un meta buscador Web, basado en la propuesta de Melucci, que use la información del contexto (personalización) y técnicas de filtrado colaborativo?

2. Justificación

La importancia de la presente tesis de maestría se centró en la generación de un conocimiento nuevo y útil para la comunidad científica y académica internacional de recuperación de información, al proponer la combinación de conceptos relacionados con la información del contexto, basados en la propuesta de Melucci [1], con técnicas relacionadas con el filtrado colaborativo (comúnmente usadas en sistemas de recomendación) para la expansión de la consulta, el filtrado y el re-ordenamiento de los resultados que se le presentan al usuario, donde se intenta obtener mejoras sobre los resultados que arrojan los motores de búsqueda "tradicionales".

Desde una perspectiva práctica, este proyecto fue conveniente ya que buscó disminuir el tiempo invertido por las personas en los procesos de recuperación de información al evitarles la lectura y revisión de temas no relacionados con sus consultas. En la práctica el modelo propuesto muestra promisorios resultados en pruebas realizadas sobre colecciones cerradas y con usuarios.

Durante el desarrollo del proyecto se pusieron en práctica los conocimientos adquiridos en las asignaturas de la maestría en computación, así como también se debieron apropiar nuevos conocimientos y desarrollar nuevas habilidades de investigación en el área específica de recuperación de información.

Como producto final de este proyecto se definió un modelo de un meta buscador Web que contempla los siguientes pasos generales de una tarea de búsqueda en la Web: 1) Registrarse e ingresar al sistema, 2) ingresar la consulta basado en palabras clave, 3) pre-procesar la consulta, (4) expandir la consulta, (5) recuperar los documentos de los buscadores tradicionales, (6) filtrar la Información, (7) visualizar y calificar resultados y (8) Modificar la información del contexto e información de feedback para la comunidad. Los mayores aportes de la presente tesis se realizaron en los pasos 4, 6 y 8.

En el paso 4, la expansión de la consulta, con el modelo se hace de manera automática y sin que el usuario sea consciente, un proceso de expansión de consulta basado en la información de contexto disponible del usuario y de la comunidad. En el paso 6, filtrado y re-ordenamiento de los resultados: tomando como base la propuesta de Melucci (información de contexto representada en un espacio vectorial no orto normal), se toma provecho de la información de contexto del usuario y se complementa con la información de la comunidad (basada en técnicas de filtrado colaborativo) para filtrar y re-ordenar la información que se muestra al usuario en una lista ordenada (rankeada) de documentos tal y como lo presentan los buscadores tradicionales. Y en el paso 8, se realiza la modificación de la información del contexto e información de feedback para el usuario y la comunidad, lo anterior basado en la calificación que el usuario realiza de los resultados obtenidos, el sistema realiza la gestión de la información de contexto del usuario y de la comunidad.

3. Objetivos

A continuación se describen los objetivos del proyecto, conforme fueron aprobados por el Comité de Investigaciones de la Facultad de Ingeniería Electrónica y Telecomunicaciones en el documento de anteproyecto.

3.1 Objetivo General

Definir y evaluar un meta buscador web basado en la propuesta de Massimo Melucci [1], que integre la información del contexto[1] y técnicas de filtrado colaborativo, intentando mejorar la relevancia de los resultados originalmente reportados por los buscadores Web tradicionales, tales como Google, Yahoo! y/o Bing.

[1] Ejemplos de propiedades contextuales son el perfil del usuario, las búsquedas anteriores, clusters o grupos de documentos previamente estudiados, fechas, lugares y calendarios.

3.2 Objetivos Específicos

- Establecer un modelo[2] de un meta buscador Web, enmarcado en el modelo vectorial de recuperación de información, basado en la propuesta de Massimo Melucci [1], vinculando técnicas de filtrado colaborativo para presentar resultados más relevantes a las necesidades de los usuarios.

- Desarrollar una aplicación Web basada en el modelo del meta buscador propuesto, con una arquitectura multi-capa basada en Servicios Web XML, que permita re-ordenar y filtrar los resultados entregados por los buscador tradicionales (Google, Yahoo! y/o Bing)[3] como fuente inicial de la búsqueda.

- Evaluar el modelo propuesto con medidas clásicas del área de la recuperación de la información, satisfacción del usuario y relevancia, a través de la Curva de Precision-Recuerdo, Mean Average Precision (MAP), Precisión en K resultados ordenados y el estadístico Kappa, comparando los resultados con los entregados originalmente por los buscadores tradicionales (Google, Yahoo! y/o Bing).

4. Metodología de Desarrollo del Meta buscador

Para el desarrollo de la presente investigación se tuvieron en cuenta tres etapas principales, cada una con el objetivo de lograr un producto específico, a saber: el modelo del meta buscador Web, la aplicación Web basada en el modelo del meta buscador propuesto y la evaluación del modelo. Para cada etapa se utilizó una metodología distinta, debido a la heterogeneidad de los productos a obtener.

[2] El modelo será representado a través de diagramas de casos de uso, diagramas de clases y diagramas de persistencia, además del pseudocódigo y la explicación de los algoritmos involucrados en la expansión de la consulta, y el filtrado y ordenamiento de los resultados, lo que incluirá una representación de los datos en un espacio vectorial n-dimensional.

[3] El modelo no contemplará los procesos de recolección automática de información en Internet (crawling), indexado, ni almacenamiento. Se parte del trabajo que en este caso ya hacen los buscadores tradicionales.

En la primera etapa, para definir el modelo del meta buscador Web se realizaron las siguientes actividades: el estudio detallado de los proyectos relacionados con la presente investigación, en cuanto a información de contexto del usuario, expansión de consulta, filtrado colaborativo y evaluación en recuperación de información; el modelo global del meta buscador Web; el modelo detallado de cada componente y los análisis básicos de complejidad y rendimiento de los algoritmos involucrados con el objetivo de asegurar que sean viables en el contexto de la investigación; el primer prototipo del meta buscador para hacer pruebas alfa e ir refinando el modelo.

En la segunda etapa concerniente a construir la aplicación Web basada en el modelo del meta buscador propuesto, se realizó con una instanciación del Proceso Unificado [8] el cual consta de las siguientes fases:

- **Inicio:** se realizó un modelo de casos de uso simplificado con los casos de uso más críticos del sistema. Luego se obtuvo una arquitectura provisional que describía los subsistemas más importantes.

- **Elaboración:** se especificaron detalladamente los casos de uso y se diseñó la arquitectura del sistema. Se usaron los siguientes artefactos: diagrama de casos de uso, diagrama de clases y modelo entidad relación de la base de datos, además del pseudocódigo y la explicación de los algoritmos involucrados en la expansión de la consulta, y el filtrado y ordenamiento de los resultados.

- **Construcción**: en esta fase se creó el meta buscador Web, se desarrollaron un total de seis (7) iteraciones, con una duración a de dos (2) semanas cada iteración, con las siguientes metas: *(i)* Arquitectura base del meta buscador *(ii)* Gestión de usuarios *(iii)* Incorporación de filtrado colaborativo *(iv)* Gestión del Feedback del usuario *(v)* expansión de la consulta *(vi)* Estadísticas, registro de datos para el análisis de pruebas *(vii)* Pruebas alfa y ajustes generales a la aplicación web.

- **Transición**: esta fase cubrió el periodo de transformación del producto en versión beta.

Adicionalmente, se realizó la pasantía investigativa con el Grupo de I+D en Sistemas y Tecnologías de la Información (STI) de la Universidad Industrial de Santander, en la ciudad de Bucaramanga, cuyos resultados principales fueron: la creación de la segunda versión del meta buscador y la elaboración del articulo titulado “Modelo de Búsqueda Web Basado en Información del Contexto del Usuario y Técnicas de Filtrado Colaborativo”

En la tercera etapa se realizó la evaluación del modelo y del meta buscador Web donde se utilizaron las siguientes medidas: Curva de Precision-Recuerdo, Mean Average Precision (MAP), Precisión en K resultados ordenados y el estadístico Kappa. La evaluación se realizó con cuatro experimentos usando una colección cerrada de documentos y once pruebas con estudiantes del Programa de Ingeniería de Sistemas de la Universidad del Cauca (con un mínimo de 14 personas por prueba).

Finalmente, cabe destacar que en forma transversal se realizó el proceso de documentar los resultados que se fueron obteniendo parcialmente, se efectuó una constante recolección y análisis bibliográfico (para mantener actualizado el estado del arte), y se hizo la sistematización del proyecto en cuanto a su proceso de desarrollo y los productos desarrollados.

5. Resultados Obtenidos

A continuación se listan los resultados/productos obtenidos con el desarrollo de la presente tesis de maestría:

- Artículo: “Modelo de Búsqueda Web Basado en Información del Contexto del Usuario y Técnicas de Filtrado Colaborativo”. Ver anexo A.

- Prototipo software del meta buscador Web basado en la información del contexto y el filtrado colaborativo disponible en http://www.mybestmetawebsearch.com.

- Monografía del trabajo de grado. Corresponde al presente documento, donde se describe el proceso seguido en el desarrollo del proyecto, los aportes más significativos, las conclusiones y recomendaciones para el desarrollo de futuras investigaciones.

6. Organización del Documento

A continuación en la Parte II se describen los conceptos teóricos que fueron soporte en el desarrollo de la investigación, entre ellos, se encuentran la propuesta [1] de Massimo Melucci, la recuperación de la información y el filtrado colaborativo, temas en los que se fundamenta el proyecto, además conceptos básicos del proceso de evaluación de sistemas de recuperación de información. En la parte III se presenta el modelo de meta buscador Web propuesto donde se describen detalladamente cada uno de sus componentes principales. Posteriormente, en la parte IV se describe el meta buscador haciendo énfasis en el proceso de desarrollo usado para la construcción del mismo, donde se muestran los aspectos más relevantes del análisis, diseño e implementación de la aplicación Web obtenida. En la parte V se encuentra la evaluación del modelo propuesto para verificar la precisión en los resultados, seguido, en la parte VI se presentan las conclusiones a las que se llega con el desarrollo de la investigación y las recomendaciones para trabajos futuros. Finalmente en la parte VII se presenta el glosario con una lista de los términos empleados en la investigación y una corta explicación de los mismos. Además se presenta la lista de referencias bibliográficas que soportan formalmente la investigación.

Parte II – Contexto Teórico

La facilidad que tiene cualquier usuario para publicar información en Internet, hace que ésta además de crecer exponencialmente, lo haga de una manera desordenada, haciendo potencialmente difícil el proceso de consultar información útil frente a unos requerimientos específicos de usuario [9]. Los modelos de presentación de resultados usados hoy en día suelen utilizar criterios como la similitud para organizar los documentos con base en los requerimientos que expresa el usuario en la consulta [9, 10] sin embargo la relevancia de los documentos recuperados no satisface necesariamente las necesidades del usuario. Este hecho ha llevado a considerar la posibilidad de usar información contextual del usuario o de múltiples usuarios (en enfoques colaborativos), para desarrollar SRIs que brinden un mayor grado de satisfacción al usuario durante el proceso de recuperación de información [11, 12]

7. Recuperación de la Información (RI)

La recuperación de información es un área interdisciplinaria de estudio que busca las mejores formas de representar, almacenar, organizar y acceder ítems de información en forma automática [7], donde los ítems de información se consideran documentos (normalmente no estructurados) que están relacionados con las solicitudes de búsqueda de un usuario [13]. La recuperación de información se caracteriza porque ofrece al usuario la posibilidad de realizar búsquedas sobre grandes cantidades de documentos teniendo en cuenta: concordancias parciales o las mejores concordancias frente a una solicitud de información, un mecanismo de inferencia basado en la inducción, un modelo de búsqueda probabilístico, la posibilidad de clasificar los documentos en múltiples temas, el uso de un lenguaje de consulta similar al natural implicando condiciones de consulta que son incompletas, un

despliegue de documentos ordenados por relevancia y con una alta posibilidad de equivocarse en el orden de presentación de dichos documentos [7, 14].

Los temas centrales de investigación en recuperación de información se iniciaron con la definición de mecanismos eficientes de almacenamiento (índices, índices ponderados, índices invertidos, índices probabilísticos, clasificación automática de palabras claves, discriminación y representación), clasificación automática, estructuras de archivos, estrategias de búsqueda (funciones de concordancia, búsqueda serial, agrupamiento representativo, retroalimentación, re-consultas, búsqueda probabilística) y evaluación (rendimiento, satisfacción) del sistema en una colección "controlada" de documentos [7, 13, 14] . Pero con el tiempo, y específicamente el cambio que ha generado Internet en la vida de todas las personas, la recuperación de información Web o búsqueda Web (uno de los servicios más esenciales de este ambiente [2, 3, 13] se ha tenido que tomar aportes conceptuales y metodológicos de mayor cantidad de áreas de conocimiento. En este sentido, la estadística y probabilidad, la inteligencia artificial, el reconocimiento de patrones, el procesamiento paralelo y otras áreas han incorporado muchas otras técnicas "no tradicionales" de recuperación de información, entre ellas redes bayesianas, lógica difusa, algoritmos genéticos, procesamiento de lenguaje natural, algoritmos concurrentes, almacenamiento distribuido; además, el estudio de datos multimediales, el manejo de múltiples idiomas, la navegación y la visualización de los datos ha tomado mucha mayor importancia [7, 13, 15]

7.1 Modelos Clásicos en la Recuperación de Información

En la actualidad existen varios modelos de recuperación de información (RI), siendo los más destacados [7, 13, 14] el modelo booleano, el vectorial y el probabilístico. A continuación se muestra la definición formal de un modelo de RI.

Definición: Un modelo de recuperación de información es una cuádrupla de la forma; $[D,Q,F,R(q_{i,}d_j)]$ [7, 13, 16] donde: D es la representación del conjunto de documentos, Q es la representación del conjunto de la información necesitada por el

usuario, también llamadas consultas, F es el marco de trabajo que modela las representaciones de los documentos, las consultas y sus relaciones y $R(q_i, d_j)$ es la función que permite establecer el orden de presentación de los resultados (vínculos a documentos d_j donde $d_j \in D$) con respecto a una consulta $q_i \in Q$ [13]

7.1.1 Modelo Booleano

El modelo booleano es un modelo de recuperación basado en teoría de conjuntos y algebra booleana. Se basa en la utilización de operadores lógicos como AND, OR y NOT para la creación de consultas [13]. Los documentos se encuentran representados por conjuntos de palabras o términos clave, donde se les asocia un peso binario, uno (1) si el término aparece en el documento por lo menos una vez y cero (0) si el término no aparece. Las consultas se materializan en expresiones booleanas, que tienen una semántica precisa. Cada término de la consulta se identifica con el conjunto de los documentos que contienen dicho término. Después se crean las intersecciones de conjuntos y se seleccionan aquellas que cumplen con las condiciones de la consulta. El grado de similitud entre la consulta del usuario y un determinado documento también es binario, uno (1) si el documento es relevante para la consulta y cero (0) sino es relevante [7, 13].

Para comprender mejor el modelo se presenta un ejemplo ilustrativo de su funcionamiento. Se supone que un usuario desea realizar una consulta que debe contener el término t_1 y además el término t_2 o la negación del término t_3, es decir que t_3 no esté presente: $q = t_1 \wedge (t_2 \vee \neg t_3)$. Cada uno de los operadores puede ser representado utilizando el diagrama de Venn (ver Figura 1).

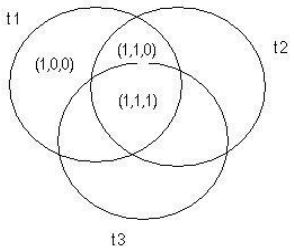


Figura 1. Diagrama de Venn (Tomado de [7])

El modelo booleano representa un documento d_j en forma de tupla (t_1, t_2, t_3), donde por ejemplo la tupla $(1,1,0)$ representa los documentos que contienen los términos t_1 y t_2, además todos los documentos pertenecientes a este conjunto de datos son relevantes para la consulta, en comparación con el conjunto de documentos pertenecientes a la tupla $(0,1,0)$, que no son relevantes para la consulta.

La ventaja del modelo booleano es que es simple de comprender, pero su principal desventaja radica en que el criterio de recuperación no es óptimo, al considerar un documento relevante solamente cuando contiene exactamente los términos que se consultan, además no hay un grado de relevancia específico, ya que un documento es relevante a la consulta o no lo es. Las desventajas hacen que el modelo booleano sea considerado un modelo de recuperación de datos en lugar de un modelo de recuperación de información.

7.1.2 Modelo Espacio Vectorial

También conocido como modelo vectorial, está basado en tres principios: La equiparación parcial, la ponderación de los términos en los documentos y la ponderación de los términos en la consulta; adicionalmente, tanto un documento como una consulta se representan mediante conjuntos ordenados de números (no solamente de ceros y unos, sin embargo en este modelo también el cero representa la ausencia del término en el documento). Gracias a esta representación, tanto los documentos como las consultas pueden tratarse matemáticamente como vectores en un espacio n dimensional, donde n es el número de términos en el conjunto de documentos. En el modelo vectorial basta fijar un criterio de similitud para poder ordenar por relevancia los documentos de una colección en relación a una consulta, una de las maneras más habituales de cuantificar el nivel de cercanía entre vectores es mediante el coseno del ángulo que forman [7, 13]. El modelo vectorial ha sido usado en operaciones de recuperación de información, filtrado de información, categorización automática de información, entre otros. La descripción formal para un documento puede expresarse como se muestra a continuación en la (Ecuación 1) [7]:

$$D_i \rightarrow \vec{d} = (T_{i1}, T_{i2}, .., T_{in})$$

Ecuación 1. Descripción formal de un documento en el modelo vectorial

Donde D_i es el documento i-ésimo, de un conjunto N de documentos, con un conjunto de m características consideradas como ocurrencias de palabras o términos T_{ik} en dicho documento. Se debe tener en cuenta que existen palabras como artículos, preposiciones, pronombres, entre otros, que no tienen unas buenas capacidades de selección o búsqueda, estas palabras son conocidas como palabras vacías, y que se reconocen porque tienen una frecuencia de aparición demasiado alta. Dichas palabras normalmente son eliminadas de la representación vectorial del documento, a menos que la representación del documento sea en texto completo.

En el modelo se construye una matriz, donde las columnas representan los términos y las filas los documentos, expresados en función de la frecuencia de aparición de cada término (vectores en términos algebraicos). De esta manera, un conjunto de N documentos se almacenaría en una matriz de n filas por N columnas, siendo n el total de términos almacenamos en dicho conjunto de documentos. Por ejemplo: un documento puede ser expresado como d1 = (1, 2, 0, 0, 0,.., 1, 3), donde cada uno de los valores es el número de veces que aparece cada término en el documento. La longitud del vector es igual al total de términos de la matriz.

Teniendo en cuenta el ejemplo anterior es posible que una palabra aparezca más de una vez en un mismo documento, pero para la consulta del usuario dicho término no tenga suficiente relevancia, por lo tanto el simple conteo (frecuencia observada) de términos de un documento no es suficiente garantía para clasificar la relevancia de cada documento frente a una consulta. Por otro lado, también puede ser necesario que algunas palabras deban ser consideradas con más peso o más significativas que otras. De acuerdo con lo anterior se ha visto la necesidad de dar un peso a cada uno de los componentes del vector, teniendo en cuenta que se debe evitar privilegiar

documentos muy extensos frente a otros menos extensos, para lograrlo se deben normalizar los vectores de los documentos a través de la Ecuación 2:

$$\vec{d}_i = \frac{1}{\sqrt{\sum_{j=1}^{n} w_{ij}^2}} (w_{i1}, w_{i2}, ..., w_{in})$$

Ecuación 2. Normalización de los vectores de los documentos

La ecuación anterior multiplica cada uno de los elementos del vector, por el inverso de su norma, lo que garantiza un porcentaje de acuerdo a la cantidad de apariciones de un determinado término dentro del conjunto de todas las apariciones de los términos para un documento específico.

Se debe tener en cuenta que el cálculo del peso de cada término, en el vector documento, se estima basado en el siguiente análisis: un término es importante para la relevancia de un documento si posee una frecuencia alta de aparición en el mismo, pero si su frecuencia de aparición es alta en muchos otros documentos de la colección, no es beneficioso para distinguir un documento de los demás. Para resolver esta situación primero se calcula la frecuencia del término en el documento (tf), luego se consulta la frecuencia del término en toda la colección de documentos; si la frecuencia es muy elevada, es posible que pertenezca al conjunto de palabras vacías, como solución se opta por eliminarlo del conjunto de términos de la colección. Así, es posible afirmar que la importancia de un término es inversamente proporcional a su frecuencia en la colección de documentos, a esto se lo conoce con el nombre de frecuencia inversa del documento (idf). Con estas bases se calcula el peso de cada elemento del vector documento teniendo en cuenta su frecuencia inversa en la colección y su frecuencia dentro de cada documento [7, 13] mediante la Ecuación 3.

$$w_{ij} = tf_i \cdot idf_j$$

Ecuación 3. Peso de cada elemento del vector documento

Aunque se han desarrollado varias formulas para calcular los pesos de los términos en los documentos, una de las más usadas es la Ecuación 4:

$$w_{ij} = tf_{ij} \cdot \log \frac{N}{df_j}$$

Ecuación 4. Peso de cada elemento del vector documento

Donde N es el número de documentos de la colección y df_j es la cantidad de documentos donde aparece el término j.

De igual forma el proceso elaborado para los documentos se aplica a la consulta realizada por el usuario, transformando la consulta en un vector m-dimensional, para luego ser procesada buscando entregar los documentos más relevantes a la consulta. Dicha relevancia está basada en una medida de similitud, es decir un documento en particular es relevante para la consulta del usuario si su representación vectorial es similar a la representación vectorial de la consulta realizada. Se dispone de varias fórmulas para hallar la medida de similitud, sin embargo aquellas que representan la similitud de coseno son las más usadas [7, 13] (Ecuación 5 y la Ecuación 6).

$$cos\,\theta = sim(dj, q) = \frac{\overrightarrow{dj}.\vec{q}}{\|dj\|\|\vec{q}\|}$$

Ecuación 5. Similitud de cosenos

$$cos\,\theta = sim(dj, q) = \frac{\sum_{i=1}^{t} w_{i,j} * w_{i,q}}{\sqrt{\sum_{i=1}^{n} w_{i,j}^2}\sqrt{\sum_{i=1}^{n} w_{i,q}^2}}$$

Ecuación 6. Similitud de cosenos

La medida de similitud entre un documento d_j y una consulta q es el coseno del ángulo entre los dos vectores que los representan en el espacio vectorial. Si la similitud entre los vectores da como resultado cero, significa que no hay coincidencia

alguna entre los componentes de los vectores, ya que el producto escalar será cero. Si la similitud entre los vectores es 1, se está presentando la similitud máxima que sólo se da cuando todos los componentes de los vectores son iguales, en este caso la función del coseno obtiene su máximo valor, la unidad.

7.1.3 Modelo Probabilístico

El modelo probabilístico fue introducido en 1976 por Robertson y Jones. Este modelo se tomó en cuenta desde principios de la década de los 70's por su capacidad de llevar los problemas de RI a un terreno formal bien fundamentado [13]. Dentro de los modelos probabilísticos el más simple es conocido como el modelo de independencia binaria (BIM), cuyas asunciones son: a) los documentos se presentan de forma booleana (esto es, un vector de unos y ceros indicando la presencia/ausencia de un término), b) la ocurrencia de dos términos distintos es estadísticamente independiente entre sí, c) los términos que no se encuentran en la consulta no afectan los resultados, y d) la relevancia entre documentos es estadísticamente independiente. Sin embargo, estas simplificaciones son cuestionables porque existen casos particulares donde fallan.

En el modelo BIM, se asume una noción binaria de relevancia tal que un documento sólo puede pertenecer a uno de los grupos, relevantes o no relevantes. Bajo esta asunción se puede definir una variable aleatoria usando la Ecuación 7:

$$R(d,q) = \begin{cases} 1\ si\ d\ es\ relevante\ para\ la\ consulta\ q \\ 0\ si\ de\ no\ es\ relevante\ para\ la\ consulta\ q \end{cases}$$

Ecuación 7. Definición de una variable aleatoria en el modelo BIM

Es decir $R(d,q)$, en adelante simplemente R, es una variable aleatoria bidiscreta binaria, la cual no es nula sólo para los documentos relevantes para la consulta.
Dada una necesidad de información puntual, la propuesta del modelo es presentar los documentos en orden decreciente de probabilidad de relevancia

$P(R = 1|d, q)$ [13]. Dada una consulta q, el modelo asigna a cada documento d_j una medida de similitud dada por la Ecuación 8 [7]

$$Odds(d_j\ relevante\ para\ q) = \frac{P(d_j\ relevante\ para\ q)}{P(d_j\ no\ relevante\ para\ q)}$$

Ecuación 8. Medida de similitud en el modelo BIM

El modelo probabilístico define la similitud del documento d_j con la consulta q de acuerdo a la Ecuación 9:

$$Similitud\ (d_j, q) = \frac{P(R|\vec{d}_j)}{P(\bar{R}|\vec{d}_j)}$$

Ecuación 9. Medida de similitud en el modelo probabilístico

Donde R es el conjunto de documentos inicialmente relevantes y $\overline{R}$ su complemento, $P(R|\vec{d}_j)$ es la probabilidad de que un documento d_j sea relevante para la consulta q, y $P(\bar{R}|\vec{d}_j)$ es la probabilidad de que un documento d_j no sea relevante para la consulta q.

Cabe destacar que existen otros modelos del BIM que producen fórmulas de similitud del mismo tipo que las del modelo vectorial [13]. La principal ventaja del modelo probabilístico es que se basa en un marco teórico firme e incorpora una función de relevancia continua inherente al modelo. Por otro lado, las desventajas del modelo son: la necesidad de conjeturar la separación inicial entre documentos relevantes y no relevantes, y para el caso de utilizar BIM, asunciones como la independencia entre términos pueden ser poco realistas.

En la práctica ocurre que algunos modelos comienzan siendo vectoriales y luego migran al probabilístico efectuando algunas variaciones en las fórmulas de similitud [13].

8. Evaluación en Recuperación de Información

La recuperación de la información se ha desarrollado como una disciplina empírica que requiere una evaluación cuidadosa y exhaustiva para demostrar el desempeño de las nuevas técnicas. El proceso de evaluación relacionado con esta propuesta, implica colecciones controladas y no controladas de documentos que se presentan en listas rankeadas (ordenadas) de resultados [7, 13].

En RI las medidas más clásicas de evaluación son la precisión (número de documentos relevantes recuperados sobre el total de documentos recuperados), el recuerdo (número de documentos relevantes recuperados sobre el total de documentos relevantes) y la medida-F (media armónica de la precisión y el recuerdo).

La comunidad de RI ha propuesto como una medida de comparación de los resultados de diferentes sistemas de recuperación, la ***curva de precisión-recuerdo***, una gráfica dentada que muestra los valores de precisión y recuerdo en los mejores k resultados entregados por el sistema (ver Figura 2) [7, 13]. Esta curva normalmente involucra el cálculo de una precisión interpolada a ciertos niveles de recuerdo r, está definida como la máxima precisión encontrada a cualquier nivel r' ≥ r, que se define como $P_{interpolada}(r) = \max_{r' \geq r} p(r)$.

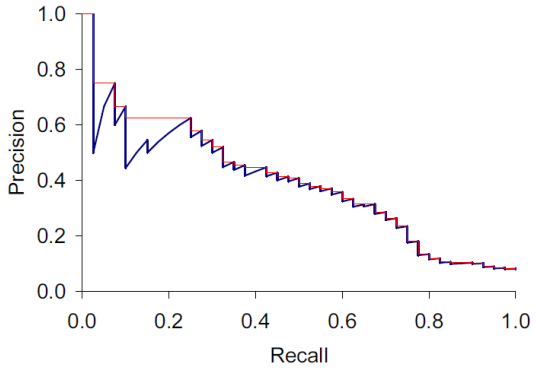


Figura 2 Curva de Precisión-Recuerdo

En recientes años, la comunidad TREC (Text Retrieval Conference) [17] ha propuesto la ***Mean Average Precision (MAP)***, como una medida sencilla que provee

una medida de la calidad de la precisión a través de diferentes niveles de recuerdo. La precisión promedio (average precision) es el promedio de los valores obtenidos en la precisión para el conjunto de los primeros k documentos existentes después de que cada documento relevante es recuperado, y este valor es entonces promediado sobre las necesidades de información de los usuarios. Es decir, Si el conjunto de documentos relevantes para alguna necesidad de información $q_j \in Q$ es $\{d_1, d_2, \ldots d_{m_j}\}$ y R_{jk} es el conjunto de los resultados de la recuperación ordenados desde el resultado más importante hasta el documento d_k, entonces:

$$MAP(Q) = \frac{1}{|Q|} \sum_{j=1}^{|Q|} \frac{1}{m_j} \sum_{k=1}^{m_j} \Pr ecision(R_{jk})$$

Ecuación 10. Mean Average Precision (MAP)

Cuando un documento relevante no es recuperado, el valor de la precisión en la ecuación anterior toma cero (0). Para una sola necesidad de información, la precisión promedio se aproxima al valor interpolado del área bajo la curva de la precisión-recuerdo. Por lo tanto, MAP es aproximadamente la media del área bajo la curva de la curva de precisión-recuerdo para un conjunto de preguntas.

Desafortunadamente estas medidas no se pueden aplicar en colecciones no controladas y no tiene en cuenta que los resultados son presentados en listas ordenadas. Por lo anterior se hace necesario contar con medidas que sirvan en este contexto [7, 13] además, contar con medidas que sean más afines con las necesidades de recuperación de los usuarios en Internet. A ellos, lo que más les importa es qué tan buenos son los resultados en las primeras páginas (ya que normalmente ellos no revisan toda la lista de resultados). Esto motiva a que la evaluación involucre una medición de los valores de precisión en ciertos puntos de los resultados recuperados, como por ejemplo los primeros 10 o 20 documentos. Esta medida se conoce como la "***precisión en K***" (Precision at K). Esta medida tiene la ventaja de no requerir ninguna estimación del conjunto total de resultados relevantes, pero tiene la desventaja de ser menos estable.

Finalmente, es interesante considerar y medir qué tan de acuerdo están los usuarios (jueces del sistema) sobre los juicios de relevancia. En las ciencias sociales, una medida común para el acuerdo entre los jueces es el estadístico ***kappa***, que está diseñada para juicios categóricos conforme se presenta en la Ecuación 11.

$$kappa = \frac{P(A) - P(E)}{1 - P(E)}$$

Ecuación 11. Kappa

Donde P(A) es la proporción de las veces que los jueces están de acuerdo, y P(E) es la proporción de las veces que se espera llegar a un acuerdo por casualidad (azar). Para calcular este último hay varias opciones, si simplemente se está tomando una decisión de dos clases y no se asume algo más, entonces la tasa de acuerdo de oportunidad esperada es de 0,5. Sin embargo, normalmente la distribución de clases asignado está sesgada, y lo habitual es utilizar las estadísticas marginales para calcular el acuerdo esperado. Teniendo en cuenta que en muchos casos (como en la presente tesis) se tienen más de un juez se hace necesario adaptar esta fórmula de Kappa. Fleiss propone la evaluación para cualquier número de observadores que dan grados categóricos, a un número fijo de artículos o artefactos de juicio. Este nuevo indicador Kappa de Fleiss, permite interpretar los resultados según el grado de concordancia entre los observadores con una medida que puede tomar valores entre -1 y +1.

9. Personalización en Recuperación de Información

La personalización de la recuperación de información, es un enfoque en el que se usan los perfiles de los usuarios, adicionalmente a la consulta, para estimar los intereses de los usuarios y seleccionar el conjunto de documentos relevantes [18]. En este proceso, la consulta describe la búsqueda actual del usuario, que es el interés local [19], mientras el perfil del usuario describe las preferencias del usuario sobre un largo periodo de tiempo (interés global). Dependiendo de la forma como los

intereses globales afectan los locales, las operaciones de consulta se clasifican dentro de dos operaciones: expansión de consulta y reponderación de términos [7]. Un sistema puede tener una combinación de las dos técnicas, cambiando los pesos de los términos (teniendo en cuenta, inclusive la retroalimentación que hacen los usuarios en los resultados de consultas previas) y adicionando nuevos términos a la consulta (expansión de la consulta). Con esto es claro que, la personalización en recuperación de información pretende mejorar la experiencia del usuario incorporando la subjetividad del usuario al proceso de recuperación.

La exploración de los intereses del usuario implícitos y sus preferencias han sido identificadas como una dirección importante para superar el estancamiento potencial de las actuales tecnologías de recuperación. Con respecto al tema, los primeros trabajos en el campo de modelado de usuario y sistemas adaptativos se remontan a finales de los años 70 [20, 21]. Las técnicas de personalización ganaron significado en los años 90 con el aumento de redes de computadores de gran escala los cuales habilitaron servicios masivos, heterogéneos y con consumidores finales que día a día son menos previsibles [22]. De hecho, desde un principio se ha producido trabajo significativo en términos de logros académicos y productos comerciales [23-25]

Cabe aclarar que, el objetivo de la personalización es crear sistemas software con la capacidad adaptar algunos aspectos de su funcionalidad y/o apariencia en tiempo de ejecución a las particularidades de un usuario, para un mejor ajuste a sus necesidades. Para ello, el sistema debe tener una representación interna (modelo) del usuario. Algunos aspectos de software que han sido sujetos a personalización incluyen: filtrado de contenidos [26], secuenciación [27], representación de contenidos [28], recomendación [29], búsqueda [30, 31], interfaces de usuario [32-34], tareas de secuenciación [35], o ayuda en línea [36]. Además, algunos dominios de aplicación típica para el modelado de usuario y sistemas adaptativos incluyen: educación [27, 28, 35 , 37], comercio electrónico [38-41], noticias [29, 42, 43], bibliotecas digitales [44, 45], herencia cultural [46], turismo [23], etc.

Otra aplicación importante de ésta área son los meta buscadores personalizados, que re-direccionan una consulta de entrada a uno o más buscadores externos, ejecutando una mezcla o agregación de cada lista de resultados de búsqueda. Los términos asociados al perfil del usuario pueden ser adicionados a las consultas originales y enviados a cada motor de búsqueda [47], normalmente los términos necesitan reponderación para acceder a las funciones internas de los buscadores. Otra opción, es brindada por la relevancia por feedback [21, 48] que es un caso particular donde la reformulación de la consulta toma lugar. Esta técnica toma el juicio de relevancia explicito del usuario, quien decide cuales documentos retornados por la consulta son o no relevantes. Cuando se usa personalización, en lugar de extraer la información relevante de las interacciones con el usuario, el sistema de búsqueda usa la representación del perfil del usuario.

Cuando se hace uso de personalización basada en links, ésta afecta directamente a las técnicas de ranking de documentos. Una ventaja de este enfoque es que el sistema no tiene que tomar en cuenta el contexto del documento, sólo los hyperlinks presentes en alguna página Web. En general los algoritmos de personalización basados en links son modificaciones del PageRank de Google [30, 49] o el Authority HITS y el Algoritmo Hub [11, 50]. Existen diferentes maneras para introducir búsquedas personalizadas en algoritmos Page Rank, por ejemplo, Page rank sensitivo a tópicos y documentos relevantes [30, 50]. Las alteraciones de los algoritmos Page Rank personalizados son en su mayoría fáciles de desarrollar, pero hay aún un desequilibrio relacionado con la escalabilidad, ya que calcular estos valores requiere altos recursos computacionales, y actualmente es imposible calcular un valor Page Rank personal completo para cada usuario. Algunas soluciones planteadas han sido, el cálculo de sólo un pequeño conjunto de valores para un pequeño conjunto de tópicos [49, 51], o algoritmos más eficientes donde vectores Page Rank parciales son calculados, permitiendo la combinación de estos para un vector personalizado final [30]. La importancia radica en que un motor de búsqueda personalizado puede calcular un valor ranking personalizado para cada documento en el conjunto de resultados. El beneficio de dicha propuesta es que este valor sólo

tiene que ser calculado para el resultado obtenido del conjunto de documentos. La desventaja es que este valor tiene que ser calculado en tiempo de consulta. Dicho algoritmo también está disponible para meta buscadores [52], siendo capaz de calcular una puntuación personalizada en tiempo real, al entrar en el contenido de los documentos o usando sólo los resúmenes provistos.

La reorganización de los resultados es un factor importante en procesos de recuperación de información personalizada, los n primeros documentos relevantes a la consulta (top n), se reorganizan de acuerdo con la relevancia de estos para el perfil del usuario, [12]. En [53] usan una red Bayesiana jerárquica, utilizando feedback implícito y explícito del usuario, el contenido se reorganiza de acuerdo con el modelo del usuario representado en la red Gausiana jerárquica. La principal ventaja es que los modelos de otros usuarios se pueden usar para resolver el problema de cold start (arranque sin información), donde el sistema no tiene ninguna información acerca de un nuevo usuario para el sistema. En cuanto a documentos ponderados usando tópicos, en [54, 55] se pueden encontrar ejemplos, donde los perfiles de usuario son representados como un conjunto de tópicos taxonómicos, expresados explícitamente por el usuario. El conjunto resultante es finalmente reorganizado usando una medida de distancia entre los tópicos asociados a los documentos y los tópicos en el perfil del usuario. Los resultados de la consulta pueden ser agrupados en un conjunto de categorías (clusters), presentando primero las categorías más relevantes para el usuario [56-58]. El algoritmo toma 1) el conjunto del resultado general para la consulta, 2) obtiene el conjunto de categorías relacionadas para los documentos en el conjunto resultante, 3) re organiza el conjunto de categorías de acuerdo con el perfil de usuario.

El sistema presentado en [59] representa los intereses del usuario en términos de relaciones y valores (ej. Películas de amor, películas de un director x), los resultados se clasifican en términos de propiedades y son ordenados de acuerdo a la relevancia para el usuario. En [60] se presenta un meta buscador con técnicas de clustering, los resultados son agrupados jerárquicamente por el título y resumen. El usuario es

entonces capaz de filtrar los resultados al mostrar el interés para uno de los clusters. Las técnicas de Soporte de Navegación le sugieren al usuario un conjunto de links que están más relacionados con sus preferencias. La relevancia de cada link en el documento se calcula de acuerdo con la relevancia del documento señalado para el usuario. La relevancia de estos documentos enlazados también podría ser tomada en cuenta, teniendo un algoritmo iterativo similar al Web Crawler personalizado local. También se pueden usar otros parámetros, como la relevancia para el usuario de los documentos accedidos previamente en la ruta que finaliza en el documento actual [56].

La cantidad de motores de búsqueda con capacidad de personalización ha aumentado, desde motores de búsqueda sociales, donde los usuarios pueden sugerir colaborativamente cuales son los mejores resultados para una consulta dada [61], hasta motores de búsqueda verticales [47, 62], donde los usuarios pueden personalizar un motor de búsqueda de dominio específico, hay un gran interés por parte de compañías comerciales de motores de búsqueda como: Yahoo!, Microsoft o Google, pero es la última la que ha mostrado verdaderas habilidades de personalización.

En personalización Google ha desarrollado algunas iniciativas, Google Personal por ejemplo, plantea la búsqueda personalizada de Google [63] (actualmente descontinuada), basada en temas de categorías Web (del proyecto de directorio abierto, DMOZ en http://www.dmoz.org) y seleccionado manualmente por el usuario. Google Co-op [62] que permite la creación de motores de búsqueda compartidos y personalizados en el sentido en que los usuarios sean capaces de etiquetar páginas Web y filtrar resultados con este nuevo metadato. Finalmente, iGoogle [64], que hace énfasis en las capacidades de personalización, donde el usuario puede agregar a su página noticias, fotos, predicciones del tiempo y entre otros.

Eurekster aunque está más orientado a los “grupos de búsqueda”, es un motor de búsqueda que incluye la habilidad para construir explícitamente el perfil de usuario por medio de los términos, documentos y dominios [47].

K-bus es un motor de búsqueda creado por la compañía Entopia Knowledge Bus [65] que fue seleccionado como la mejor tecnología de búsqueda en el 2003, provee recuperación de información altamente personalizada. Para clasificar las respuestas, el motor tiene en cuenta el nivel de habilidad de los autores de los contenidos retornados por la búsqueda, y la habilidad de los usuarios que enviaron la consulta. Estos niveles de habilidad se calculan teniendo en cuenta interacciones previas de diferentes clases entre el autor y el usuario sobre algunos contenidos.

MyYahoo Las características del motor de búsqueda personal de Yahoo [66] son todavía muy sencillas. Los usuarios pueden rechazar la URL cuando aparece en los resultados o guardar las páginas para una Web personal que otorga una mayor prioridad en esas páginas una vez que aparezcan en un conjunto de búsqueda de resultados.

Por otro lado, uno de los factores y desarrollos clave hacia la creación de soluciones personalizadas tiene que ver con el contexto, la definición y el tratamiento del contexto, que varía significativamente dependiendo de la aplicación de estudio [67]. En recuperación de la información el contexto tiene un amplio significado, que va desde los elementos circundantes en una aplicación de recuperación XML [68], los últimos elementos seleccionados o compras en los sistemas de información proactiva [38, 69], documentos a los que se ha accedido recientemente [70], páginas Web visitadas [71], consultas anteriores y datos obtenidos a través de clicks [71-74], textos circundando una consulta [75, 76] textos resaltados por un usuario [75], etc. Los sistemas sensibles al contexto (context-aware) pueden ser clasificados por 1) el concepto que el sistema tiene para contexto, 2) como se adquiere el contexto, 3) como se representa la información del contexto 4) como se usa la representación del contexto para adaptarse al sistema. Una solución simple para la adquisición del

contexto es la aplicación de técnicas de retroalimentación explícita, como retroalimentación relevante [48, 77]. La retroalimentación relevante crea una representación de contexto por medio de una interacción explicita con el usuario.

10. Bases para Recuperar Información en Contexto

En [1] Melucci propone un modelo de manejo de la información del contexto, tomando cada propiedad o característica del contexto como una base no ortonormal de un espacio vectorial que luego es usado para establecer una función probabilística denominada “probabilidad de relevancia o función de ranking”, con la que se reordena la presentación de los resultados al usuario. Este modelo es general, independiente del medio y aplicable a varias tareas. Adicionalmente, usa múltiples fuentes de evidencia presentes en una descripción de contexto (las propiedades de contexto se utilizan para denotar una de las formas en que el contexto opera sobre la materialización de objetos de información, por ejemplo, tiempo de visualización, retención de documentos, el espacio, el contenido y el tipo de documento), funciones de rastreo, matrices de densidad (que incorporan información acerca de la ocurrencia de algunos factores contextuales en términos de preguntas cuyas respuestas están sujetas a medidas de probabilidad) y los proyectores (proyección de cualquier punto x del espacio vectorial a un punto del subespacio imagen de la transformación), para mejorar las estructuras de información de feedback implícito personalizado para cada usuario y para cada tarea de búsqueda.

El término factor contextual se usa para significar uno de los posibles valores de una propiedad contextual, por ejemplo, "introducción" y "matemáticas” son los posibles valores de la propiedad "tipo de documento". Si bien los factores de una propiedad contextual son mutuamente excluyentes, los factores de diferentes propiedades no. La idea consiste en que cada factor puede ser instanciado como un vector de un espacio, es decir, como una n-tupla de números.

Una característica importante de este modelo es que todo se describe como un subespacio de un espacio vectorial complejo o como una combinación lineal de subespacios: Un rayo (semirecta generada por un vector) es un ejemplo de subespacio de este tipo. Esta característica da a entender que los objetos se describen como rayos, planos, o combinaciones de ellos. Además, los factores contextuales, también se describen como rayos, planos, o combinaciones de ellos.

En [78] Melucci también reporta la idea de modelar el contexto usando bases del espacio vectorial. La premisa básica es que un vector base modela un documento o una consulta, que la semántica del documento o la consulta depende del contexto y que un cambio en el contexto puede ser modelado por una transformación lineal de una base vectorial. En otras palabras, cada documento o consulta está asociado a una base vectorial distinta y la conexión de una base a otra está gobernada por una transformación lineal.

En un escenario donde un usuario está accediendo a un SRI con el fin de recuperar los documentos relevantes para su necesidad de información, el contexto influencia no sólo la selección de las palabras clave de la consulta, sino su semántica y la forma en la que ésta se relaciona a otras palabras. El contexto es expresado por la instancia específica del vector base, el cual corresponde a las palabras clave empleadas para expresar dicha necesidad de información.

Formalmente, este modelo se expresa de la siguiente manera: Sea d un vector de un documento que puede ser generado por una base P_B y sea q un vector que representa una consulta (q no necesariamente se genera sobre la misma base) que puede ser generado por una base P_Q. Por lo tanto d está representado por $d = P_B.a$; mientras que $q = P_Q.b$ donde a y b son coeficientes usados para combinar los vectores base de P_B y P_Q respectivamente. Si la relevancia es estimada por el producto interno usual, los documentos son ordenados (rankeados) por $d^T.q = (P_B.a)^T.\left(P_Q.b\right) = a^T.\left({P_B}^T.P_Q\right).b.$

Lo anterior revela que las relaciones entre las palabras claves utilizadas para expresar los documentos y las consultas depende de dos contextos: el que haya participado en la creación de los documentos y el utilizado en la formulación de la consulta.

A continuación se presenta un ejemplo numérico (Ver Figura 3). Suponga que un documento d está representado por el vector d y que se genera por una base P_B cuyos vectores base son t_1, t_2, que no son ortogonales sino independientes uno del otro. Los coeficientes a_1, a_2 combinan linealmente a t_1, t_2. Los cuales están definidos por: $P_B = \begin{bmatrix} 4 & 2 \\ 2 & 4 \end{bmatrix}$, $a = \begin{bmatrix} \frac{1}{3}, & \frac{1}{3} \end{bmatrix}^T$, entonces $d = P_B.a = [2, \quad 2]^T$. De manera similar la consulta q se representa por el vector q, pero se genera por la base de P_Q y los coeficientes b_1, b_2 así: $P_Q = \begin{bmatrix} -2 & -2 \\ 4 & -2 \end{bmatrix}$, $b = \begin{bmatrix} \frac{2}{4}, & 2 \end{bmatrix}^T$, entonces $q = P_Q.b = [-5, \quad -2]^T$. Si otra consulta q' se genera por P_B pero con los mismos coeficientes de q, entonces $q' = P_B.b = [6, \quad 9]^T$. Los vectores de las consultas se generan por los mismos coeficientes, pero q' está más cerca a d, ya que se generó utilizando su misma base, mientras que q se generó por P_Q, que es "distante" de P_B. Este ejemplo explica cómo el conocimiento del contexto puede ser crucial para obtener un "correcto" ordenamiento de los documentos.

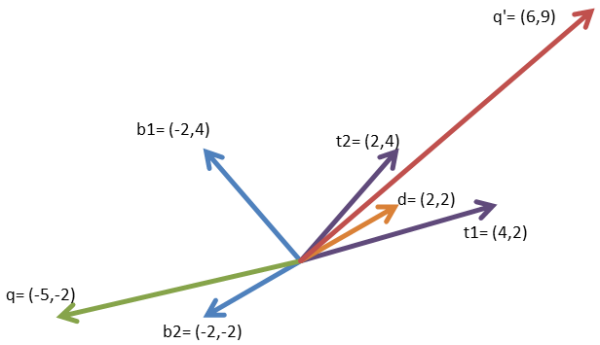


Figura 3. Un ejemplo del modelado del contexto

Por otro lado, como las palabras claves son representadas como vectores base, diferentes contextos deberían reflejarse sobre bases diferentes. Hay más de una base para una colección de documentos. Similarmente no existe una única base para

todas las consultas, hay tantas bases como contextos, aún para cada consulta única. El cambio de contexto puede ser modelado como una transformación de matrices [78]. Así, un cambio del contexto P_B a un nuevo contexto P_Q se logra transformando la correspondiente matriz P_B a la matriz P_Q.

Preposición 1: Dadas dos matrices P_B y P_Q, existe una única matriz P_C tal que $P_Q = P_B.P_C$ [79], en otras palabras se necesita encontrar la matriz adecuada para calcular el cambio de contexto.

El cálculo de los elementos que forman los vectores base permite emplear el contexto en indexación y recuperación. De hecho, la clasificación de documentos cambia radicalmente si el contexto está representado por una base en lugar de otra, ya que diferentes vectores base pueden conducir a diferentes matrices de correlación.

Preposición 2. Dada una matriz simétrica S con vectores columna independientes, existe una matriz no singular P_B tal que $S = P_B^T.P_B$ y P_B es única [79].

Para estimar S de un documento un par de enfoques están disponibles: (i) el uso de ontologías, que representan el dominio en términos de clases y jerarquías de especialización. (ii) Anotación de documentos, que pueden ser automáticamente realizados asociando los términos de los documentos a clases [80]. Por lo tanto, S_{ij} se puede calcular usando por ejemplo, la frecuencia de correlación dentro de clases. Alternativamente, los documentos de texto se pueden segmentar dentro de ventanas de texto, y los términos correlacionados dentro de ellos se pueden calcular usando medidas de correlación tradicionales tales como coeficiente del coseno o el coeficiente de Dice [13]. De esta forma, el elemento S_{ij} es la medida de correlación de t_i y t_j en el mismo segmento.

Para entender mejor lo expuesto anteriormente, a continuación se presenta un ejemplo. Sean d_1, d_2, d_3, d_4 los siguientes cuatro documentos: $d_{1=}$ herramientas del lenguaje, d_2= lenguaje de modelado unificado, d_3= lenguajes de programación, d_4= lenguaje de modelado unificado.

Estos documentos se describen por seis (6) términos (descriptores) t1=lenguaje, t2=modelado, t3=relación, t4=clase, t5=modernas, t6=diagrama. Sean S1 y S2 las matrices de correlación correspondientes a los contextos de Ciencias de la Computación y de Lenguas Modernas respectivamente (ver Figura 4).

Las matrices P_{B1} y P_{B2} que se muestran en la Figura 5 representan los proyectores de los contextos de Ciencias de la Computación y de Lenguas Modernas respectivamente, los cuales se calcularon usando la descomposición de Cholesky (que retorna una matriz triangular con columnas independientes P_B tal que $S = {P_B}^T . P_B$) [78]:

S_1=

1	0,7	0,5	0,5	0,8	0,7
0,7	1	0,8	0,5	0,3	0,4
0,5	0,8	1	0,8	0,2	0,7
0,5	0,5	0,8	1	0,2	0,8
0,8	0,3	0,2	0,2	1	0,1
0,7	0,4	0,7	0,8	0,1	1

S_2=

1	0,5	0,6	0,4	0,7	0,4
0,5	1	0,3	0,2	0,6	0,2
0,6	0,3	1	0,1	0,5	0,2
0,4	0,2	0,1	1	0,7	0,1
0,7	0,6	0,5	0,7	1	0,1
0,4	0,2	0,2	0,1	0,1	1

Figura 4. Matrices de correlación S_1 y S_2

P_{B1}=

1.00	0,70	0,50	0,50	0,80	0,70
0,00	0,71	0,63	0,21	-0,3	-0,1
0,00	0,00	0,59	0,70	0,05	0,72
0,00	0,00	0,00	0,46	-0,3	-0,06
0,00	0,00	0,00	0,00	0,32	-1,7
0,00	0,00	0,00	0,00	0,00	0,00

P_{B2}=

1,00	0,50	0,60	0,40	0,70	0,40
0,00	0,86	0,00	0,00	0,28	0,00
0,00	0,00	0,80	-0,1	0,10	-0,05
0,00	0,00	0,00	0,90	0,48	-0,07
0,00	0,00	0,00	0,00	0,42	-0,3
0,00	0,00	0,00	0,00	0,00	0,85

Figura 5. Proyectores de los contextos Ciencias de la Computación y de Lenguas Modernas

Sea *q* la consulta "modelado" representada por los coeficientes $b = [0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0]$. Si esta consulta se expresa en el contexto de "Ciencias de la Computación" el vector de consulta es $q_1 = P_{B1}.b$= $[0{,}70 \quad 0{,}71 \quad 0 \quad 0 \quad 0 \quad 0]^T$. Por otro lado, si el contexto fuera "Lenguas Modernas" el vector consulta $q_2 = P_{B2}.b$ sería $[0{,}50 \quad 0{,}86 \quad 0 \quad 0 \quad 0 \quad 0]^T$.

De manera similar, considere los coeficientes a_1, a_2, a_3 y a_4 de los documentos d_1, d_2, d_3 y d_4 respectivamente:

$a_1 = [1 \quad 0 \quad 0 \quad 0 \quad 1 \quad 0]^T$

$a_2 = [1 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0]^T$.

$a_3 = [1 \quad 0 \quad 1 \quad 1 \quad 0 \quad 1]^T$.

$a_4 = [1 \quad 1 \quad 1 \quad 1 \quad 0 \quad 1]^T$.

En la Tabla 1 y la Tabla 2 se muestran los resultados de aplicar la función de ranking para la la consulta "modelado" en el contexto de Ciencias de la Computación y en el contexto de Lenguas Modernas respectivamente: Al aplicar la función de ranking para la consulta "modelado", en el contexto de Ciencias de la Computación, el orden de la presentación de los documentos seria d_4, d_3, d_2, d_1:

Documento	Función de ranking	Resultado
d_4	$a_4^T.P_{B1}^T.P_{B1}.b$	3,4
d_3	$a_3^T.P_{B1}^T.P_{B1}.b$	2,4
d_2	$a_2^T.P_{B1}^T.P_{B1}.b$	1,7
d_1	$a_1^T.P_{B1}^T.P_{B1}.b$	1,0

Tabla 1. Función de ranking en el contexto de Ciencias de la Computación

Para la misma consulta en el contexto de Lenguas Modernas, el orden de la presentación de los documentos seria d_4, d_2, d_3, d_1:

Documento	Función de ranking	Resultado
d_4	$a_4^T.P_{B2}^T.P_{B2}.b$	2,2
d_2	$a_2^T.P_{B2}^T.P_{B2}.b$	1,5

d_3	$a_3^T . P_{B2}^T . P_{B2} . b$	1,2
d_1	$a_1^T . P_{B2}^T . P_{B2} . b$	1,1

Tabla 2. Función de ranking en el contexto Lenguas Modernas

La explicación de los diferentes resultados se deriva de la estrecha relación entre los términos, que son mayores en el contexto de Ciencias de la Computación que en Lenguas modernas, como se muestra en S1 y S2. Además, el orden de la presentación de los documentos es diferente dependiendo de cómo ellos mismos y la consulta se ven reflejados en cada uno de los contextos. Por lo tanto, para poder obtener un "correcto" ordenamiento de los documentos, es decir, para entregar resultados más relevantes a las necesidades de los usuarios es de vital importancia conocer su contexto.

11. Filtrado Colaborativo

A mediados de los años 90, diversos investigadores empezaron a trabajar en los sistemas de recomendación motivados por las limitaciones de los sistemas de búsqueda tradicionales. Estos sistemas de recomendación han sido diseñados para recolectar la experiencia de los usuarios con los ítems de información y hacer recomendaciones con base en esa experiencia. Existen básicamente dos técnicas conocidas de filtrado colaborativo: basadas en usuario, que pretenden encontrar a otros usuarios que tengan gustos similares; y las basadas en ítem, donde un usuario estaría interesado en encontrar ítems que son similares a los que le gustaron a un usuario anterior [81], Amazon [38] es es tal vez el sistema de recomendación más conocido y usado en la actualidad, y es de especial interés porque maneja más de 29 millones de usuarios y varios millones de ítems en su catalogo. Se basa en un algoritmo de filtrado colaborativo de ítem a ítem [82]. En general el filtrado colaborativo es el proceso de filtrar información o patrones usando técnicas que involucran la colaboración de múltiples agentes, puntos de vista, fuentes de datos, entre otros [83-85], para el caso de Amazon los ítems de información son los

productos y las personas que los compran dan sus distintos puntos de vista, que sirven de recomendación a quienes compran productos similares.

Uno de los principales problemas que enfrentan la mayoría de técnicas de filtrado colaborativo es la falta de información, conocido en la literatura como el problema del ramp-up o cold start [86]. Este problema se refiere a dos situaciones: la primera, cuando un nuevo usuario ingresa al sistema, ya que no se tiene información sobre sus preferencias, la segunda, es análoga, pero se presenta cuando se crea un nuevo servicio, ya que no hay ninguna información (experiencias) de ningún usuario. Otro tipo de técnica de filtrado colaborativo hace uso de un enfoque basado en la preferencias, el usuario es encuestado sobre sus preferencias relacionadas con ítems, y soportándose en la teoría de la utilidad multi-atributo se encuentran los ítems (productos) preferidos, independientemente del tamaño del conjunto de alternativas [85]. Estos sistemas, en muchos casos generan una sobrecarga cognitiva en el usuario, al momento de definir sus preferencias, y no son adecuados cuando las preferencias de los usuarios son extrañas. Otras investigaciones muestran diferentes perspectivas de usar el filtrado colaborativo, por ejemplo, el uso de la información que generan los usuarios (social bookmarking) para mejorar la búsqueda web [87]; el uso de ontologías junto con el filtrado colaborativo ítem a ítem para superar el problema de falta de información [85]. Marlin [84] en su tesis doctoral hace una descripción detallada de las técnicas y su relación directa con el aprendizaje de máquina.

A continuación se muestran investigaciones que se han realizado desde el año 2004 hasta la actualidad (2012) con respecto a recuperación de información en búsquedas Web considerando: personalización, filtrado colaborativo, expansión de consultas y/o feedback de usuario:

En [84] se muestra el filtrado colaborativo desde una perspectiva de aprendizaje de maquinas, analiza los métodos de predicción existentes y muestra que muchos de ellos son simples aplicaciones o modificaciones de uno o más métodos de

aprendizaje de máquina de clasificación, regresión, clustering, reducción de dimensionalidad, y estimación de la densidad, adicionalmente presenta un nuevo procedimiento experimental para examinar formas más claras de generalización, implementan un total de 9 métodos de predicción. El autor del proyecto propone como trabajo futuro tomar los modelos existentes, y considerar la eliminación de algunas de las hipótesis que sustentan el desarrollo de estos modelos, además la ampliación de algunos de los modelos que se estudiaron hasta llegar a nuevas formulaciones de filtrado colaborativo, así como, ampliar algunos de los métodos propuestos para usar con grandes colecciones de documentos electrónicos tales como la Web.

En [88] presentan un entorno de búsqueda colaborativo denominado VisSearch que intenta compartir resultados de búsqueda Web entre las personas que tienen intereses comunes (por ejemplo los estudiantes universitarios que toman el mismo curso) para tomar ventaja del conocimiento pasado (por ejemplo, recursos web útiles recolectados por los estudiantes que tomaron la misma clase en semestres anteriores). Consta de tres componentes principales: 1) VisSearch Client, responsable de proporcionar la mayoría de las interfaces de usuario para buscar en la Web y organizar los resultados de la búsqueda, para lo cual crea representaciones visuales de varios procesos de búsquedas, tales como la presentación de una consulta o la creación de un bookmark en un sitio Web útil, de modo que los usuarios pueden reutilizar esas búsquedas previamente realizadas. 2) VisSearch Server, recoge los resultados de búsqueda web de varios VisSearch Clients y los guarda en un repositorio central y, 3) VisSearch Recommendation Engine, compara los resultados de la búsqueda Web recolectados en el VisSearch Server con cada uno de los otros para encontrar patrones significativos en las consultas y en los recursos Web útiles resultantes, aplicando un algoritmo de minería de datos de reglas de asociación. Dichos patrones son usados posteriormente como recomendaciones para orientar a otros estudiantes que buscan sobre los mismos temas o similares. Aunque las recomendaciones fueron útiles, dicha información no fue capaz de cubrir los diversos intereses de los sujetos del grupo experimental, hay dos razones

principales: 1) el tamaño de la colección de datos usada para encontrar reglas de asociación en este estudio no fue suficientemente grande comparados con estudios de minería de datos usuales, 2) los estudiantes que participaron fueron muy diversos, lo que sugiere que el ambiente VisSearch necesita ser evaluado con un grupo de usuarios más homogéneo. Finalmente, en este estudio no mencionan cual tecnología juega un rol más significante y en qué contexto.

En [89] se presenta un método de búsqueda PCS (Page Clipping Synthesis) que permite extraer párrafos relevantes de otros resultados de búsqueda Web, aplica dinámicamente un algoritmo genético, que emplea en el método de codificación de genes un patrón de coincidencia basado en texto sencillo, para generar un conjunto de los mejores recortes de la página ejecutados en un período de tiempo controlado. Este enfoque es limitado en la codificación páginas Web con multimedia y es difícil generar nuevas combinaciones, como cosas, relaciones, hechos, y las tendencias de los párrafos. Los autores afirman que necesitan más retroalimentación del usuario para afinar los mecanismos de búsqueda. Sin embargo, los recortes de la página ofrecen a los usuarios la información que más les interesa, por tanto, los usuarios ahorran tiempo y muchos problemas en la navegación de los hipervínculos.

En [90] se presenta un algoritmo de clustering llamado WebDCC (Web Document Conceptual Clustering), que aplica el concepto de aprendizaje incremental, sin supervisión sobre los documentos Web para adquirir perfiles de usuario. A diferencia de la mayoría de los enfoques de perfiles de usuario este algoritmo ofrece soluciones comprensibles de clustering que pueden ser fácilmente interpretados y explorados por usuarios y otros agentes. Al extraer semántica de las páginas Web, este algoritmo también produce resultados intermedios que finalmente pueden ser integrados en un formato comprensible por maquinas como una ontología. Los resultados empíricos del uso de este algoritmo en el contexto de un agente inteligente de búsqueda han demostrado que puede alcanzar altos niveles de precisión en lo que se refiere a páginas Web.

La tesis [91] explora cómo la personalización en la búsqueda Web puede ayudar a las personas a aprovechar interacciones de información pasadas únicas, presenta un modelo de lo que las personas recuerdan sobre los resultados de la búsqueda, y muestra que es posible de forma invisible para el usuario, mezclar información nueva dentro de listas de resultados anteriormente encontrados. La personalización repite resultados de búsqueda, de esta forma permite que las personas encuentren de manera efectiva tanto la información de nueva como antigua utilizando la misma lista de resultados

En [92] examinan los cuellos de botella generados por los métodos de expansión de consulta convencionales, que se basan en la recuperación de documentos que se utilizan como fuente para obtener los términos de la expansión, e investigan métodos alternativos para reducir el tiempo de evaluación de las consultas. Proponen un método que toma términos candidatos de los resúmenes de los documentos, reduciendo significativamente el tiempo requerido para la expansión de la consulta en un factor de 5-10 y que mantiene la efectividad de los métodos convencionales para mejorar la eficacia promedio en recuperación de información.

En [93] presentan un método para la expansión de consulta, que extrae términos adicionales para la consulta, basado en técnicas de realimentación de relevancia de usuario. De acuerdo a dicha retroalimentación, el método calcula el grado de importancia de los términos relevantes de los documentos en la base de datos documental, así los términos relevantes que tienen un mayor grado de importancia pueden llegar a ser términos adicionales para la consulta. El método propuesto utiliza reglas difusas para inferir los pesos de los términos adicionales. Luego, los pesos de los términos adicionales y los pesos de los términos originales de la consulta se utilizan para formar el nuevo vector de consulta, y este vector se usa para recuperar los documentos. Como resultado, el método de expansión de consulta propuesto aumenta las tasas de precisión y las tasas de recall de los sistemas de recuperación de información.

En [94] muestran cómo a través de una adecuada interpretación y diseño del experimento, el feedback implícito puede proporcionar datos de entrenamiento económicos y precisos en forma de preferencias por parejas. Proveen un algoritmo de aprendizaje de máquina que puede utilizar estas preferencias, y demuestra cómo integrar todo en un motor de búsqueda operacional que aprende. En conjunto estos dos experimentos muestran cómo el uso del feedback implícito y el aprendizaje de máquina pueden producir motores de búsqueda altamente especializados. Mientras que los sesgos hacen que los datos del feedback implícito sean difíciles de interpretar, y las preferencias de parejas resultantes pueden ser utilizados para un aprendizaje eficaz. Sin embargo, aún queda mucho por hacer que van desde hacer frente a cuestiones de privacidad y el efecto de las nuevas formas de spam hasta el diseño de experimentos interactivos y de métodos de aprendizaje activo.

En [95] se propone un método que utiliza SVD (Singular Value Decomposition), junto con información demográfica en varios puntos del procedimiento de filtrado con el fin de mejorar la calidad de las predicciones generadas. Las pruebas de eficiencia del enfoque resultante se realizaron con dos enfoques de filtrado colaborativo comúnmente usados (basado en usuarios y basado en ítems). La parte experimental del trabajo consiste en una serie de variaciones del enfoque propuesto. Los resultados muestran que la combinación de los datos demográficos con SVD es prometedora, ya que no sólo aborda algunos de los problemas que registran los sistemas de recomendación, sino que también ayuda a mejorar la precisión de los sistemas que emplean la misma.

El objetivo de [96] es proponer y clasificar automáticamente una lista de nuevos ítems a un usuario, basados en patrones de votación anteriores de otros usuarios con gustos similares. El modelo propuesto puede ser considerado como un sistema de recomendación colaborativo basado en computación Soft. Muestra la combinación de redes Bayesianas, las cuales permiten una representación intuitiva de los mecanismos que rigen las relaciones entre los usuarios, y la teoría de conjuntos Fuzzy, lo que permite representar la ambigüedad o vaguedad en la descripción de

las calificaciones, mejorando la exactitud del sistema. El modelo puede ser aplicado para resolver diferentes tareas de recomendación (tales como encontrar buenos ítems o predecir tasas).

En [97] se presenta un método sencillo e intuitivo de extracción de registros de consulta, en motores de búsqueda, para obtener un filtrado social rápido. Con el fin de conseguir recomendaciones bien equilibradas, combinan dos métodos: primero, se modela el comportamiento de búsqueda secuencial de los usuarios del motor de búsqueda y este comportamiento se interpreta como el refinamiento de consulta del lado del cliente, que es aprovechado para aprender información útil que ayuda a generar consultas relacionadas. En segundo lugar, se combina este método con un método de similitud de texto tradicional o basado en contenido, para compensar la falta de sesiones de consulta y escasez de datos de registros (logs) de consulta reales. De acuerdo a las pruebas realizadas el método se adapta al comportamiento de búsqueda dinámico de los usuarios y se obtienen buenas recomendaciones que permiten filtrar información rápidamente.

En [98] se muestra un sistema de recomendación de documentos en Internet que permite personalizar el contenido para luego hacer sugerencias basándose en el perfil de navegación del usuario. El método adopta un enfoque de expansión semántico, donde se incluyen relaciones “es un” y “no es un” para conectar conceptos, con el fin de construir el perfil de usuario mediante el análisis de los documentos previamente leídos por la persona. Una vez que el perfil del cliente se construye, el sistema puede proporcionar contenido personalizado. Los resultados de un estudio experimental muestran que el enfoque propuesto es significativamente mejor que el enfoque tradicional basado en palabras clave en la captura de los intereses de los usuarios. No obstante, la principal limitación del enfoque es cómo construir una red de expansión semántica completa y útil para cubrir los principales conceptos y relaciones.

En [99] proponen un método de generación de snippet (resumen de un documento que le permite al usuario entender si dicho documento es relevante sin acceder a él) que aplica un enfoque estadístico de expansión de consulta con realimentación de pseudo-relevancia y técnicas para resumir el texto, con el fin de extraer frases destacadas y obtener snippets de buena calidad. Todo el proceso se resume como sigue: (1) Cada frase se segmenta, y se analiza para extraer términos nominales como los términos candidatos de la expansión de consulta, (2) Se separan la frases relevantes y no relevantes, por si cada frase incluye un término de búsqueda, (3) El peso de la relevancia de los términos candidatos se estima por la función de ponderación estadística y la consulta inicial se expande mediante el uso de los términos candidatos con el mayor peso de relevancia, (4) La puntuación de importancia de cada frase se calcula utilizando el peso de relevancia de los términos de la consulta expandida y la información de ubicación de esa frase, (5) Por último, se genera el snippet con las frases de mayor puntuación de relevancia. En los resultados experimentales, el método propuesto mostró un rendimiento mucho mejor que otros métodos, incluyendo los de los buscadores comerciales como Google y Naver.

En [100] presentan una propuesta para mejorar la clasificación en sistemas de contenido social, por medio de un modelo que integra las preferencias del usuario y los términos de consulta relacionados semánticamente. Usan las anotaciones colaborativas de un catálogo de libros en línea para crear un grafo de anotación social y estudiar el efecto de la personalización y el smoothing (suavizado) para aumentar la longitud de la consulta. En la propuesta se demuestra que la personalización y el suavizado permiten al usuario encontrar contenido relevante con menos términos de consulta en comparación con la clasificación de contenido basado en frecuencia de términos TF-IDF.

En [101] se estudia el problema de la recomendación formado por dos tareas: (i) filtrar ítems útiles/interesantes, (ii) guiar al usuario a buenas recomendaciones. En el documento se centran en la segunda tarea mostrando un experimento que

demuestra que los algoritmos de aprendizaje de máquina comúnmente aplicados en la primera tarea son inútiles cuando se aplica a la tarea de guiar. En un experimento centrado en la segunda tarea los autores del proyecto observaron un muy mal comportamiento de las técnicas de aprendizaje de máquina comúnmente empleadas en esta área y afirman que este comportamiento tiene que ver con las siguientes características distintivas del proceso de guiar: 1) la tarea de guiar persigue dos objetivos al mismo tiempo: trata de recomendar tanto como sea posible y trata de recomendar sólo recomendaciones que se puedan seguir, 2) hay una necesidad de una nuevos casos de entrenamiento para mejorar la tarea de recomendación mientras el trabajo avanza.

En [102] describen el protocolo Asknext, que permite automatizar el intercambio de conocimiento, mediante la conexión de agentes que utilizan redes sociales. El protocolo combina sistemas sociales donde los usuarios responden a preguntas (o consultas), con sistemas de retroalimentación social, para ordenar los resultados de búsqueda. En Asknext, cada agente representa a un usuario, sus contactos y su conocimiento. Cuando un agente recibe una pregunta, se trata de responder, si no tiene una respuesta disponible, el agente envía un mensaje con la pregunta a sus contactos o lo muestra a su usuario. Los aportes principales de Asknext incluyen la reducción del envío de mensajes, después que una necesidad de información ha sido satisfecha. Finalmente, se compara su desempeño con los resultados de Sixearch (basado también en protocolos de búsqueda social) y se concluye que Asknext mejora significativamente la escalabilidad de los protocolos de búsqueda social.

En [103] proponen un modelo para la evaluación de relevancia multidimensional (en un contexto de RI donde la relevancia se modela como una propiedad multidimensional de los documentos) que realiza una agregación de criterios teniendo en cuenta la existencia de una relación de priorización entre ellos. Los operadores de priorización propuestos son “scoring”, que modela una situación donde el peso de un criterio menos importante es proporcional al grado de

satisfacción de criterios más importantes y “and” que considera que el criterio con menor satisfacción, en el grado de satisfacción global, depende tanto de su grado de satisfacción y de su importancia para el usuario. Una ventaja de los operadores propuestos es que permiten calcular los pesos de los criterios de una manera sencilla y sin requerir ningún método de aprendizaje. La utilidad y la eficacia de este modelo se demostró por medio de un caso de estudio sobre RI personalizada con múltiples criterios de relevancia de los documentos como: referencialidad, cobertura, conveniencia y fiabilidad.

En [104] se presenta un sistema de recuperación de vídeos que se puede utilizar para realizar consultas basadas en contenido, en una gran base de datos de videos de manera eficiente. Se muestra que mediante el uso de ABRS-SVM, una técnica para incorporar retroalimentación de Relevancia (RF) en los resultados de la búsqueda, es posible lograr rápidamente resultados útiles incluso cuando se trata de consultas de acción humana muy complejas, tales como las películas de Hollywood, donde las diferencias en el punto de vista, la iluminación, el vestuario y la forma como se lleva a cabo la acción confunden la precisión. A pesar de esta dificultad, se demuestra que es posible alcanzar, después de sólo unas pocas iteraciones de retroalimentación de relevancia, mejoras significativas en la precisión de los resultados de la búsqueda, sin ruptura semántica o comprensión cognitiva de la consulta del video original. Como trabajo futuro proponen incluir reconocimiento de objetos, datos de audio y el uso de más información contextual sobre las escenas, o el conocimiento sobre la estructura del cuerpo humano, de acuerdo con realimentación de relevancia, con el fin de mejorar aún más la capacidad para organizar y buscar videos con acciones humanas complejas.

En [105] presentan una fórmula para cuantificar el concepto de similitud semántica e introducen un nuevo esquema de clasificación semántica para búsquedas basadas en palabras claves XML, teniendo en cuenta que a diferencia de los datos de texto, los datos XML contienen una semántica rica, que obviamente son útiles para la recuperación de información. A diferencia de la mayoría de los métodos actuales

para la búsqueda por palabras claves XML, que o bien no tienen en cuenta clasificación de relevancia o realizan clasificación de relevancia con las técnicas tradicionales de IR de texto, en este trabajo se propone clasificar la relevancia entre una consulta por palabras claves y un fragmento de XML por su similitud semántica, sobre la base de un análisis en profundidad de las necesidades de información del usuario y la semántica estructural XML. Los experimentos demuestran que el esquema propuesto supera los enfoques existentes en términos de calidad de búsqueda y alcanza una alta eficiencia y escalabilidad.

De acuerdo a lo expuesto anteriormente, se puede notar que durante los últimos años se han venido desarrollando diferentes propuestas, que pretenden disminuir la sobrecarga de información con la que se cuenta en la actualidad y mejorar la relevancia de los resultados entregados por los sistemas de recuperación de información tradicionales; sin embargo no se ha intentado sacar provecho del adecuado manejo de la información de contexto del usuario, propuesto por Melucci [1], e integrarlo con técnicas de filtrado colaborativo, tal como se propone en la presente tesis de maestría.

Parte III – Modelo de Meta Buscador Propuesto

A continuación se describe la propuesta del modelo de meta buscador Web, que integra el filtrado colaborativo (basado en ítems) con la propuesta de Massimo [1] basada en proyectores sobre planos que se originan en la información del contexto del usuario. Este modelo cuenta con los siguientes pasos y componentes generales para su funcionamiento (ver Figura 6):

1. **Registrarse e Ingresar al sistema:** el usuario llena inicialmente un formulario para registrase. Luego cada vez que pretenda usar el meta buscador Web deberá realizar un proceso de login.

2. **El usuario ingresa la consulta basado en palabras clave**: El usuario digita la consulta de manera tradicional, basado en palabras claves. En principio lo que se busca es que el modelo se comporte (desde la perspectiva del usuario) de la misma manera como lo hacen los actuales buscadores web.

3. **Pre procesamiento de consulta**: En el pre procesamiento de la consulta se eliminan acentos y caracteres especiales, se organizan las palabras de la consulta, se eliminan las palabras vacías, se dejan todas las palabras en minúsculas y se realiza el proceso de stemming (llevar diferentes palabras a una raíz común, por ejemplo running y runner a su raíz común run) y luego se calcula la frecuencia de los términos en la colección de documentos [7, 13].

4. **Expansión de la consulta**: de manera automática, el sistema realiza un proceso de expansión de consulta basado en la información de contexto disponible del usuario y de la comunidad y muestra una lista desplegable con los términos que se espera que complementen mejor los ya digitados por el usuario, para que éste los seleccione. Este primer proceso de expansión se hace explícitamente y con la aprobación del usuario (interacción del modelo con el usuario). Luego, cuando el

usuario termina de digitar la consulta y solicita formalmente la búsqueda, el modelo realiza un proceso de expansión de consulta que es oculto al usuario, en el cual se agregan términos a la consulta original, basado en los términos originales de la consulta y el contexto del usuario. Esta última consulta se denomina consulta expandida.

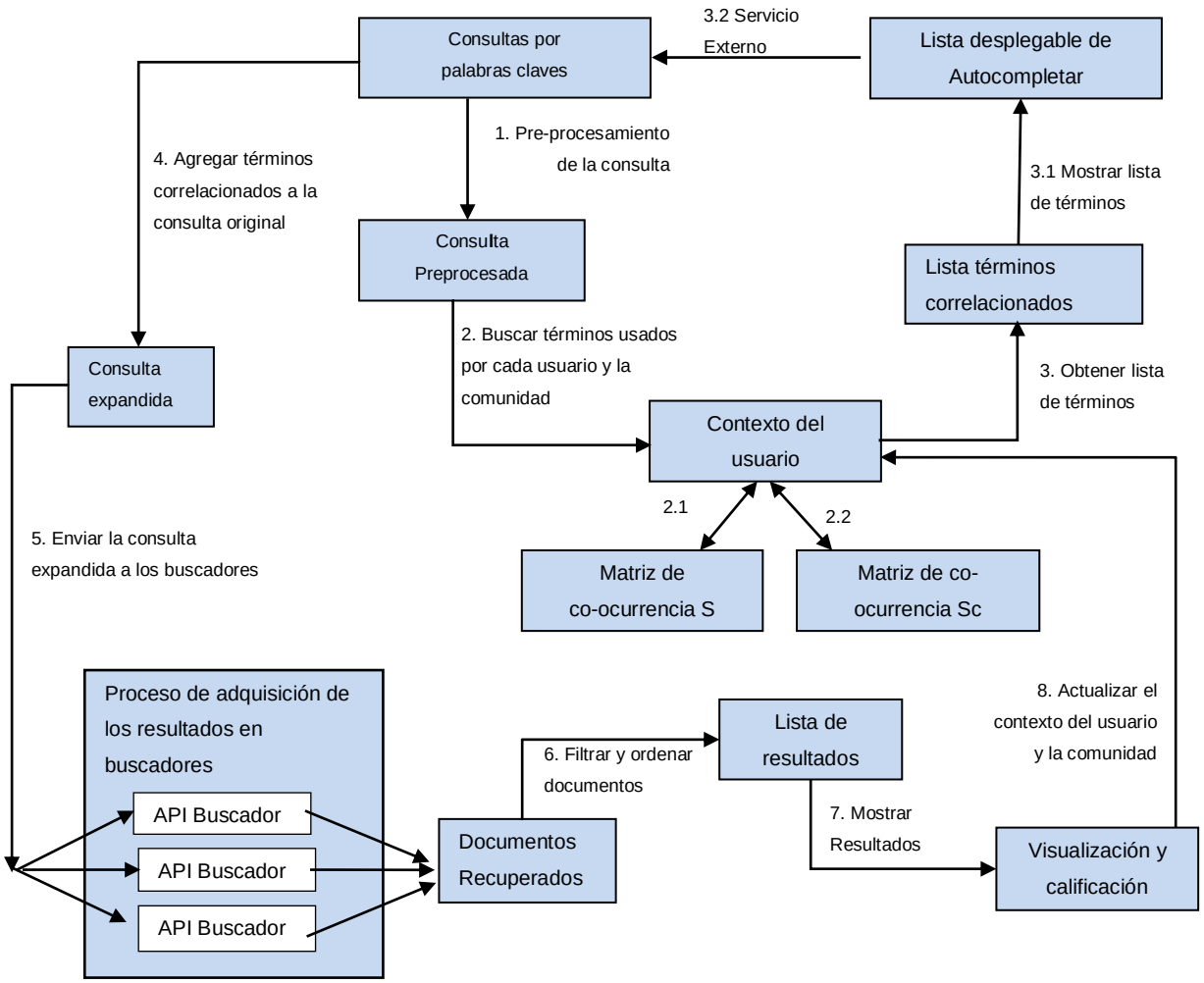


Figura 6. Modelo General del Meta buscador

5. **Recuperación de documentos de los buscadores tradicionales**: Se envía la petición de búsqueda (consulta expandida) a las APIs de los buscadores tradicionales (por ejemplo: Google, Bing y/o Yahoo!). Los documentos se organizan en una tabla de términos por documentos estándar, comúnmente

conocida como Matriz de Términos por Documentos [13], aplicando el pre procesamiento de cada documento y registrando solamente la frecuencia de los términos en los documentos.

6. **Filtrado de información**: basándose en la propuesta de Massimo [1] (información de contexto representada en un espacio vectorial no ortogonal), se toma provecho de la información de contexto del usuario y se complementa con la información de la comunidad, basado en técnicas de filtrado colaborativo [84], para filtrar y ordenar la información que se muestra al usuario en una lista ordenada (rankeada) de documentos tal y como lo presentan los buscadores tradicionales.

7. **Visualizar y calificar resultados**: a través de una interfaz gráfica de usuario, el usuario del modelo califica los documentos como relevantes o no relevantes a las necesidades de información inicialmente definidas en la consulta.

8. **Modificar información del contexto e información de feedback para la comunidad**: a partir de la calificación que el usuario realiza de los resultados obtenidos, el sistema realiza la gestión de la información de contexto del usuario y de la comunidad. Con esta información de contexto actualizada se afecta el proceso de expansión, filtrado y ordenado de las futuras consultas.

12. Modelo Propuesto en Detalle

Una de las estructuras básicas para poder entender el modelo en detalle, es la forma como se almacena el **contexto del usuario**. Este contexto se almacena en una matriz triangular superior que contiene los términos usados por cada usuario y la relación que tienen cada uno de los términos entre ellos, de ahora en adelante denominada la Matriz de co-ocurrencia S (ver Figura 7). De igual forma la comunidad de usuarios cuenta con una matriz de co-ocurrencia de términos, denominada S_c.

Los pasos 1 a 3 fueron descritos en el modelo general, en este apartado se profundiza en los pasos subsiguientes. En el proceso de **expansión de la consulta (**paso 4**)** se buscan los términos correlacionados utilizados por el usuario y otros usuarios de la comunidad teniendo en cuenta el parámetro numero_Terminos, que indica la cantidad de términos que se desean obtener para mostrar en la lista de autocompletar, actualmente fijado a 10 términos, teniendo en cuenta que ésta es cantidad comúnmente usada en la lista desplegable de autocompletar de los buscadores web tradicionales, pero dicho valor puede ser modificado de acuerdo a los deseos del investigador. En la Figura 8. a) se muestra la representación del proceso de expansión de consulta como una caja negra, y en la Figura 8. b) el esquema del proceso con mayor detalle.

	t_1	t_2		t_K
t_1	1	0,6	0,7	0,5
t_2	0	1	0,8	0,93
…	0	0	1	0,3
t_K	0	0	0	1

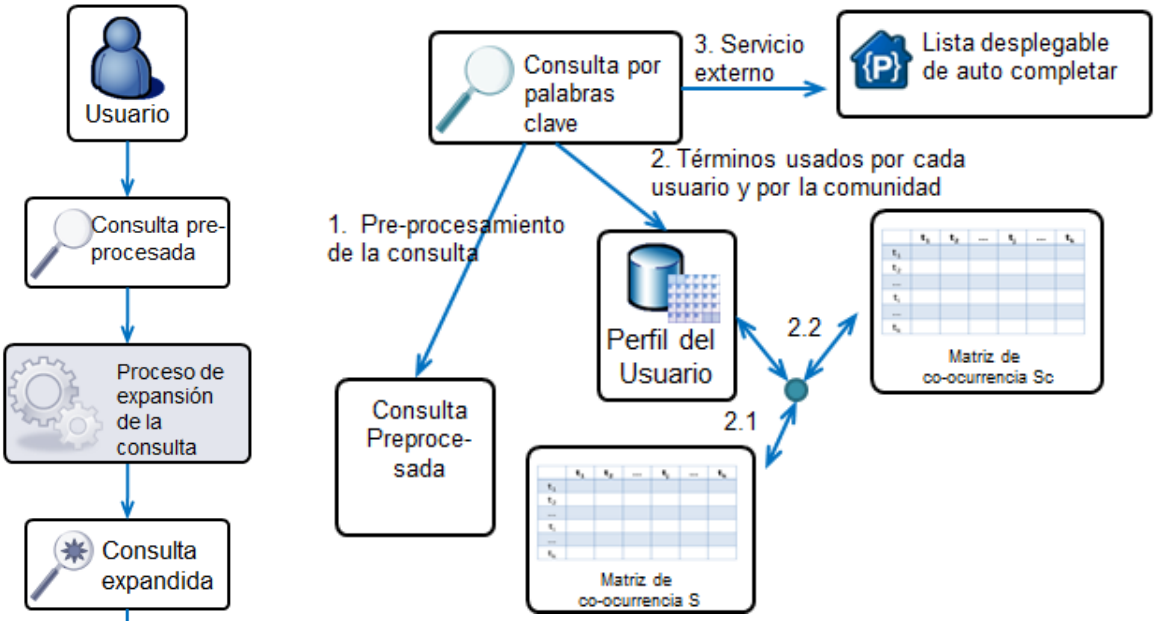
Figura 7. Matriz de co-ocurrencia S con información del contexto del usuario

Después de hacer el pre-procesamiento de la consulta se hace la revisión del contexto del usuario (paso 2 de la Figura 8 b), como se detalla a continuación:

1. La expansión inicial de la consulta se hace a partir de la matriz de co-ocurrencia S del usuario, donde se obtiene la intersección de los términos de mayor co-ocurrencia de la información contextual del usuario con los términos ingresados por el usuario en la consulta actual.

2. Si la intersección es vacía o si aún no se alcanza la cantidad de términos establecidos en el valor del parámetro numero_Terminos, se hace una segunda expansión a partir de la matriz de co-ocurrencia S_c de la comunidad, donde se obtiene la intersección de los términos de mayor co-ocurrencia de la comunidad

con los términos ingresados por el usuario en la consulta actual. Se debe validar si aún no se alcanza el valor del parámetro numero_Terminos,



(a) Proceso como una caja negra

(b) Proceso detallado

Figura 8. Proceso de expansión de consulta

3. Se procede a una tercera expansión a partir de la matriz de co-ocurrencia S del usuario, donde se obtiene la unión de los términos de mayor co-ocurrencia de la información contextual del usuario con los términos ingresados por el usuario en la consulta actual.

4. Si aún no se alcanza el valor del parámetro numero_Terminos, se hace una cuarta expansión, a partir de la matriz de co-ocurrencia S_c de la comunidad, donde se obtiene la unión de los términos de mayor co-ocurrencia de la comunidad con los términos ingresados por el usuario en la consulta actual.

5. En el caso que no se obtengan términos después de aplicar los pasos 1 a 4, se procede a usar un servicio externo para realizar el proceso de expansión tradicional usado por motores como Google, Yahoo! o Bing.

Cabe resaltar que en los procesos de intersección se debe actualizar la correlación, calculando el promedio de las coocurrencias de los términos comunes. De igual forma, cuando se realiza unión sobre las listas de términos correlacionados, se debe calcular el promedio de las correlaciones y luego se deben eliminar los términos repetidos. Posteriormente, se ordena la lista descendentemente (por correlación) y se concatena con los términos o palabras claves usadas originalmente por el usuario actual.

Las matrices S y S_c se registran en la medida en que los usuarios califican los documentos retornados por el sistema. Esta situación implica que para las consultas realizadas antes de la primera calificación, las matrices S y S_c pueden estar vacías ($S = \emptyset$, $S_c = \emptyset$), en este caso los pasos mencionados (1 a 4) anteriormente no hacen expansión de consulta, dependiendo de la matriz que se encuentre vacía.

Finalmente, considerando que el usuario puede ignorar las opciones sugeridas de consulta expandida, el modelo realiza un proceso de expansión adicional que es transparente al usuario, agregando la cantidad de términos con mayor correlación que sean necesarios hasta obtener la consulta expandida final, con la cantidad de términos establecido en el parámetro num_TerminosEImplicita (actualmente fijado a un valor de 5 términos) o que tenga una longitud menor o igual a 70 caracteres, según las restricciones en la longitud de las llamadas a las API's de los buscadores tradicionales.

Después de realizar el proceso de expansión de la consulta, se continúa con la Recuperación de documentos de los buscadores tradicionales (paso 5) teniendo en cuenta que la consulta ahora está compuesta de las palabras claves digitadas

directamente por el usuario, las seleccionadas de la lista de autocompletar y además las obtenidas en el proceso oculto de expansión de consulta.

El proceso de adquisición realiza en paralelo la recolección de los resultados en los diferentes buscadores tradicionales, por ejemplo: Google, Yahoo! y Bing (ver Figura 9). A medida que los resultados son retornados por los buscadores tradicionales se realiza el pre-procesamiento de las entradas, este proceso incluye: Remoción de caracteres especiales, conversión del texto a minúsculas, remoción de palabras vacías y stemming del documento. Del proceso de adquisición de datos se obtiene una colección de documentos representados por

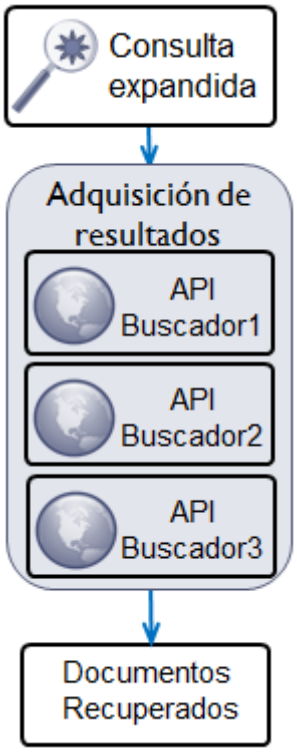


Figura 9. Proceso de recuperación de documentos de los buscadores tradicionales

Posteriormente se realiza el paso 6 correspondiente al **filtrado de información,** el sistema filtra los resultados, ubicando en las primeras posiciones los documentos de mayor relevancia para el usuario. Aquí los resultados se ordenan teniendo en cuenta

el filtrado colaborativo basado en ítems, donde se buscan los términos correlacionados utilizados por el usuario y otros usuarios de la comunidad (Términos consultados anteriormente junto con los términos de consulta actual), además del contexto del usuario, basado en la propuesta [1] explicada anteriormente.

En este paso se tiene en cuenta la creación de una matriz triangular superior auxiliar, denominadas Matriz Scruda del usuario (Figura 10), donde se almacena los términos usados por cada usuario y el valor n_{ij} que representa el número de veces que los termino t_i y t_j se encuentran juntos en los documentos. También se tiene una Matriz Scruda para la comunidad.

	t1	t2	…	tK
t1	1	7	2	5
t2	0	1	3	9
…	0	0	1	6
tK	0	0	0	1

Figura 10. Matriz S cruda del usuario

El proceso que se realiza para filtrar los documentos es el siguiente:

1. Se toman los documentos recuperados de los buscadores tradicionales y se indexan en memoria, se lee el snippet de cada documento recuperado y se llena una matriz auxiliar de Documentos donde se almacena: URL, titulo, texto, texto revisado, texto sin palabras vacías, términos, posición en Google, posición en Yahoo, posición en Bing y orden (que se calcula en el siguiente paso).

2. Se calcula la similitud de cosenos frente a la consulta y se almacena en el campo orden de la matriz Documentos, esto con el fin de poder iniciar el manejo del contexto.

3. Se obtiene el contexto actual del usuario, consultando sus términos relevantes que se encuentran en la matriz de co-ocurrencia S

4. Se calcula la función de ranking, basado en la propuesta [1], para lo cual se obtienen los vectores que representan a la consulta y a los documentos, se calculan los proyectores de la matriz de co-ocurrencia S, posteriormente, se realizan los cálculos necesarios (obtener las transpuestas de los vectores y de las matrices y se realizan los productos de acuerdo al algoritmo presentado en la Figura 11) para finalmente, mostrar los documentos de acuerdo la relevancia que tengan para el usuario de mayor a menor.

Sea *a* el vector que representa la consulta
Sea P_B los proyectores de la matriz S
Sea $l_{(i)}$ el conjunto de documentos
Sea d un documento en $l_{(i)}$ representado por el vector *b*
Para cada documento $d \in l(i)$

$$P_d = a^T . \left({P_B}^T . P_B\right). b$$

Fin para
Ordenar documentos por P_d de mayor a menor
Presentar resultados al usuario

Figura 11. Algoritmo de ranking

Cabe aclarar, que el modelo de meta buscador Web propuesto es *mono temático*, es decir un meta buscador donde los usuarios buscan sobre un único tema, persiguen objetivos comunes, y se considera que tienen un contexto similar. Por otro lado, el meta buscador es *mono lenguaje***,** actualmente funciona específicamente para el idioma inglés. De esta manera, para calcular la función de ranking se debe tener en cuenta que tanto el documento como la consulta se representan en el mismo contexto, de este modo la función de ranking se define como $P_d = a^T . \left({P_B}^T . P_B\right). b$, que obtiene la probabilidad de que un documento haya sido materializado dentro de un contexto basado en la consulta [1]; donde *a* y *b* son coeficientes usados para combinar los vectores base del documento y de la consulta respectivamente. El vector *a* de la consulta tiene un 1 si la palabra clave o término i aparece en la consulta y 0 en caso contrario, de manera similar el vector *b* del documento tiene un 1 si la palabra clave o término i aparece en el documento y 0 en otro caso. P_B es el

proyector del subespacio extendido por los términos evaluados por el usuario, calculado utilizando la Descomposición de Cholesky o SVD.

Luego, se continúa con **la visualización y calificación de los resultados** (paso 7): A continuación el usuario califica como relevantes o no relevantes los documentos desplegados. En seguida, se actualizan la matriz de co-ocurrencia S del usuario actual y la matriz de co-ocurrencia S_C de la comunidad, basada en la información de dos matrices auxiliares denominadas S^+ y C.

La matriz S^+ (ver Figura 12) contiene el número de veces que un término aparece en los documentos relevantes y el número total de veces que aparece en todos los documentos, además el número total de documentos relevantes y el número total de documentos calificados.

	t_1	t_2	t_3	…	t_F	Número de doc.
Doc. Relevantes						
Total documentos						

Figura 12. Matriz S^+

C es una matriz de términos por documentos (ver Figura 13), donde el documento es sólo la URL (identificador) y cada elemento *(i, j)* de la matriz almacena el valor $cf_{i,j}$ calculado por la Ecuación 12

$$cf_{i,j} = \begin{cases} 1, si\ el\ termino\ aparece\ en\ documento \\ 0, de\ otro\ modo \end{cases}$$

Ecuación 12. Valor $cf_{i,j}$ de la Matriz C

Es importante mencionar que para calcular las matrices de co-ocurrencia de los usuarios, los documentos se segmentan utilizando ventanas de texto basado en la propuesta [1], que se usan para colocar los términos en su propio contexto y para relacionar estos términos con otros en el mismo contexto. El tamaño de esta ventana se estableció en 13, en concreto, 6 términos a la izquierda y 6 términos a la derecha

del término objetivo. Este valor puede ser configurado de acuerdo al tamaño de los resúmenes de las páginas web.

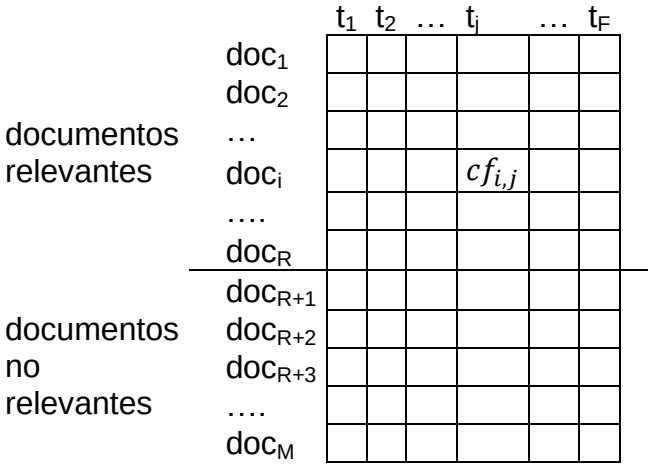


Figura 13. Matriz C de términos por documentos

La matriz de co-ocurrencia S se calcula inicialmente (cuando no existe para el usuario actual como se muestra en la Figura 14) multiplicando cada valor $cf_{i,j}$ de la matriz C por la importancia relativa del término en los documentos consultados por el usuario.

Para el cálculo de la co-ocurrencia de los términos se propone la Ecuación 13

$$S_{ij} = \frac{n_{ij}}{n_i + n_j - n_{ij}} * \frac{r_i}{n_i} * \frac{r_j}{n_j}$$

Ecuación 13. Co-ocurrencia de los términos

Donde:

n_{ij} es el número de veces que el término i y el término j están juntos en los documentos del usuario (se obtiene de la matriz Scruda),

n_i es un número de veces que termino i aparece en los documentos del usuario

r_i: Son las apariciones relevantes del término i para el usuario

n_j es un número de veces que termino j aparece en los documentos del usuario

r_j: Son las apariciones relevantes del término j para el usuario

Es decir, se realiza el producto entre la primera fracción que representa la relación que tienen un par de términos, con la segunda fracción que representa la importancia relativa del primer término, por la tercera fracción que representa la importancia relativa del segundo término.

Sea S la matriz k x k de co-ocurrencia de términos
Sea D el conjunto de documentos recuperados
Para cada documento $d \in D$
  Para cada termino $t_i \in d$
    Sea V_i una ventana centrada alrededor de t_i
      Para cada termino t_j que pertenece a V_i
        $S_{ij} = \frac{n_{ij}}{n_i + n_j - n_{ij}} * \frac{r_i}{n_i} * \frac{r_j}{n_j}$
        Si $(n_i = 0 \; or \; n_j = 0 \; or \; (n_i + n_j - n_{ij} = 0))$ entonces
          $S_{ij} = 0$
        Fin si
        $S_{j,i} \leftarrow S_{i,j}$
      Fin para
    Fin para
    $S_{i,i} \leftarrow 1$
  Fin para
Se calculan los proyectores de S basado en SVD

Figura 14. Algoritmo para calcular la matriz de co-ocurrencia y los proyectores

Las matrices propuestas permiten que el proceso de actualización de la matriz de co-ocurrencia S, cuando llega una nueva evaluación positiva (relevante) o negativa de un documento se pueda realizar rápidamente, actualizando sólo los términos del documento que están en la consulta actual y en la matriz S (ver Figura 15). La matriz S+ se amplía con los nuevos términos del documento que se acaba de calificar, el documento se registra en la matriz C y progresivamente se actualiza la matriz S.

Posteriormente, la matriz S_c de la comunidad, se actualiza calculando la correlación de los términos existentes en dichas matrices con los nuevos términos relevantes usados en la consulta actual, basada en la Ecuación 13 pero teniendo en cuenta los datos de la comunidad.

Sea S la matriz k x k de co-ocurrencia de términos
Sea d el documento recientemente evaluado
Para cada termino $t_i \in d$
Sea V_i una ventana centrada alrededor de t_i
Para cada termino t_j que pertenece a V_i

$$S_{ij} = \frac{n_{ij}}{n_i + n_j - n_{ij}} * \frac{r_i}{n_i} * \frac{r_j}{n_j}$$

Si ($n_i = 0 \; or \; n_j = 0 \; or \; (n_i + n_j - n_{ij} = 0)$) entonces

$$S_{ij} = 0$$

Fin si
$S_{j,i} \leftarrow S_{i,j}$
Fin para
Fin para

Figura 15. Actualización de la matriz S

Finalmente, cabe destacar que el modelo del meta buscador Web tiene como aportes los siguientes: la integración del filtrado colaborativo con la información del contexto propuesta por Massimo Melucci y la actualización dinámica que se hace sobre las matrices S^+, C, S y S_c. sin embargo el costo computacional del modelo propuesto es del orden $O(n^2)$, donde n representa la cantidad de términos de todos los documentos evaluados por el usuario.

Parte IV – Implementación del Meta Buscador

A continuación se describe la construcción del prototipo del meta buscador Web desarrollado en esta investigación como instrumento de evaluación del modelo propuesto en el capítulo previo.

13. Resultados Obtenidos

A continuación se presentan algunos resultados obtenidos del análisis, diseño, e implementación de la aplicación Web basada en el modelo del meta buscador propuesto.

13.1 Casos de Uso

En la primera etapa, correspondiente al análisis del sistema, se estableció el diagrama de casos de uso del meta buscador (ver la Figura 16) a saber: Registrarse, iniciar sesión, realizar búsqueda, consultar documento y evaluar el documento (como relevante o no relevante). A continuación, se describe el caso de uso de alto nivel "Realizar búsqueda"

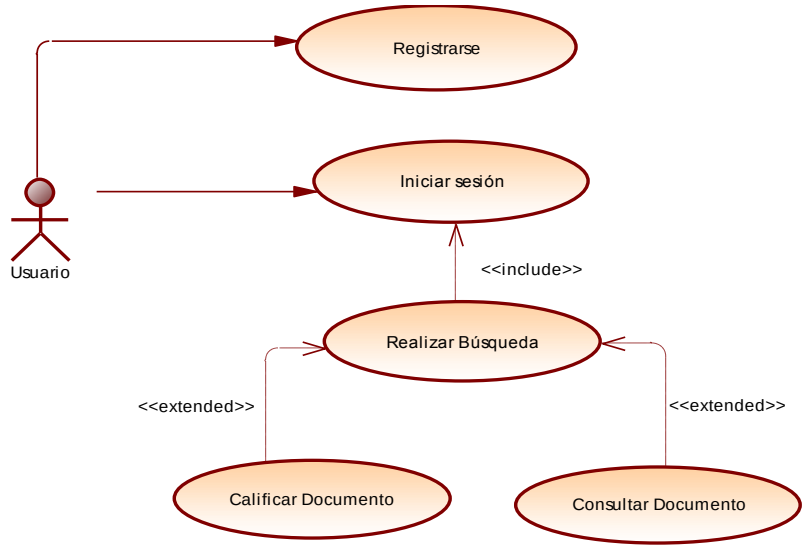
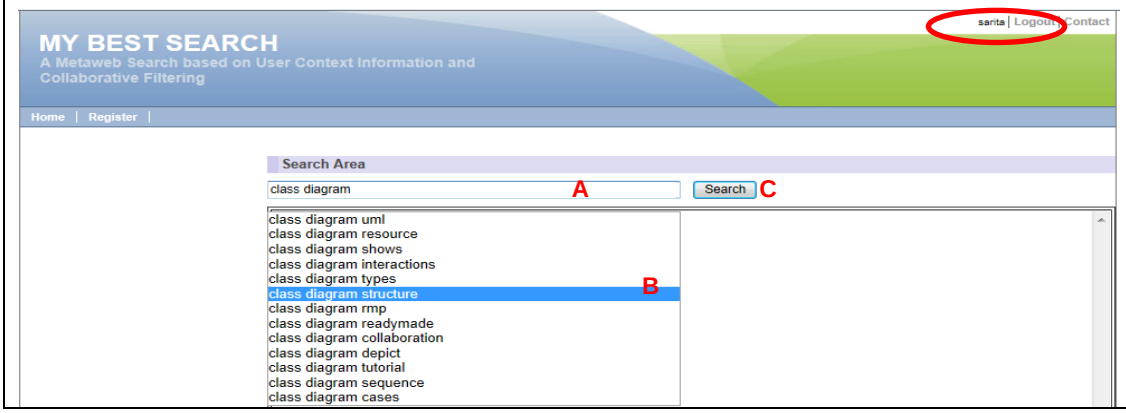


Figura 16. Diagrama de Casos de uso para Usuarios

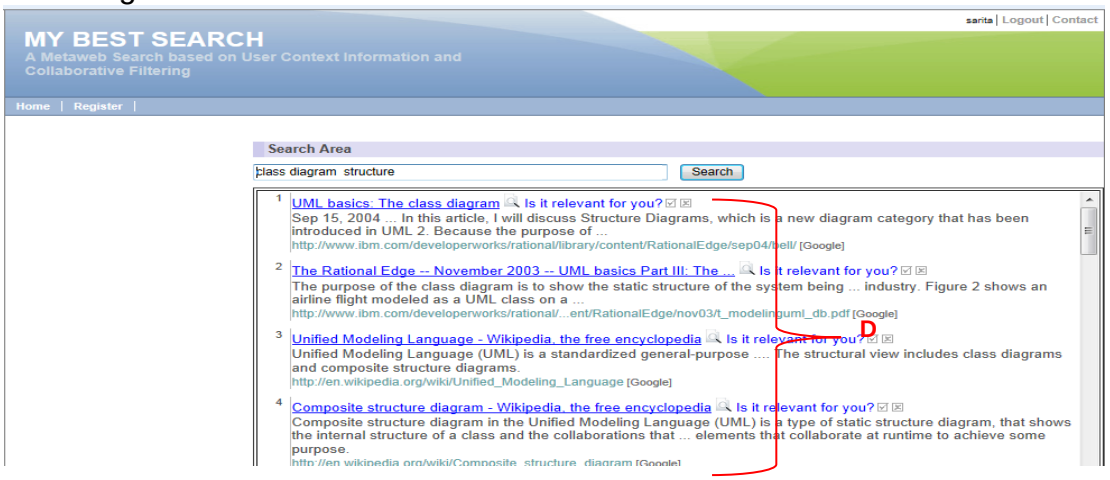
Caso de uso	Realizar búsqueda
Actor	Usuario
Descripción	Este caso de uso comienza cuando un usuario registrado desea realizar una búsqueda, para lo cual ingresa la consulta basada en palabras claves, el sistema pre-procesa la consulta, hace el proceso de expansión y envía la consulta expandida a los buscadores tradicionales, luego se hace el filtrado de los resultados y finalmente los presenta ordenados de acuerdo a la relevancia que tengan para el usuario (aplicando los pasos anteriormente explicados en el modelo).

Tabla 3. Caso de uso de alto nivel Realizar Búsqueda

En cuanto al diseño del sistema, en la Tabla 4 se describe el caso de uso real Realizar búsqueda.

Caso de uso	Realizar búsqueda
Actor	usuario
Propósito	Permitir que el usuario utilice el meta buscador y obtenga resultados de acuerdo a sus necesidades de información
Resumen	Este caso de uso comienza cuando un usuario registrado desea realizar una búsqueda, para lo cual ingresa la consulta basada en palabras claves, el sistema pre-procesa la consulta, hace el proceso de expansión y envía la consulta expandida a los buscadores tradicionales, luego se hace el filtrado de los resultados y finalmente los presenta ordenados de acuerdo a la relevancia que tengan para el usuario.
Interfaz gráfica asociada 1: 	

Interfaz gráfica asociada 2:



Curso normal de los eventos	
Acción del actor	**Respuesta del sistema**
1. El usuario digita en el campo [A] una consulta basada en palabras claves y da clic en el botón [C] de la interfaz gráfica asociada 1.	2. El sistema realiza los siguientes pasos: 2.1 Pre-procesa la consulta 2.2 De acuerdo a la información del contexto del usuario y de la comunidad realiza el proceso de expansión de la consulta y la muestra en la lista desplegable [B] 2.3 Realiza un proceso de expansión que es oculto al usuario. 2.4 Envía la consulta que ahora está compuesta de las palabras claves digitadas directamente por el usuario, las seleccionadas de la lista de autocompletar y además las obtenidas en el proceso oculto de expansión de consulta, a los buscadores tradicionales (Google, Bing y Yahoo) 2.5 Realiza el pro-procesamiento de los documentos recuperados 2.6 Indexa y filtra los documentos 2.7 Muestra los documentos ordenados de acuerdo a la relevancia que tienen para el usuario como se muestra en el

	campo [D] de la interfaz gráfica asociada 2.
Curso alterno de los eventos	
Acción del actor	**Respuesta del sistema**
1. El usuario presiona el botón [C] sin digitar la consulta en el cuadro de texto [A].	2. El sistema retorna el foco al cuadro de texto [A] sin ejecutar acciones.

Tabla 4. Caso de uso real Realizar Búsqueda

13.2 Arquitectura del sistema

Para el sistema se definió una arquitectura que consta de tres capas (ver Figura 17): lógica de presentación, lógica de negocio y lógica de servicios. Al utilizar este tipo de arquitectura se tiene como principales ventajas: la alta escalabilidad, la flexibilidad, la facilidad de construcción y la facilidad del mantenimiento del sistema. A continuación se describen brevemente las funciones que se realizan en cada una de las capas.

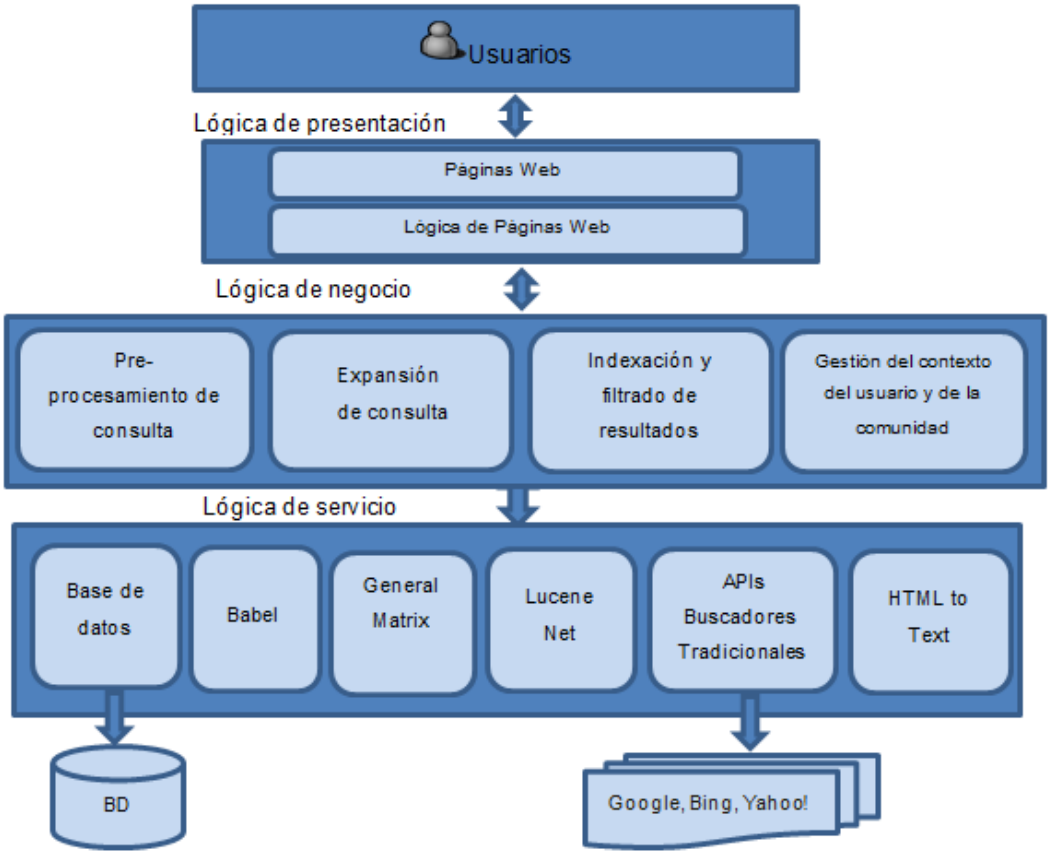


Figura 17. Arquitectura del sistema

Capa de Presentación: también conocida como interfaz grafica de usuario, aquí se encuentran las páginas Web que presentan el sistema al usuario, le comunican la información, capturan la información del usuario y validan los datos entrantes procedentes de éstos. Este nivel se comunica únicamente con la capa de lógica de negocio.

Capa de Lógica de Negocio: es aquí donde se establecen todas las reglas que deben cumplirse. Esta capa se comunica con la capa de presentación y con la capa de lógica de servicios. Entre las funcionalidades que se encuentran en esta capa están: mostrar al usuario la opción de autocompletar, realizar la recuperación de los documentos de los buscadores tradicionales, realizar la indexación y pre-procesamiento de los documentos recuperados, filtrar los documentos y ordenarlos de acuerdo a la relevancia para el usuario, gestionar el contexto del usuario y la información de la comunidad.

Capa de lógica de servicios: En esta capa se encuentra la lógica que permite solicitar al gestor de base de datos almacenar o recuperar datos de él y dar la persistencia a los objetos de la lógica del negocio; por otro lado, permite utilizar la funcionalidad de las APIs de los buscadores tradicionales y de herramientas auxiliares para realizar procesos específicos del sistema.

- Base de datos: implementa la persistencia y el acceso a los datos ocultando los detalles de los repositorios de datos a los niveles superiores. El acceso a datos implementa componentes software que se encargan de acceder a la base de datos para leer y escribir el estado de los objetos de negocio, independizando la aplicación del acceso al motor de la base de datos. En este caso particular se escogió Microsoft SQL Server 2008 Express para el almacenamiento de la información.
- APIs de los buscadores tradicionales: por medio de las APIs de Google, Yahoo! (Esta API en 2012 se volvió paga y no pudo ser usada para la evaluación) y Bing

permite acceder a los servicios ofrecidos por estos buscadores en internet, entre los que se destacan, los servicios de búsqueda en los tres buscadores y el servicio de autocompletar de Google.

- Herramientas adicionales: Babel (Identifica el idioma en el que está escrito un texto), Lucene.Net (API que permite realizar de manera eficiente búsquedas de texto), Html to Text (permite limpiar una cadena de texto que recibe como entrada) y General Matrix (permite hacer operaciones sobre matrices).

13.3 Diagrama de Clases

En la Figura 18 se muestra de manera general el diagrama de clases que componen el sistema, en la Tabla 5 se resume la funcionalidad de cada clase:

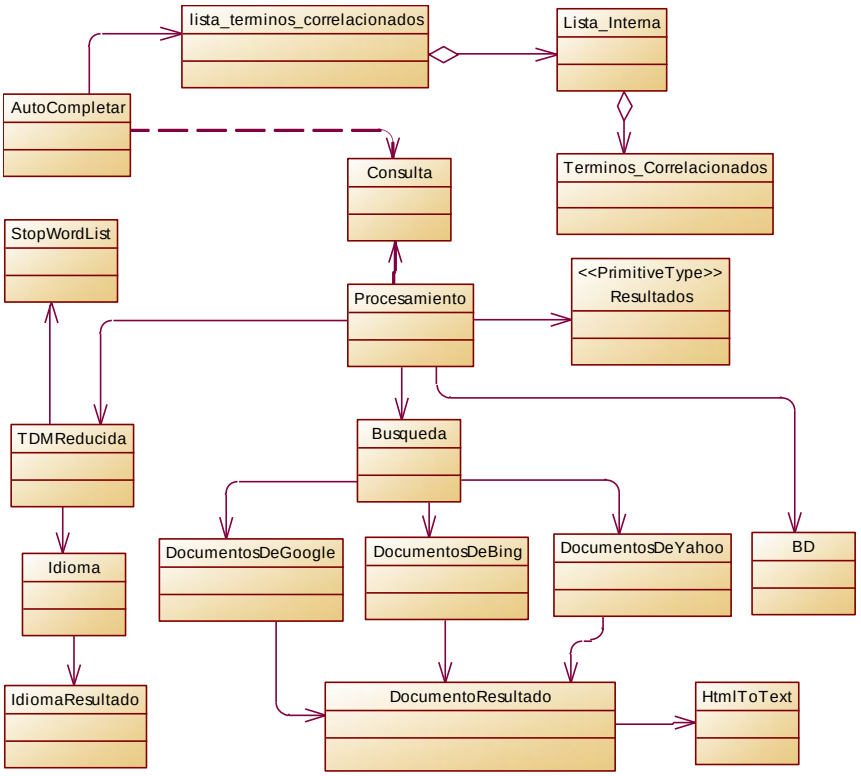


Figura 18. Diagrama de Clases de análisis

CLASE	FUNCION
Consulta	Esta clase realiza el pre-procesamiento de la consulta, es decir de las palabras claves que el usuario digita, para lo cual: elimina los signos de puntuación, convierte el contenido a minúsculas, filtra las palabras vacías y lleva las palabras en ingles a su raíz léxica.
Términos_ correlacionados	Clase que contiene dos campos: el término y el valor de la correlación que ese término tiene con otro término.
Lista_interna	Clase conformada por una lista de la clase Términos_correlacionados. Por cada palabra clave de la consulta, obtiene una lista de los términos correlacionados del usuario, y una lista de los términos correlacionados de la comunidad. Realiza los procesos de intersección y unión entre las listas de términos correlacionados descritos en el paso 4 del modelo del meta buscador propuesto
Lista_terminos_ correlacionados	Clase formada por una lista de la clase Lista_interna, permite obtener una lista con los términos de mayor correlación a los digitados por el usuario y realizar la expansión de la consulta implícita, que es oculta al usuario. Por otro lado, por medio de esta clase se ejecuta el procedimiento almacenado que permite realizar la calificación de un documento como relevante o como no relevante.
Autocompletar	Permite mostrar una la lista auto desplegable de las palabras claves digitadas por el usuario y con los términos que tengan mayor correlación con ellas (proceso expansión de consulta).
Búsqueda	Esta clase se encarga realizar el proceso de recuperación de documentos de los buscadores tradicionales
TDMReducida	Esta clase indexa en memoria los documentos recuperados usando Lucene.NET y realiza el pre-procesamiento de los mismos
DocumentosDeGoogle	Clase que usa la API de Google para hacer el proceso de recuperación de documentos
DocumentosDeBing	Clase que usa la API de Bing para hacer el proceso de recuperación de documentos
DocumentosDeYahoo	Clase que usa la API de Yahoo! para hacer el proceso de recuperación de documentos
StopWordList	Permite identificar y eliminar las palabras vacías en una cadena de texto

CLASE	FUNCION
Idioma	Identifica el idioma utilizado en la consulta
Procesamiento	En esta clase realiza el procesamiento de los documentos recuperados: construye la matriz TDM, calcula la similitud de coseno frente a la consulta digitada por el usuario, permite obtener el orden en el que deben aparecer los resultados, realiza el manejo del contexto (obtener matrices de correlación, calcular proyectores , convertir la consulta a un vector proyectado en el contexto del usuario, convertir los documentos en una Lista de Vectores de Documentos, ordenar los documentos con base en la función de clasificación (ranking). Vea la explicación de la Figura 11
LuceneService	Permite realizar el stemming a los documentos con el algoritmo de Portter, además realizar el proceso de indexación en memoria de los documentos recuperados y construir la matriz de conceptos por documento.
Base de datos	Clase que permite manejar la conexión al motor de base de datos, realizar una consulta SELECT a través de la conexión activa, ejecutar operaciones DML (INSERT, UPDATE, DELETE), revisar una cadena y eliminar código malicioso, e implementar las funciones que ejecutan los procedimientos almacenados para mantener el contexto del usuario y de la comunidad actualizados.

Tabla 5. Descripción de la funcionalidad de cada clase

13.4 Implementación

La implementación del modelo se desarrolló utilizando una arquitectura multi-capa basada en servicios Web XML, en C# (C sharp) Visual Studio .NET 2010. La Figura 19 y la Figura 20 muestran la vista de clases, generada por Visual Studio .NET, la primera corresponde a la lógica de negocios y la segunda a la lógica de servicios establecidas en la arquitectura del sistema. La explicación de cada clase se presenta en la Tabla 5.

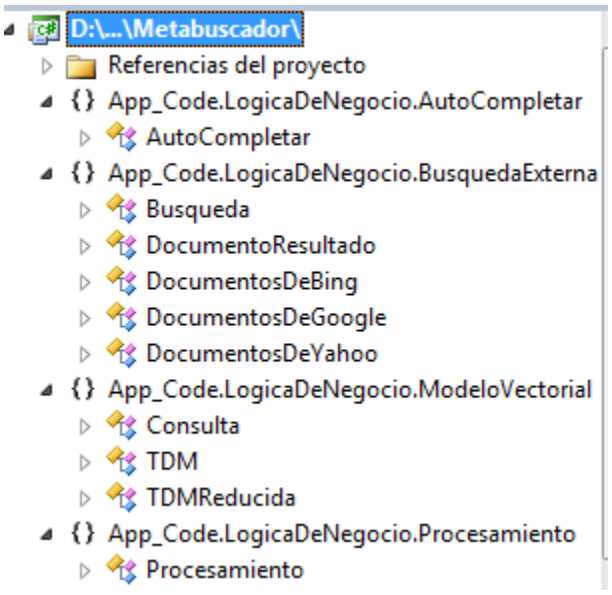


Figura 19. Vista de clases de la lógica de negocio en Visual Studio .NET

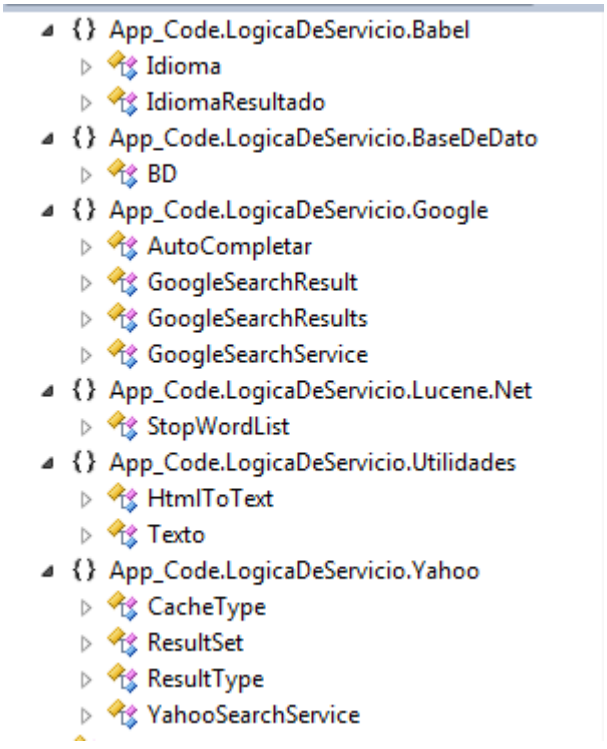


Figura 20. Vista de clases de la lógica de servicios en Visual Studio .NET

13.5 Modelo de la Base de Datos

Todas las matrices del usuario y de la comunidad definidas en el modelo del meta buscador son almacenadas en tablas como se muestra en la Figura 21, de la siguiente manera:

- Matriz S+ = TotalDocUsuario + Termino_Usuario
- Matriz C = Documento_usuario + termino_documento
- Matriz Scomunidad = TotalDocComunidad + Terminos
- Matriz Ccomunidad = Termino_Documentos + Documentos

En la Tabla 6 se explica en forma resumida la finalidad de cada tabla de la base de datos.

TABLA	DESCRIPCION
Documento	Almacena la información del documento para la comunidad
Documento_usuario	Almacena la información de los documentos evaluados por por el usuario
TotalDocUsuario	Almacena la cantidad de documentos evaluados por el usuario y la cantidad de documentos que han sido relevantes para él
TotalDocComunidad	Almacena la cantidad de documentos evaluados por toda la comunidad de usuarios y la cantidad de documentos que se evaluaron como relevantes.
Termino	Almacena la información de los términos para la comunidad
Termino_documento	Almacena la relación de los términos que aparecen en cada documento de la comunidad
Termino_usuario	Almacena la información de los términos evaluados por un usuario
matrizScudra	Almacena el número de veces que un par de términos evaluados por un usuario se encuentran juntos
MatrizScrudaComunidad	Almacena el número de veces que un par de términos evaluados por la comunidad de usuarios se encuentran juntos
MatrizSusuario	Almacena el contexto del usuario, teniendo en cuenta que para controlar el tamaño de esta tabla sólo se guarda la relación de dos términos que superen un umbral de correlación, que se establece en un parámetro de la aplicación
MatrizScomunidad	Similar a la MatrizSusuario pero teniendo en cuenta

TABLA	DESCRIPCION
	los términos de toda la comunidad
Aspnet_User	Almacena la información básica de un usuario del sistema

Tabla 6. Descripción de tablas de la base de datos del sistema

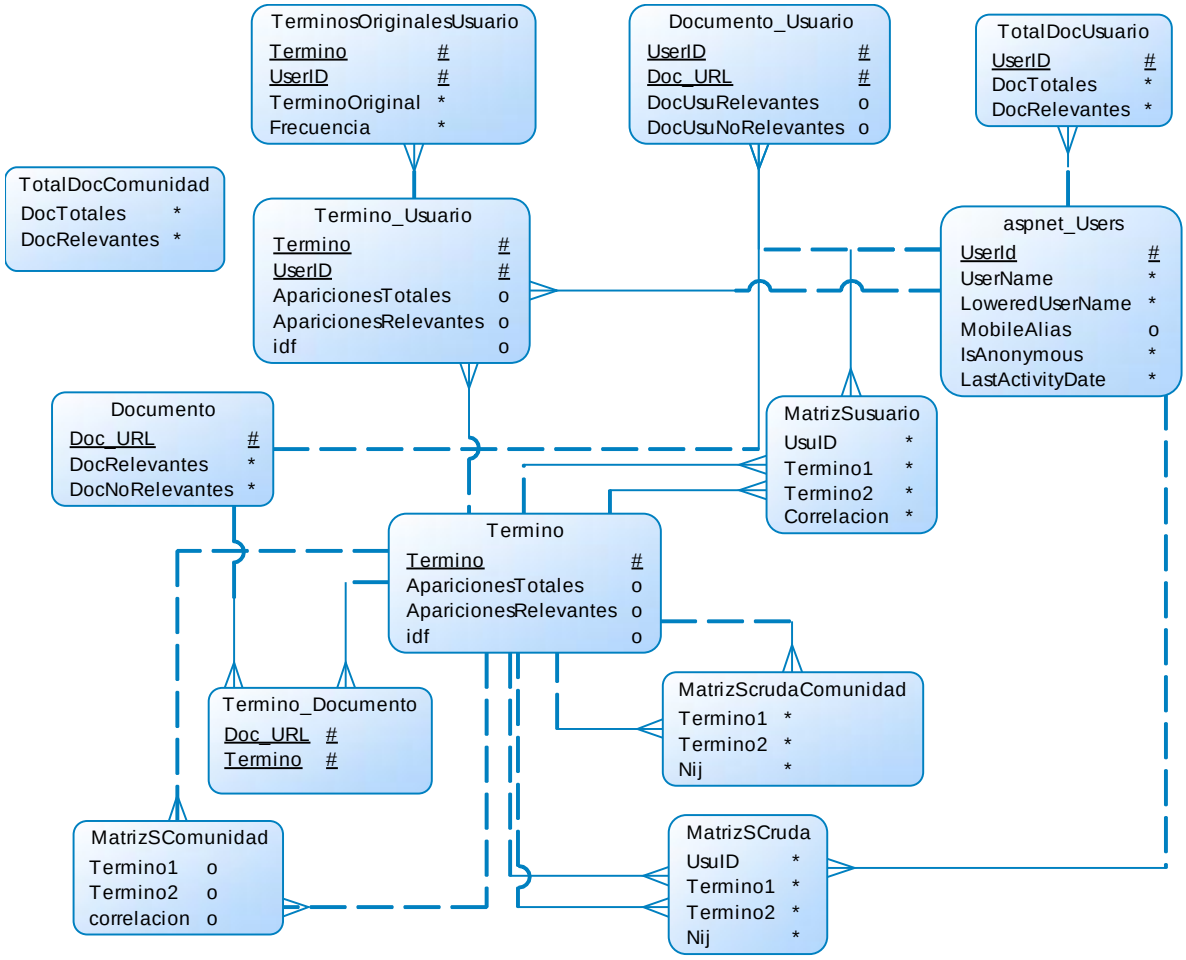


Figura 21. Estructura de base de datos relacional

Parte V – Evaluación del Modelo

El rendimiento, la funcionalidad, la usabilidad, entre otros aspectos se deben tener en cuenta en el momento de realizar la evaluación de un Sistema de Búsqueda Web [106-108]. Especialmente, se debe dar importancia a la satisfacción del usuario en cuanto a la relevancia de los resultados reportados por el sistema [109, 110] y su orden de presentación, porque en ocasiones sólo las primeras páginas recuperadas son leídas, sin tener en cuenta las demás, lo que puede generar que el usuario abandone la búsqueda sin obtener el resultado deseado [13].

En este capítulo se presenta el proceso de evaluación del modelo del meta buscador Web propuesto, donde se aplicaron medidas clásicas del área de la recuperación de la información como satisfacción del usuario y relevancia, a través de la Curva de Precision-Recuerdo, Mean Average Precision (MAP), Precisión en K resultados ordenados y el estadístico Kappa, comparando los resultados con los entregados originalmente por los buscadores tradicionales (Google y Bing).

14. Evaluación del Modelo

El proceso de evaluación se dividió en dos partes, la primera en relación a la precisión del modelo utilizando una colección cerrada de textos denominada CACM (Communications of ACM); y la segunda parte en relación con la precisión del modelo y su comparación con los resultados de búsqueda obtenidos por los buscadores tradicionales (Google y Bing).

Se realizaron un total cuatro (4) experimentos con la colección cerrada CACM, y once (11) pruebas con usuarios finales donde participaron como mínimo 14 estudiantes en cada prueba. Los resultados obtenidos en cada prueba fueron analizados y se realizaron algunos ajustes al modelo que se mencionan más adelante.

14.1 Evaluación del modelo usando la colección cerrada CACM

El conjunto de datos (dataset) usado para los cuatro experimentos fue la colección de textos CACM disponible en forma gratuita en http://ir.dcs.gla.ac.uk/resources/test_collections (Colecciones de prueba del Grupo de I+D en Recuperación de Información de la Universidad de Glasgow en Escocia, Reino Unido). Este dataset es una colección de títulos y resúmenes de artículos publicados en la revista "Communications of the ACM". En la colección se encuentran 3204 documentos y 64 consultas. Para cada consulta, asesores humanos leyeron todos los documentos y evaluaron cuáles de ellos son relevantes. En la presente investigación se tomaron las 52 consultas que tenían completos los juicios de relevancia en la colección.

Los 4 experimentos se hicieron con memoria de consulta, es decir, se simuló la ejecución de una consulta cinco veces, guardando el feedback de los resultados en el contexto del usuario. Para el **experimento 1**, la primera ejecución denominada "Básica" usa la similitud de Lucene (una variante de la similitud de cosenos); la segunda una expansión de la consulta basada en los documentos relevantes o no relevantes, que se presentaron en la consulta básica, a esta expansión se le denomina "expansión 1"; luego se realizó una "expansión 2" con los juicios de relevancia de expansión 1 y de la misma forma se realizó una expansión 3 y una expansión 4. Lo anterior con el objetivo de simular el proceso de refinación de las búsquedas que realiza un usuario cuando está buscando repetidamente sobre un tema especifico.

La Tabla 7 muestra los valores de precisión-recuerdo obtenidos en el primer experimento. Los resultados muestran mejoras consistentes, el resultado de la consulta básica usando Lucene, inicia en un 55,8% de precisión para un nivel de recuerdo de 10%, y decrece hasta un 7,2% cuando el nivel de recuerdo es del 100%. Luego en la expansión 1, se muestra una mejora apreciable que comienza en 75,1% de precisión para un nivel de recuerdo de 10%, y decrece hasta un 7,9% cuando el

nivel de recuerdo es del 100%. En la Figura 22 (a) se muestra la gráfica de los resultados del primer experimento, donde se puede observar que las curvas de precisión-recuerdo para todas las expansiones realizadas son superiores o iguales en todos los niveles de recuerdo en la consulta básica.

En el experimento 2: La primera ejecución denominada “Básica” usa la similitud de Lucene (una variante de la similitud de cosenos); la segunda una expansión de la consulta basada en la primera versión del modelo del meta buscador, donde se consideraba optimizar y controlar el tamaño de las matrices principales (que representan en el contexto del usuario y de la comunidad) donde sólo se almacenaban los términos cuya importancia relativa fuera mayor o igual a 0,5 y que tuvieran un valor de correlación con otros términos superior a un umbral, por ejemplo, sólo aquellas correlaciones superiores o iguales a 0,0125, denominada “expansión 1”; luego se realizó una “expansión 2” con los juicios de relevancia de expansión 1 y de la misma forma se realizó una expansión 3 y una expansión 4.

La Tabla 8 muestra los valores de precisión-recuerdo obtenidos en el segundo experimento. Los resultados son poco consistentes y en general deficientes. En la Figura 22 (b) se puede observar que las curvas de precisión-recuerdo para las expansiones realizadas no reflejan mejoras en todos los niveles de recuerdo con respecto a la consulta básica.

Después de analizar los resultados obtenidos se puede concluir que la estrategia de usar el contexto reducido no es apropiada, la razón, es que en el contexto del usuario se debe reflejar tanto la relación relevante, como la relación no relevante de todos los términos que el usuario haya evaluado (esto también se debe tener en cuenta para la comunidad). Debido a que estas matrices son utilizadas para realizar el proceso de expansión de la consulta, y se obtenían términos que no mejoraban la relevancia de los resultados obtenidos. Por lo tanto, se realizaron algunos ajustes al meta buscador:

a. El tamaño de la ventana que inicialmente era de cinco (5) términos, específicamente dos (2) términos a la izquierda y dos (2) a la derecha del término objetivo, fue ampliado a 13 términos. La razón es que cuando la ventana es muy pequeña, en el contexto del usuario, se almacenan pocas parejas de términos junto con sus correlaciones y se dejan por fuera correlaciones que podrían ser importantes. Este cambio incrementa el tiempo de procesamiento requerido para la calificación de los documentos, que de todas formas puede ser trabajado en el fondo sin que el usuario sea consciente de ello, pero con un costo alto en trabajo para el servidor.

b. En los algoritmos para actualizar y para calcular por primera vez la correlación entre términos se almacenan todos los valores, sin tener en cuenta el umbral mínimo de correlación que tengan las parejas de términos, ni la importancia relativa de los términos, sólo se valida los casos donde la correlación puede dar cero, es decir cuando $n_i = 0\ or\ n_j = 0\ or\ (n_i + n_j - n_{ij} = 0)$, con el fin de mantener las matriz S del usuario y de la comunidad con todas las relaciones relevantes y no relevantes de todos los términos.

Experimento 3: la primera ejecución denominada “Básica” usa la similitud de Lucene (una variante de la similitud de cosenos); la segunda una expansión de la consulta basada en la segunda versión del modelo del meta buscador (con los ajustes mencionados anteriormente) denominada “expansión 1”; luego se realizó una “expansión 2” con los juicios de relevancia de expansión 1 y de la misma forma se realizó una expansión 3 y una expansión 4. La Tabla 9 muestra los valores de precisión-recuerdo obtenidos en el segundo experimento y la Figura 22 (c) su gráfica asociada. Los resultados son poco consistentes y en general son deficientes, en el valor de la precisión de una iteración a otra no siempre mejora, adicionalmente entre más refinamientos la precisión disminuye notablemente.

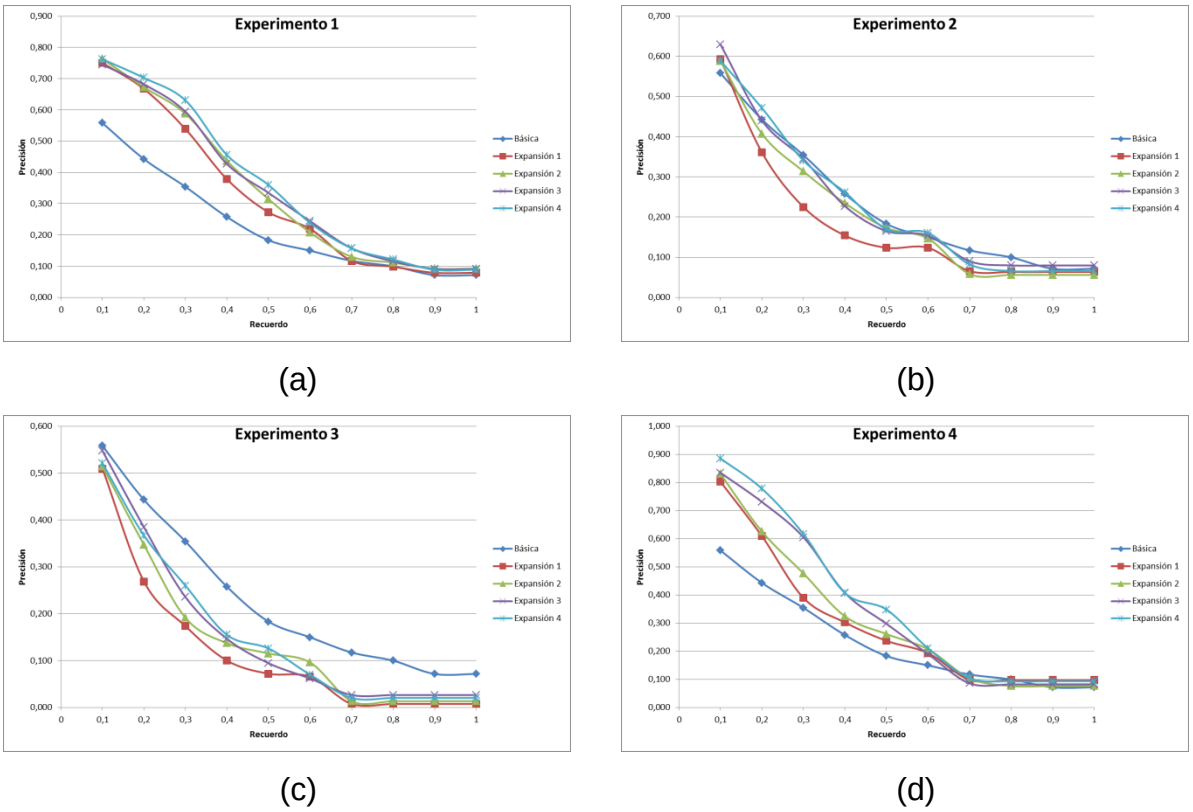


Figura 22. Curvas de precisión-recuerdo de los cuatro experimentos

	Precisión				
Recuerdo	Básica	Expansión 1	Expansión 2	Expansión 3	Expansión 4
0,1	0,558	0,751	0,763	0,745	0,762
0,2	0,443	0,667	0,673	0,682	0,703
0,3	0,354	0,539	0,588	0,594	0,630
0,4	0,257	0,378	0,437	0,427	0,456
0,5	0,183	0,272	0,315	0,334	0,360
0,6	0,150	0,220	0,208	0,244	0,239
0,7	0,117	0,117	0,130	0,157	0,157
0,8	0,100	0,098	0,112	0,116	0,122
0,9	0,072	0,079	0,092	0,091	0,089
1	0,072	0,079	0,092	0,091	0,089

Tabla 7. Valores de precisión - recuerdo para el experimento 1

	Precisión				
Recuerdo	Básica	Expansión 1	Expansión 2	Expansión 3	Expansión 4
0,1	0,558	0,593	0,589	0,630	0,589
0,2	0,443	0,361	0,408	0,440	0,471
0,3	0,354	0,225	0,314	0,345	0,341
0,4	0,257	0,154	0,235	0,227	0,262
0,5	0,183	0,124	0,173	0,165	0,170
0,6	0,150	0,124	0,147	0,154	0,161
0,7	0,117	0,066	0,058	0,090	0,082
0,8	0,100	0,064	0,056	0,080	0,066
0,9	0,072	0,064	0,056	0,080	0,066
1	0,072	0,064	0,056	0,080	0,066

Tabla 8. Valores de precisión-recuerdo para el experimento 2

	Precisión				
Recuerdo	Básica	Expansión 1	Expansión 2	Expansión 3	Expansión 4
0,1	0,558	0,509	0,514	0,547	0,521
0,2	0,443	0,269	0,347	0,384	0,367
0,3	0,354	0,173	0,191	0,235	0,260
0,4	0,257	0,100	0,137	0,146	0,155
0,5	0,183	0,072	0,115	0,094	0,126
0,6	0,150	0,066	0,096	0,062	0,070
0,7	0,117	0,008	0,013	0,027	0,020
0,8	0,100	0,008	0,013	0,027	0,020
0,9	0,072	0,008	0,013	0,027	0,020
1	0,072	0,008	0,013	0,027	0,020

Tabla 9. Valores de precisión-recuerdo para el experimento 3

Después de analizar los resultados obtenidos, se revisó el proceso correspondiente al filtrado de información, donde inicialmente se obtenía una matriz de co-ocurrencia temporal St que estaba conformada por los términos de la matriz S, más los términos de la consulta que estaban en la matriz S cruda y por los términos de la consulta que estaban en los documentos recuperados, además la relación que tenían cada uno de los términos entre ellos; basado en esta matriz se realizaban algunos cálculos utilizando la función de ranking (Figura 11) para establecer cuáles documentos eran

más relevantes para el usuario, de acuerdo a la consulta y a su contexto. Este proceso era demasiado costoso en términos de rendimiento para el sistema y finalmente no se lograba tener mejoras significativas en los resultados obtenidos, por lo tanto se decidió trabajar únicamente con los términos de la Matriz S que representa el contexto actual del usuario (que después de realizar el primer ajuste, cuenta con los todos los términos relevantes y no relevantes que haya evaluado).

Experimento 4: la primera ejecución denominada "Básica" usa la similitud de Lucene (una variante de la similitud de cosenos); la segunda una expansión de la consulta basada en la tercera versión del modelo del meta buscador (versión definitiva explicada en la parte III del presente documento) denominada "expansión 1"; luego se realizó una "expansión 2" con los juicios de relevancia de expansión 1 y de la misma forma se realizó una expansión 3 y una expansión 4.

La Tabla 10 muestra los valores de precisión-recuerdo obtenidos en el cuarto experimento. Los resultados muestran mejoras consistentes, el resultado de la consulta básica usando Lucene, inicia en un 55,8% de precisión para un nivel de recuerdo de 10%, y decrece hasta un 7,2% cuando el nivel de recuerdo es del 100%. Luego en la expansión 1, se muestra una mejora apreciable que comienza en 80,3% de precisión para un nivel de recuerdo de 10%, y decrece hasta un 9,7% cuando el nivel de recuerdo es del 100%. En la Figura 22 (d) se muestra la gráfica de los resultados del cuarto experimento, donde se puede observar que las curvas de precisión-recuerdo en cada iteración, en general van mejorando o se conservan, en todos los niveles de recuerdo con respecto a la iteración anterior.

	Precisión				
Recuerdo	Básica	Expansión 1	Expansión 2	Expansión 3	Expansión 4
0,1	0,558	0,803	0,830	0,835	0,885
0,2	0,443	0,609	0,626	0,731	0,778
0,3	0,354	0,389	0,476	0,605	0,617
0,4	0,257	0,302	0,324	0,407	0,407
0,5	0,183	0,237	0,261	0,297	0,348

0,6	0,150	0,193	0,207	0,190	0,210
0,7	0,117	0,098	0,104	0,086	0,104
0,8	0,100	0,097	0,076	0,082	0,093
0,9	0,072	0,097	0,076	0,082	0,093
1	0,072	0,097	0,076	0,082	0,093

Tabla 10. Valores de precisión-recuerdo para el experimento 4

14.1.1 Comparación contra Rocchio

Con fin de dar validez a los resultados obtenidos en el cuarto experimento se calculó la curva precisión-recuerdo para el algoritmo de Rocchio, que propone la Ecuación 14 para generar la consulta expandida. Donde q es la consulta inicialmente digitada por el usuario, R es un conjunto de documentos relevantes, R' es un conjunto de documentos no relevantes, α, β y γ son constantes de afinación del modelo y q_e es la consulta expandida [7, 13].

$$q_e = \alpha \times q + \frac{\beta}{|R|}\sum_{d \in R} d - \frac{\gamma}{|R'|}\sum_{d \in R'} d$$

Ecuación 14. Formula Rocchio

La primera ejecución denominada "Básica" usa la similitud de Lucene (una variante de la similitud de cosenos); la segunda una expansión de la consulta basada en el algoritmo de Rocchio denominada "expansión 1"; luego se realizó una "expansión 2" con los juicios de relevancia de expansión 1 y de la misma forma se realizó una expansión 3 y una expansión 4. La Tabla 11 muestra los valores de precisión-recuerdo obtenidos el experimento con Rocchio.

Los resultados muestran mejoras consistentes, el resultado de la consulta básica usando Lucene, inicia en un 55,8% de precisión para un nivel de recuerdo de 10%, y decrece hasta un 7,2% cuando el nivel de recuerdo es del 100%. Luego en la expansión 1, se muestra una mejora bastante considerable que comienza en 93,4%

de precisión para un nivel de recuerdo de 10%, y decrece hasta un 15,9% cuando el nivel de recuerdo es del 100%. En la Figura 23 se muestra la gráfica de los resultados con Rocchio, donde se puede observar que las curvas de precisión-recuerdo en cada iteración, en general van mejorando o se conservan, en todos los niveles de recuerdo con respecto a la iteración anterior.

	Precisión				
Recall	Básica	Expansión 1	Expansión 2	Expansión 3	Expansión 4
0,1	0,558	0,934	0,981	0,981	0,981
0,2	0,443	0,825	0,912	0,912	0,912
0,3	0,354	0,716	0,800	0,822	0,826
0,4	0,257	0,557	0,650	0,677	0,673
0,5	0,183	0,419	0,478	0,507	0,512
0,6	0,150	0,320	0,366	0,367	0,374
0,7	0,117	0,229	0,221	0,218	0,218
0,8	0,100	0,178	0,170	0,170	0,170
0,9	0,072	0,159	0,150	0,150	0,150
1	0,072	0,159	0,150	0,150	0,150

Tabla 11. Valores de precisión-recuerdo para Rocchio

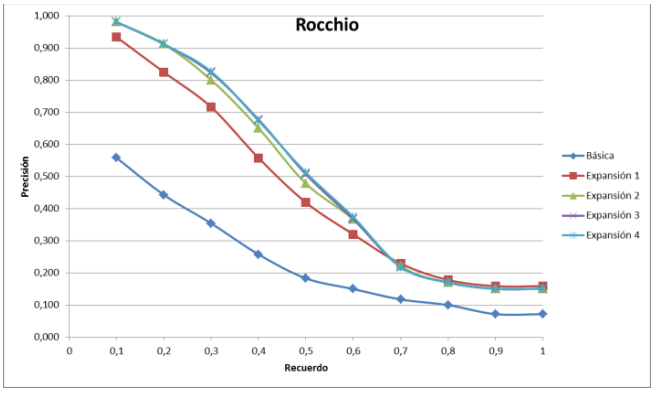


Figura 23. Curva precisión-recuerdo de Rocchio

Comparando los resultados de la Tabla 10 con los de la Tabla 11, correspondientes a aplicar expansiones con el modelo del meta buscador propuesto y el algoritmo de Rocchio respectivamente, en la expansión 1, Rocchio comienza en 93,4% de

precisión para un nivel de recuerdo de 10%, mientras que el meta buscador propuesto, en este nivel de recuerdo, comienza con 80,3% (una diferencia de 13,1%). Cuando se hace la última expansión el meta buscador alcanza un 88,5% de precisión para un nivel de recuerdo de 10% y Rocchio un 98,1% (una diferencia de 9,6%). Por lo tanto se puede notar que a pesar de obtener resultados consistentes con el modelo propuesto, los resultados entregado por Rocchio son mejores y de menor costo computacionalmente hablando.

14.2 Evaluación con Usuarios

Se realizaron pruebas con estudiantes de diferentes semestres del Programa de Ingeniería de Sistemas de la Universidad del Cauca. A continuación se muestran las 2 últimas pruebas realizadas sobre MyBestMetaWebSearch, teniendo en cuenta que todas se hicieron de la misma forma, para cada prueba se seleccionaron 3 consultas, luego para cada consulta se realizaron 3 iteraciones de búsqueda y para cada iteración se evaluaron los primeros 8 documentos, es decir, se realizó la primera consulta (primera iteración) y se evaluaron los primeros 8 documentos, luego se repitió la consulta (segunda iteración) y se realizó nuevamente la evaluación de los primeros 8 documentos, esa misma consulta se digitó por tercera vez (tercera iteración) y se realizó nuevamente la evaluación de los primeros 8 documentos, lo mismo se hizo para la segunda y tercera consulta.

14.3 Prueba 1

Esta prueba se realizó con catorce (14) estudiantes del programa de Ingeniería de Sistemas de noveno semestre de la Universidad del Cauca, pertenecientes al curso “Proyecto I”. Quienes fueron divididos en dos grupos, con igual cantidad de estudiantes, el primer grupo realizó la evaluación de MyBestMetaWebSearch, y el segundo la evaluación de los buscadores tradicionales Google y Bing. Las consultas utilizadas fueron las siguientes:

Consulta 1: use case diagram
Consulta 2: include use case diagram
Consulta 3 extend use case diagram

Los estudiantes evaluaron los primeros 8 documentos recuperados, en cada una de las tres iteraciones que se hicieron con la primer consulta, como relevante (R), no relevante (N), e inaccesible (X), cuando el documento Web no pudo ser visto por el usuario. Este mismo proceso se realizó para la segunda y tercer consulta. A continuación se muestra el cálculo y la comparación de los resultados encontrados en cada iteración, al final se sacan las conclusiones de los resultados de la prueba.

14.3.1 Prueba 1: Precisión en K Resultados y MAP

Se tomaron los resultados de las evaluaciones de la primer consulta se sumaron todas las evaluaciones iguales para cada uno de los documentos evaluados, se sacaron totales y se calculó la exactitud, la precisión y la precisión media como se muestra en la Tabla 12. Este proceso también se hizo para la segunda y tercer consulta, como resultado se obtuvieron la Tabla 13 y la Tabla 14 respectivamente.

Iteración	Resultado	R	N	X	Total	Exactitud	Precisión	Precisión media (MAP)
1	1	5	1	1	7	71,4%	71,4%	65,7%
	2	4	3	0	7	57,1%	64,3%	
	3	5	1	1	7	71,4%	64,3%	
	4	5	2	0	7	71,4%	71,4%	
	5	5	2	0	7	71,4%	68,6%	
	6	2	5	0	7	28,6%	61,9%	
	7	4	3	0	7	57,1%	61,2%	
	8	5	2	0	7	71,4%	62,5%	
2	1	2	4	1	7	28,6%	28,6%	53,1%
	2	4	3	0	7	57,1%	42,9%	
	3	5	1	1	7	71,4%	64,3%	
	4	4	3	0	7	57,1%	64,3%	
	5	5	2	0	7	71,4%	57,1%	
	6	4	3	0	7	57,1%	57,1%	

	7	3	4	0	7	42,9%	55,1%	
	8	4	3	0	7	57,1%	55,4%	
3	1	4	2	1	7	57,1%	57,1%	63,0%
	2	4	2	0	6	66,7%	61,5%	
	3	5	2	0	7	71,4%	69,2%	
	4	4	3	0	7	57,1%	64,3%	
	5	5	2	0	7	71,4%	64,7%	
	6	4	1	2	7	57,1%	63,4%	
	7	3	3	1	7	42,9%	60,4%	
	8	6	0	1	7	85,7%	63,6%	

Tabla 12. Prueba 1 “use case diagram”– Estadísticas

Para la primer consulta (ver Tabla 12) en su primera iteración la precisión oscila entre 61,2% y 71,4%, obteniendo una precisión media de 65,7%,para la segunda iteración la precisión oscila entre 28,6%y 64,3%, obteniendo una precisión media de 53,1%y finalmente para la tercera iteración el valor de la precisión oscila entre 57,1% y 69,2%, obteniendo una precisión media de 63,0%, se esperaría una mejora de la precisión de una iteración a otra iteración, gracias al manejo del contexto del usuario y el filtrado colaborativo entre usuarios, sin embargo se puede notar que en este caso particular la precisión comenzó a disminuir gradualmente quedando en la segunda iteración 12,61% por debajo de la precisión de la consulta base (primera iteración), luego para la tercera iteración la precisión sube 9,9% con respecto a la segunda.

Para la segunda consulta (ver Tabla 13) en su primera iteración la precisión oscila entre 57,1% y 85,7%, obteniendo una precisión media de 70,2%, para la segunda iteración la precisión oscila entre 50,0% y 85,7%, obteniendo una precisión media de 61,8% y finalmente para la tercera iteración el valor de la precisión oscila entre 57,1% y 100,0%, obteniendo una precisión media de 68,9%, en este caso particular la precisión en la segunda iteración disminuye 8,4% con respecto a la primera , sin embargo en la iteración 3 la precisión mejora en 7,1% con respecto a la segunda iteración, sin alcanzar a superar el resultado obtenido inicialmente, de este modo se

nota una disminución de la precisión en 1,3% con respecto a la primera iteración Figura 24 (b).

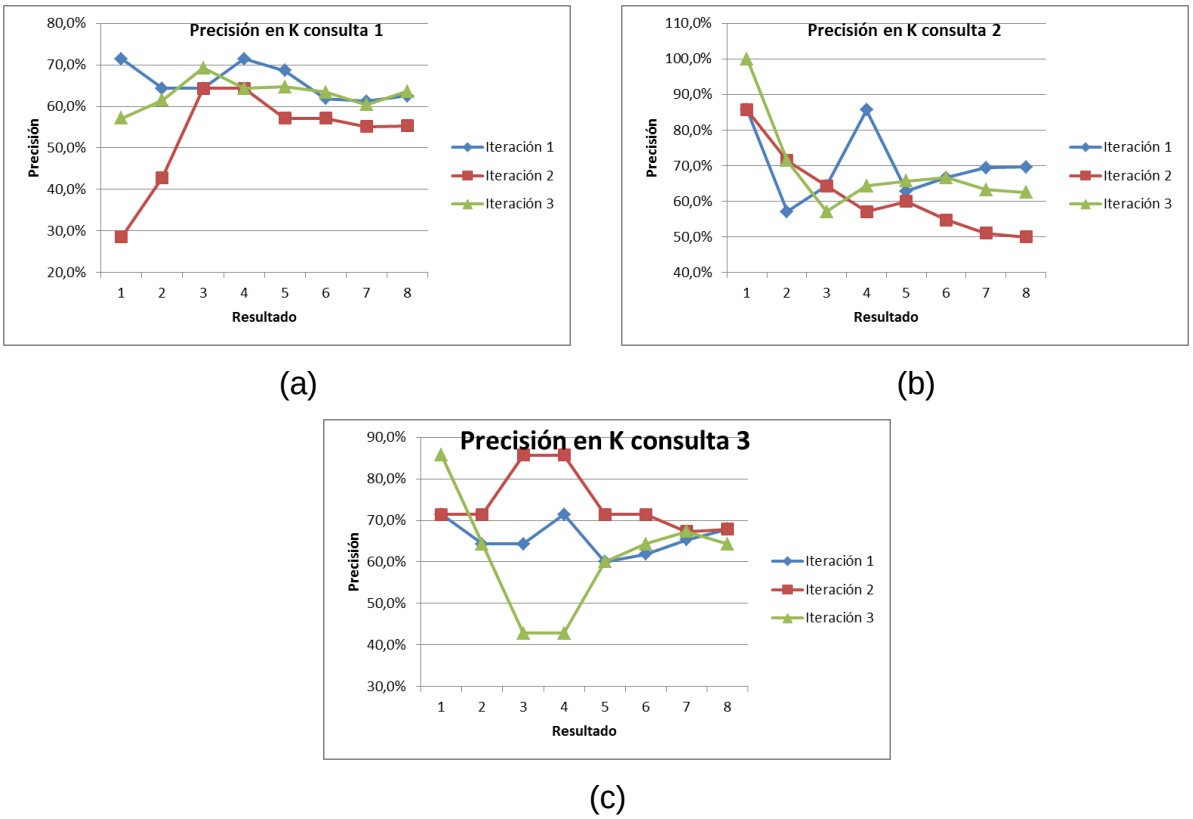


Figura 24. Gráfica de la precisión para las consultas tres consultas

Iteración	Resultado	R	N	X	Total	Exactitud	Precisión	Precisión media (MAP)
1	1	6	1	0	7	85,7%	85,7%	70,2%
	2	2	5	0	7	28,6%	57,1%	
	3	7	0	0	7	100,0%	64,3%	
	4	5	2	0	7	71,4%	85,7%	
	5	2	5	0	7	28,6%	62,9%	
	6	6	1	0	7	85,7%	66,7%	
	7	6	0	1	7	85,7%	69,4%	
	8	5	2	0	7	71,4%	69,6%	
2	1	6	1	0	7	85,7%	85,7%	61,8%
	2	4	3	0	7	57,1%	71,4%	

	3	5	2	0	7	71,4%	64,3%	
	4	3	2	2	7	42,9%	57,1%	
	5	3	2	2	7	42,9%	60,0%	
	6	2	4	1	7	28,6%	54,8%	
	7	2	4	1	7	28,6%	51,0%	
	8	3	3	1	7	42,9%	50,0%	
3	1	7	0	0	7	100,0%	100,0%	68,9%
	2	3	4	0	7	42,9%	71,4%	
	3	5	2	0	7	71,4%	57,1%	
	4	4	2	1	7	57,1%	64,3%	
	5	4	2	1	7	57,1%	65,7%	
	6	5	2	0	7	71,4%	66,7%	
	7	3	4	0	7	42,9%	63,3%	
	8	4	3	0	7	57,1%	62,5%	

Tabla 13. Prueba 1 “include use case diagram”– Estadísticas

Para la tercera consulta (ver Tabla 14) en su primera iteración la precisión oscila entre 60,0% y 71,4%, obteniendo una precisión media de 65,8%, para la segunda iteración la precisión oscila entre 67,3% y 71,4%obteniendo una precisión media de 74,0% y finalmente para la tercera iteración el valor de la precisión oscila entre 42,9% y 85,7%, obteniendo una precisión media de 61,5%, los resultados obtenidos son similares a los presentados con la consulta 2, se refleja un comportamiento oscilatorio de la precisión de una iteración a otra y el resultado de la precisión en la ultima iteración queda por debajo de la primera iteración Figura 24 (c)

Iteración	Resultado	R	N	X	Total	Exactitud	Precisión	Precisión media (MAP)
1	1	5	2	0	7	71,4%	71,4%	65,8%
	2	4	3	0	7	57,1%	64,3%	
	3	5	2	0	7	71,4%	64,3%	
	4	5	2	0	7	71,4%	71,4%	
	5	2	5	0	7	28,6%	60,0%	
	6	5	2	0	7	71,4%	61,9%	
	7	6	1	0	7	85,7%	65,3%	
	8	6	1	0	7	85,7%	67,9%	

2	1	5	2	0	7	71,4%	71,4%	74,0%
	2	5	2	0	7	71,4%	71,4%	
	3	7	0	0	7	100,0%	85,7%	
	4	5	2	0	7	71,4%	85,7%	
	5	3	4	0	7	42,9%	71,4%	
	6	5	2	0	7	71,4%	71,4%	
	7	3	4	0	7	42,9%	67,3%	
	8	5	2	0	7	71,4%	67,9%	
3	1	6	1	0	7	85,7%	85,7%	61,5%
	2	3	4	0	7	42,9%	64,3%	
	3	3	4	0	7	42,9%	42,9%	
	4	3	4	0	7	42,9%	42,9%	
	5	6	1	0	7	85,7%	60,0%	
	6	6	1	0	7	85,7%	64,3%	
	7	6	1	0	7	85,7%	67,3%	
	8	3	4	0	7	42,9%	64,3%	

Tabla 14. Prueba 1 "extend use case diagram"– Estadísticas

14.3.2 Prueba 1: Índice Kappa

A continuación se presentan los resultados de los cálculos realizados para medir el nivel de concordancia de los usuarios con respecto a las evaluaciones hechas sobre cada uno de los documentos, cálculo que sólo se puede hacer para la primera iteración de la primer consulta en cada una de las pruebas, debido a que para las demás iteraciones los documentos recuperados no son los mismos. El índice usado para medir la concordancia entre los juicios de los usuarios se denomina Kappa de Fleiss (ver Tabla 15)

Consulta	RESULTADO	R	N	X	TOTAL	Pi
1	1	5	1	1	7	0,48
	2	4	3	0	7	0,43
	3	5	1	1	7	0,48
	4	5	2	0	7	0,52
	5	5	2	0	7	0,52
	6	2	5	0	7	0,52
	7	4	3	0	7	0,43
	8	5	2	0	7	0,52

TOTAL	35	19	2	56	3,90
Pr	0,63	0,34	0,04		0,49
Pr^2	0,39	0,12	0,00	0,51	Pe-

Kappa de Fleiss: -0,04 Poor Discordance

Tabla 15. Prueba 1 – Kappa de Fleiss. Consulta 1

El valor de Kappa de Fleiss es de -0,04 lo que quiere decir que hubo un desacuerdo "pobre" entre los usuarios al evaluar los 8 documentos en la primera iteración de la primer consulta (ver Tabla 12).

Por último se procesaron los resultados de cada una de las consultas y se calculó la precisión total de la prueba, los resultados obtenidos se aprecian en la Tabla 16. Donde la precisión total es la precisión media de cada uno de los documentos para cada consulta y la precisión media total es la precisión media de los 8 documentos por cada iteración.

Consulta		1	2	3		
Iteración	Resultado	Precisión	Precisión	Precisión	Precisión total	Precisión media (MAP)
1	1	71,4%	85,7%	71,4%	76,2%	67,2%
	2	64,3%	57,1%	64,3%	61,9%	
	3	64,3%	64,3%	64,3%	64,3%	
	4	71,4%	85,7%	71,4%	76,2%	
	5	68,6%	62,9%	60,0%	63,8%	
	6	61,9%	66,7%	61,9%	63,5%	
	7	61,2%	69,4%	65,3%	65,3%	
	8	62,5%	69,6%	67,9%	66,7%	
2	1	28,6%	85,7%	71,4%	61,9%	63,0%
	2	42,9%	71,4%	71,4%	61,9%	
	3	64,3%	64,3%	85,7%	71,4%	
	4	64,3%	57,1%	85,7%	69,0%	
	5	57,1%	60,0%	71,4%	62,9%	
	6	57,1%	54,8%	71,4%	61,1%	
	7	55,1%	51,0%	67,3%	57,8%	
	8	55,4%	50,0%	67,9%	57,7%	

3	1	57,1%	100,0%	85,7%	81,0%	64,5%
	2	61,5%	71,4%	64,3%	65,8%	
	3	69,2%	57,1%	42,9%	56,4%	
	4	64,3%	64,3%	42,9%	57,1%	
	5	64,7%	65,7%	60,0%	63,5%	
	6	63,4%	66,7%	64,3%	64,8%	
	7	60,4%	63,3%	67,3%	63,7%	
	8	63,6%	62,5%	64,3%	63,5%	

Tabla 16. Prueba 1 – Precisión Media (MAP) Las 3 consultas - Estadísticas

Para las tres consultas en la primera iteración (ver Tabla 16) la precisión oscila entre 61,9% y 76,2%, obteniendo una precisión media de 67,2%, para la segunda iteración la precisión oscila entre 57,7% y 71,4% obteniendo una precisión media de 63,0% y finalmente para la tercera iteración el valor de la precisión oscila entre 56,4% y 81,0% obteniendo una precisión media de 64,5%, lo que refleja una disminución de la precisión en 4,2% entre la iteración 1 y la 2, luego la precisión se restablece aumentando 1,5%, de este modo entre la primera iteración y la tercera se pierde un 2,7% de precisión, a medida que el contexto del usuario y el filtrado colaborativo de la comunidad se aplican, mas adelante se da una explicación de los resultados obtenidos.

14.3.3 Prueba 1: Comparación con otros Buscadores

Como se mencionó anteriormente, los siete (7) estudiantes restantes del curso de “Proyecto I”, realizaron la prueba utilizando los buscadores tradicionales Google y Bing, usaron las mismas consultas ingresadas en MyBestMetaWebSearch: consulta 1: “use case diagram”, consulta 2: “include use case diagram”; consulta 3: “extend use case diagram”. Se tomaron los resultados de las evaluaciones de las tres consulta realizadas en Google, se sumaron todas las evaluaciones iguales para cada uno de los documentos evaluados, se sacaron totales y se calculó la exactitud, la precisión y la precisión media como se muestra en la Tabla 17 Este mismo proceso se hizo para las consultas realizadas en Bing y como se muestra en la Tabla 18.

Consulta	RESULTADO	R	N	X	TOTAL	EXACTITUD	PRECISION	PRECISION MEDIA
1	1	5	2	0	7	71,4%	71,4%	69,7%
	2	7	0	0	7	100,0%	85,7%	
	3	4	0	3	7	57,1%	78,6%	
	4	2	1	4	7	28,6%	42,9%	
	5	6	1	0	7	85,7%	68,6%	
	6	6	0	1	7	85,7%	71,4%	
	7	4	1	2	7	57,1%	69,4%	
	8	5	2	0	7	71,4%	69,6%	
2	1	4	3	0	7	57,1%	57,1%	56,0%
	2	4	0	3	7	57,1%	57,1%	
	3	2	2	3	7	28,6%	42,9%	
	4	6	1	0	7	85,7%	57,1%	
	5	4	3	0	7	57,1%	57,1%	
	6	6	0	1	7	85,7%	61,9%	
	7	2	2	3	7	28,6%	57,1%	
	8	4	3	0	7	57,1%	57,1%	
3	1	4	1	2	7	57,1%	57,1%	59,3%
	2	4	2	1	7	57,1%	57,1%	
	3	4	1	2	7	57,1%	57,1%	
	4	5	2	0	7	71,4%	64,3%	
	5	3	3	1	7	42,9%	57,1%	
	6	6	1	0	7	85,7%	61,9%	
	7	3	4	0	7	42,9%	59,2%	
	8	5	2	0	7	71,4%	60,7%	

Tabla 17. Prueba 1: Tres consultas en Google – Estadísticas

Usando el buscador Google para la primer consulta (ver Tabla 17) la precisión oscila entre 42,9% y 85,9%, obteniendo una precisión media de 69,7%, para la segunda consulta la precisión oscila entre 42,9% y 61,9%, obteniendo una precisión media de 56,0%y finalmente para la tercera consulta el valor de la precisión oscila entre 57,1% y 64,3%, obteniendo una precisión media de 59 ,3%.

Consulta	RESULTADO	R	N	X	TOTAL	EXACTITUD	PRECISION	PRECISION MEDIA
1	1	6	1	0	7	85,7%	85,7%	57,9%
	2	5	0	2	7	71,4%	78,6%	
	3	2	1	4	7	28,6%	50,0%	
	4	3	0	4	7	42,9%	35,7%	
	5	5	2	0	7	71,4%	60,0%	
	6	1	4	2	7	14,3%	52,4%	
	7	3	3	1	7	42,9%	51,0%	
	8	3	1	3	7	42,9%	50,0%	
2	1	5	2	0	7	71,4%	71,4%	71,7%
	2	4	2	1	7	57,1%	64,3%	
	3	6	0	1	7	85,7%	71,4%	
	4	7	0	0	7	100,0%	92,9%	
	5	4	2	1	7	57,1%	74,3%	
	6	4	0	3	7	57,1%	71,4%	
	7	2	4	1	7	28,6%	65,3%	
	8	3	3	1	7	42,9%	62,5%	
3	1	4	2	1	7	57,1%	57,1%	63,9%
	2	5	0	2	7	71,4%	64,3%	
	3	4	2	1	7	57,1%	64,3%	
	4	6	1	0	7	85,7%	71,4%	
	5	4	2	1	7	57,1%	65,7%	
	6	4	3	0	7	57,1%	64,3%	
	7	4	3	0	7	57,1%	63,3%	
	8	3	4	0	7	42,9%	60,7%	

Tabla 18. Prueba 1: Tres consultas en Bing – Estadísticas

Usando Bing para la primer consulta (ver Tabla 18) la precisión oscila entre 35,7% y 85,7%, obteniendo una precisión media de 57,9%, para la segunda consulta la precisión oscila entre 62,5% y 92,9%, obteniendo una precisión media de 71,7% y finalmente para la tercera consulta el valor de la precisión oscila entre 57,1% y 71,4%, obteniendo una precisión media de 63,9%.

De los resultados obtenidos en la prueba 1 se puede notar que MybestMetaWebSearch reporta para la primera consulta una precisión media menor 2,5% que Google, pero 9,3% por encima de Bing. Para la segunda consulta la precisión media de MybestMetaWebSearch esta 7% por encima de Google, y 8,7% menor que la Bing y finalmente, para la tercera consulta la precisión media de MybestMetaWebSearch esta por encima 5,2 % que Google y 0,6% de Bing (ver Tabla 19).

Precisión media (MAP)			
Consulta	Google	Bing	MyBest Search
1	69,7%	57,9%	67,2%
2	56,0%	71,7%	63,0%
3	59,3%	63,9%	64,5%

Tabla 19. Prueba 1- Comparación de la precisión media frente a buscadores tradicionales

14.4 Prueba 2

Esta prueba se realizó con dieciocho (18) estudiantes del programa de Ingeniería de Sistemas de sexto semestre de la Universidad del Cauca, pertenecientes al curso “Estructuras del Lenguaje”, quienes fueron divididos en dos grupos, el primer grupo conformado por 8 estudiantes seleccionados al azar, realizaron la evaluación de MyBestMetaWebSearch, y el segundo grupo de 10 estudiantes realizaron la evaluación de los buscadores tradicionales Google y Bing. Se utilizaron las siguientes consultas. Consulta 1: “Class diagram”; Consulta 2: “Class diagram association” y Consulta 3: “Class diagram inheritance”.

14.4.1 Prueba 2: Precisión en K Resultados y MAP

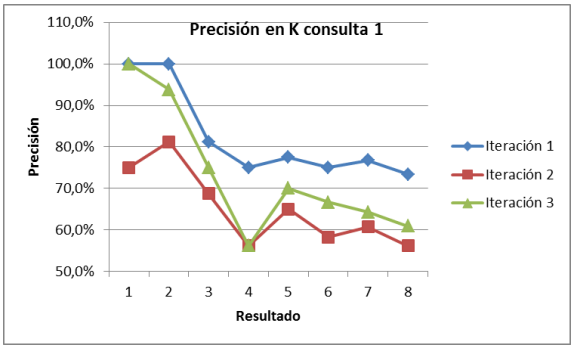
Como se mencionó inicialmente, todas las pruebas se realizaron de la misma forma, a continuación se muestra de manera resumida los resultados de la evaluación de obtenidos para cada consulta en sus tres iteraciones (ver Tabla 20)

Iteración	# Documento	Consulta 1		Consulta 2		Consulta 3	
		Exactitud	Precisión	Exactitud	Precisión	Exactitud	Precisión
1	1	100,0%	100,0%	87,5%	87,5%	100,0%	100,0%
	2	100,0%	100,0%	62,5%	75,0%	75,0%	87,5%
	3	62,5%	81,3%	62,5%	62,5%	62,5%	68,8%
	4	87,5%	75,0%	50,0%	56,3%	62,5%	62,5%
	5	37,5%	77,5%	37,5%	60,0%	75,0%	75,0%
	6	62,5%	75,0%	50,0%	58,3%	87,5%	77,1%
	7	87,5%	76,8%	37,5%	55,4%	75,0%	76,8%
	8	50,0%	73,4%	37,5%	53,1%	0,0%	67,2%
2	1	75,0%	75,0%	75,0%	75,0%	100,0%	100,0%
	2	87,5%	81,3%	50,0%	62,5%	75,0%	87,5%
	3	50,0%	68,8%	37,5%	43,8%	75,0%	75,0%
	4	62,5%	56,3%	37,5%	37,5%	75,0%	75,0%
	5	50,0%	65,0%	50,0%	50,0%	62,5%	77,5%
	6	25,0%	58,3%	12,5%	43,8%	87,5%	79,2%
	7	75,0%	60,7%	25,0%	41,1%	62,5%	76,8%
	8	25,0%	56,3%	50,0%	42,2%	25,0%	70,3%
3	1	100,0%	100,0%	87,5%	87,5%	100,0%	100,0%
	2	87,5%	93,8%	75,0%	81,3%	62,5%	81,3%
	3	62,5%	75,0%	12,5%	43,8%	87,5%	75,0%
	4	50,0%	56,3%	25,0%	18,8%	62,5%	75,0%
	5	50,0%	70,0%	50,0%	50,0%	87,5%	80,0%
	6	50,0%	66,7%	37,5%	47,9%	75,0%	79,2%
	7	50,0%	64,3%	37,5%	46,4%	100,0%	82,1%
	8	37,5%	60,9%	50,0%	46,9%	37,5%	76,6%

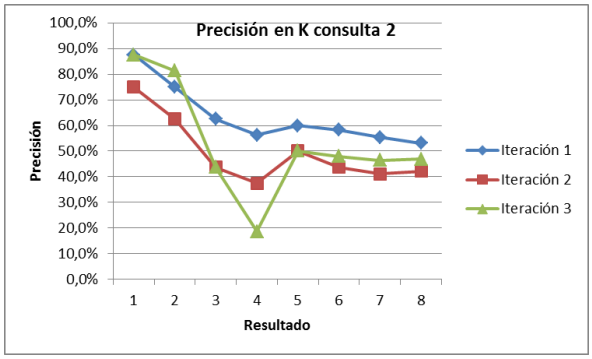
Tabla 20. Prueba 2. Precisión en K de las iteraciones de las 3 consultas

Para la primer consulta “Class diagram” (ver Tabla 20) en su primera iteración la precisión oscila entre 73,4% y 100,0%, para la segunda iteración la precisión oscila entre 56,3% y 81,3%, y para la tercera iteración el valor de la precisión oscila entre 56,3% y 100,0% ver Figura 25. Precisión en K de las 3 consultaFigura 25 (a). Para la segunda consulta “Class diagram association” en su primera iteración la precisión oscila entre 53,1% y 87,5%, para la segunda iteración la precisión oscila entre 37,5% y 75,0% y para la tercera iteración el valor de la precisión oscila entre 18,8% y 87,5 ver Figura 25 (b). Para la tercera consulta “Class diagram inheritance” en su primera iteración la precisión oscila entre 62,5% y 100,0%, para la segunda iteración la

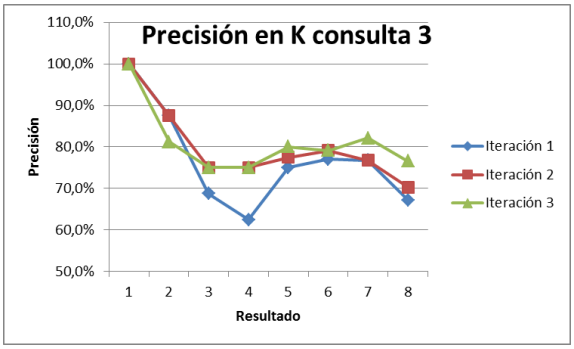
precisión oscila entre 70,3% y 100,0%, y para la tercera iteración el valor de la precisión oscila entre 75,0% y 100,0 (ver Figura 25 (c).



(a)



(b)



(c)

Figura 25. Precisión en K de las 3 consultas

En la Tabla 21 se puede observar la precisión media (MAP) de las tres consultas en cada iteración. Para la consulta 1, la precisión media baja en la segunda iteración 17,2% con respecto a la primera, luego en la tercera iteración la precisión media sube 8,2% con respecto a la segunda iteración, lo que indica que no alcanza la precisión media inicial. El mismo comportamiento se nota con respecto a la consulta 2, donde se baja la precisión media en 14 % con respecto a la primera iteración, luego la precisión media sube 3,3% en la tercera iteración con respecto a la segunda. Finalmente, en la consulta 3 se pueden observar una mejora (poco significativa) en la precisión de una iteración a otra.

Iteración	Consulta 1	Consulta 2	Consulta 3
	Precisión media (MAP)	Precisión media (MAP)	Precisión media (MAP)
1	82,4%	63,5%	76,9%
2	65,2%	49,5%	80,2%
3	73,4%	52,8%	81,1%

Tabla 21. Prueba 2 Precisión media (MAP) de las iteraciones en tres consultas

14.4.2 Prueba 2: Índice Kappa

Para medir el nivel de concordancia de los usuarios con respecto a las evaluaciones hechas sobre cada uno de los documentos, se usó el índice estadístico Kappa de Fleiss, teniendo en cuenta que el cálculo que sólo se puede hacer para la primera iteración de la primer consulta en cada una de las pruebas, debido a que para las demás iteraciones los documentos recuperados no son los mismos.

Consulta	RESULTADO	R	N	X	TOTAL	Pi
1	1	8	0	0	8	1,00
	2	8	0	0	8	1,00
	3	5	3	0	8	0,46
	4	7	0	1	8	0,75
	5	3	5	0	8	0,46
	6	5	3	0	8	0,46
	7	7	1	0	8	0,75
	8	4	3	1	8	0,32
	TOTAL	47	15	2	64	5,21
	Pr	0,73	0,23	0,03		0,65
	Pr^2	0,54	0,05	0,00	0,60	Pe-

Kappa de Fleiss: 0,14 Slight Discordance

Tabla 22. Prueba 2 - Kappa de Fleiss. Consulta 1

El valor de Kappa de Fleiss es de 0,14 lo que quiere decir que hubo un desacuerdo leve entre los usuarios al evaluar los 8 documentos en la primera iteración de la primer consulta (ver Tabla 22).

Por último se procesaron los resultados de cada una de las consultas y se calculó la precisión total de la prueba, los resultados obtenidos se aprecian en la Tabla 23. Donde la precisión total es la precisión media de cada uno de los documentos para cada consulta y la precisión media total es la precisión media de los 8 documentos por cada iteración.

Consulta		1	2	3		
Iteración	Resultado	Precisión	Precisión	Precisión	Precisión total	Precisión media (MAP)
1	1	100,0%	87,5%	100,0%	95,8%	74,2%
	2	100,0%	75,0%	87,5%	87,5%	
	3	81,3%	62,5%	68,8%	70,8%	
	4	75,0%	56,3%	62,5%	64,6%	
	5	77,5%	60,0%	75,0%	70,8%	
	6	75,0%	58,3%	77,1%	70,1%	
	7	76,8%	55,4%	76,8%	69,6%	
	8	73,4%	53,1%	67,2%	64,6%	
2	1	75,0%	75,0%	100,0%	83,3%	64,9%
	2	81,3%	62,5%	87,5%	77,1%	
	3	68,8%	43,8%	75,0%	62,5%	
	4	56,3%	37,5%	75,0%	56,3%	
	5	65,0%	50,0%	77,5%	64,2%	
	6	58,3%	43,8%	79,2%	60,4%	
	7	60,7%	41,1%	76,8%	59,5%	
	8	56,3%	42,2%	70,3%	56,3%	
3	1	100,0%	87,5%	100,0%	95,8%	69,1%
	2	93,8%	81,3%	81,3%	85,4%	
	3	75,0%	43,8%	75,0%	64,6%	
	4	56,3%	18,8%	75,0%	50,0%	
	5	70,0%	50,0%	80,0%	66,7%	
	6	66,7%	47,9%	79,2%	64,6%	
	7	64,3%	46,4%	82,1%	64,3%	
	8	60,9%	46,9%	76,6%	61,5%	

Tabla 23. Prueba 2 – Precisión media (MAP) de las 3 consultas

14.4.3 Prueba 2: Comparación con otros Buscadores

Como se mencionó anteriormente, los diez (10) estudiantes restantes del curso de "Estructuras del lenguaje", realizaron la prueba utilizando los buscadores tradicionales Google y Bing, usaron las mismas consultas ingresadas en MyBestMetaWebSearch: consulta 1: "Class diagram", consulta 2: "Class diagram association"; consulta 3: "Class diagram inheritance"; la Tabla 24 muestra las estadísticas de la evaluación por parte de los estudiantes utilizando Google.

Consulta	Resultado	R	N	X	Total	Exactitud	Precisión	Precisión media (MAP)
1	1	10	0	0	10	100,0%	100,0%	91,6%
	2	8	2	0	10	80,0%	90,0%	
	3	9	1	0	10	90,0%	85,0%	
	4	10	0	0	10	100,0%	95,0%	
	5	9	1	0	10	90,0%	92,0%	
	6	10	0	0	10	100,0%	93,3%	
	7	7	2	1	10	70,0%	90,0%	
	8	7	3	0	10	70,0%	87,5%	
2	1	9	1	0	10	90,0%	90,0%	76,2%
	2	7	3	0	10	70,0%	80,0%	
	3	7	3	0	10	70,0%	70,0%	
	4	9	1	0	10	90,0%	80,0%	
	5	4	6	0	10	40,0%	72,0%	
	6	8	2	0	10	80,0%	73,3%	
	7	8	2	0	10	80,0%	74,3%	
	8	4	4	2	10	40,0%	70,0%	
3	1	10	0	0	10	100,0%	100,0%	83,2%
	2	9	1	0	10	90,0%	95,0%	
	3	8	2	0	10	80,0%	85,0%	
	4	8	2	0	10	80,0%	80,0%	
	5	6	4	0	10	60,0%	82,0%	
	6	5	5	0	10	50,0%	76,7%	
	7	6	4	0	10	60,0%	74,3%	
	8	6	2	2	10	60,0%	72,5%	

Tabla 24. Prueba 2- Precisión de las 3 consultas en Google- Estadísticas

Usando Google, para la primera consulta la precisión oscila entre 85,0% y 100,0%, obteniendo una precisión media de 91,6%; para la segunda consulta la precisión oscila entre 70,0% y 90,0%, obteniendo una precisión media de 76,2%, y finalmente, para la consulta 3 la precisión oscila entre 72,5% y 100,0%, obteniendo una precisión media de 83,2%. De este modo, comparado los resultados obtenidos en esta prueba Google obtuvo resultados de precisión mucho más altos que la primera prueba.

Consulta	Resultado	R	N	X	Total	Exactitud	Precisión	Precisión media (MAP)
1	1	10	0	0	10	100,0%	100,0%	37,9%
	2	1	9	0	10	10,0%	55,0%	
	3	0	10	0	10	0,0%	5,0%	
	4	0	10	0	10	0,0%	0,0%	
	5	7	1	2	10	70,0%	36,0%	
	6	2	3	5	10	20,0%	33,3%	
	7	7	3	0	10	70,0%	38,6%	
	8	1	9	0	10	10,0%	35,0%	
2	1	9	1	0	10	90,0%	90,0%	78,4%
	2	7	3	0	10	70,0%	80,0%	
	3	10	0	0	10	100,0%	85,0%	
	4	7	2	1	10	70,0%	85,0%	
	5	3	7	0	10	30,0%	72,0%	
	6	9	1	0	10	90,0%	75,0%	
	7	5	5	0	10	50,0%	71,4%	
	8	5	5	0	10	50,0%	68,8%	
3	1	0	9	1	10	0,0%	0,0%	39,4%
	2	10	0	0	10	100,0%	50,0%	
	3	5	5	0	10	50,0%	75,0%	
	4	3	7	0	10	30,0%	40,0%	
	5	3	7	0	10	30,0%	42,0%	
	6	1	9	0	10	10,0%	36,7%	
	7	1	9	0	10	10,0%	32,9%	
	8	8	2	0	10	80,0%	38,8%	

Tabla 25. Prueba 2- Precisión de las 3 consultas en Bing- Estadísticas

Usando Bing, para la primera consulta la precisión oscila entre 0,0% y 100,0%, obteniendo una precisión media de 37,9%; para la segunda consulta la precisión

oscila entre 68,8% y 90,0%, obteniendo una precisión media de 78,4% y finalmente, para la consulta 3 la precisión oscila entre 0,0% y 75,0%, obteniendo una precisión media de 39,4%. En esta prueba Bing obtuvo resultados de precisión mucho más altos que la primera prueba.

De los resultados obtenidos en la prueba 2 se puede notar que MybestMetaWebSearch reporta resultados menos relevantes que los reportados por los Google. Por otro lado, los resultados obtenidos en la primera y en la tercera consulta son las relevantes que los reportados por Bing como se puede observar en la Tabla 26.

Precisión media (MAP)			
consulta	Google	Bing	MyBest Search
1	91,6%	37,9%	74,2%
2	76,2%	78,4%	64,9%
3	83,2%	39,4%	69,1%

Tabla 26. Prueba 2- Comparación de la precisión media frente a buscadores tradicionales

14.4.4 Conclusiones de las pruebas con usuarios

De acuerdo a las pruebas realizadas con los usuarios, en primer lugar se pudo observar que los snippets (resúmenes de los documentos) entregados por los buscadores tradicionales son de baja calidad, es decir, los usuarios no pudieron dar un juicio de relevancia solamente leyendo dicho snippet, sino que necesitaron consultar los documentos recuperados para poder establecer si eran o no relevantes para su necesidad de información. En segundo lugar, se pudo observar que los 8 primeros documentos recuperados en la segunda y tercera iteración en su mayoría fueron del tipo *.txt, *.zdat, *.clabel, *.list y *.arff. Lo que denota una debilidad en los procesos de ranking externos, es decir en los buscadores tradicionales utilizados aún no se hace un manejo adecuado de los documentos que contienen únicamente listas con muchos términos (sin sentido) o link farms (páginas con sólo enlaces y enlaces

que se intercambian mutuamente) lo que hace que dichos documentos tengan un alto valor de alto de “relevancia” pero que en realidad no satisfacen la necesidad de información del usuario.

Por otro lado, de las evaluaciones realizadas con los usuarios se puede concluir que MybestMetaWebSearch presenta resultados que no siempre son mejores comparados con los buscadores tradicionales, o en ocasiones sólo se obtienen mejoras que no son significativas. Esto se presenta por las siguientes razones:

1. La calidad de los snippets afecta directamente los términos que forman el contexto del usuario y la comunidad, si la calidad no es apropiada (por ejemplo: snippets que repiten términos de la pagina o que solo toman los primeros términos que aparece en la pagina y no un resumen apropiado de la misma) los términos que se ponderan no necesariamente están relacionados con el verdadero contenido de los documentos y por ello el contexto no refleja la verdadera relación entre los documentos relevantes o no, y el usuario o la comunidad. Cabe aclarar que la baja calidad de los snippets es un problema externo al modelo propuesto.
2. De la calidad de los términos almacenados en el contexto del usuario, depende la calidad del proceso de expansión implícito que se haga. Por lo tanto, si se tiene una representación real del perfil del usuario, entonces la expansión de las consultas pueden ayudar a mejorar los resultados obtenidos, en caso contrario, el proceso de expansión implícita puede desviar el objetivo de la consulta original, lo que ocasiona obtener resultados que no satisfacen la necesidad de información del usuario. Como se menciono anteriormente, esto es consecuencia de tener snippets de baja calidad (problema externo al modelo planteado)
3. El modelo es sensible a las evaluaciones realizadas previamente y si los usuarios (en este caso jueces de evaluación) califican erradamente los documentos (relevantes como no relevantes y viceversa), los proyectores (propuestos por Massimo) que reflejan la esencia del contexto de los usuarios, se alejan de la representación real del perfil de ese usuario, y de esta forma los nuevos

resultados no mejoran su precisión. Este problema no se presentó en los experimentos con colecciones cerradas donde se tenía los juicios de relevancia de personas expertas en el tema y lo que da validez a la presente propuesta.

4. En las pruebas se pudo observar que los usuarios hacen un proceso calificación de los documentos a conciencia hasta la segunda o tercera iteración de la consulta dos, luego se comienzan a distraer y a emitir juicios poco confiables, porque este proceso de calificación cada vez consume más tiempo, debido a que se deben actualizar dinámicamente las matrices S^+, C, S y S_c, que cada vez contienen más términos.

Parte VI – Conclusiones y Trabajo Futuro

15. Conclusiones

La metodología de desarrollo utilizada en la presente tesis de maestría, basada en ciclos iterativos e incrementales (modelo, aplicación, evaluación) permitió manejar la complejidad del problema y evaluar adecuadamente las características que el modelo debía cumplir en cada ciclo. Logrando así, un control más efectivo del cumplimiento gradual de los objetivos de la investigación y la disminución de los riesgos de la misma.

El modelo del meta buscador web propuesto representa una posible solución al problema planteado inicialmente de sobrecarga de información y la baja relevancia de los resultados obtenidos por los buscadores web tradicionales. Para lo cual, el modelo usa explícitamente dos estrategias: la adecuada gestión del contexto del usuario basado en la propuesta "A Basis for Information Retrieval in Context" [1], y la retroalimentación que el usuario puede registrar explícitamente al Sistema de Recuperación de Información (o búsqueda Web) usando técnicas de filtrado colaborativo basadas en ítem.

La aplicación Web denominada MyBestMetaWebSearch disponible en www.mybestmetawebsearch.com, construida con base en el modelo del meta buscador propuesto, permite filtrar y re-ordenar los resultados entregados por los buscador tradicionales (Google y Bing) de acuerdo al contexto del usuario y de la comunidad, presentando en algunos casos resultados más precisos y relevantes a las necesidades de información de los usuario.

La evaluación del modelo propuesto se realizó utilizando medidas clásicas del área de la recuperación de la información, satisfacción del usuario y relevancia, a través de la Curva de Precision-Recuerdo, Mean Average Precision (MAP), Precisión en K resultados ordenados y el estadístico Kappa. Este proceso de evaluación se dividió en dos partes:

- Los resultados de la evaluación fueron satisfactorios pero el costo computacional para mantener actualizado el contexto del usuario es de $O(n^2)$

En la primera se utilizó una colección cerrada de textos denominada CACM, con el fin de calcular la Curva de Precision-Recuerdo. Los 4 experimentos realizados se hicieron con memoria de consulta, es decir, se simuló la ejecución de una consulta cinco veces, guardando el feedback de los resultados en el contexto del usuario. Los resultados obtenidos en cada experimento permitieron realizar la afinación de algunos parámetros y de algunos algoritmos hasta obtener el modelo finalmente expuesto. La evaluación del modelo propuesto en comparación con Rocchio muestra resultados menos relevantes, aunque los resultados son consistentes y en general buenos, y en cada iteración se obtienen mejoras en la precisión.

En la segunda parte, se calcularon las medidas Mean Average Precision (MAP), Precisión en K resultados ordenados y el estadístico Kappa, para lo cual se realizaron once (11) pruebas con estudiantes de diferentes niveles del Programa de Ingeniería de Sistemas de la Universidad del Cauca, en este documento sólo se muestran las dos últimas pruebas realizadas sobre MyBestMetaWebSearch, teniendo en cuenta que todas se hicieron de la misma forma: para cada prueba se seleccionaron 3 consultas, luego para cada consulta se realizaron 3 iteraciones de búsqueda y para cada iteración se evaluaron los primeros 8 documentos. Las pruebas mostraron que los 8 primeros resultados entregados por el modelo propuesto en algunas ocasiones son mejores que los entregados por los buscadores Web tradicionales más usados hoy en día, Google y Bing, aunque dicha mejora no es significativa.

Por otro lado, la calidad de los snippets obtenidos no es la apropiada, esto tiene dos implicaciones sobre el modelo propuesto: (i) los términos que se ponderan no necesariamente están relacionados con el verdadero contenido de los documentos y por ello el contexto no refleja la verdadera relación entre los documentos relevantes o no, y el usuario o la comunidad, (ii) los procesos de expansión de consulta se basan en los términos almacenados en el contexto del usuario y de la comunidad, por lo tanto al no tener los términos adecuados, se puede desviar el objetivo de la consulta original y obtener resultados que nada tienen que ver con la necesidad de información del usuario.

El proceso de calificación que hacen los usuarios con respecto a la relevancia de los documentos recuperados tiene un costo computacional alto, porque se requiere actualizar dinámicamente las matrices S^{+}, C, S y S_c, que cada vez contienen más términos.

16. Trabajo Futuro

Como trabajo futuro se plantea generar snippets de buena calidad, de esta manera se tendrían los términos que realmente representan el contexto del usuario y se podrían manejar matrices de correlación reducidas, con el fin de mejorar el rendimiento de los procesos de creación y actualización de las matrices S^{+}, C, S y S_c, lo que a su vez se vería reflejado en mejoras en los procesos de expansión de las consultas y ranking.

Por otro se espera que el modelo del meta buscador sea multilenguaje (extender su uso por lo menos al lenguaje Español) y que se pueda manejar múltiples ejes temáticos o contextos de la comunidad diferentes.

Parte VII – Glosario y Referencias Bibliográficas

A continuación se presenta una lista de los términos empleados en la investigación y una corta explicación del mismo. Luego se presenta la lista de referencias bibliográficas que soportan formalmente la investigación.

17. Glosario

- **Recuperación de información:** Busca las mejores formas de representar, almacenar, organizar y acceder ítems de información (documentos generalmente no estructurados) en forma automática para relacionarlos con las solicitudes de búsqueda de un usuario.
- **Usuario:** Individuo que utiliza o trabaja en el sistema para encontrar respuestas a sus necesidades de información.
- **Contexto del usuario:** Las características que describen al usuario, el tiempo, el lugar o cualquier otra cosa que emerge de la interacción entre el usuario y un Sistema de Recuperación de Información [1].
- **Consulta:** Expresa la necesidad de información de un usuario a partir de un conjunto de palabras clave.
- **Documento Web:** Es un documento electrónico de cualquier tipo de formato como pdf, doc, ppt, xml, etcétera, que se encuentra disponible o publicado en la Web, que contiene información específica de un tema en particular y que es almacenado en algún sistema de cómputo que se encuentre conectado a la red mundial de información denominada Internet.
- **Indexación:** hace referencia a la acción de agregar una o más páginas web a las bases de datos de los buscadores de internet, las cuales pueden ser consultadas y reportadas en los resultados de búsquedas de los motores de búsqueda.

- **Motor de Búsqueda**. Sistema que recorre la red recolectando e indexando la mayor cantidad de información posible, gracias a programas automáticos conocidos como robots o spider, referente a una consulta realizada por medio de palabras clave o con árboles jerárquicos por temas; el resultado de la búsqueda es un listado de direcciones web en los que se mencionan temas relacionados con las palabras clave buscadas.
- **Meta Buscador Web**: Sistemas de búsqueda web que no disponen de bases de datos propias (índices), razón por la cual usan los índices de buscadores tradicionales. Recogen la petición del usuario y la envían a los buscadores, éstos devuelven los resultados al meta buscador, quien los clasifica antes de presentarlos al usuario.

18. Referencias Bibliográficas

1. Melucci, M., *A basis for information retrieval in context.* ACM Trans. Inf. Syst., 2008. 26(3): p. 1-41.
2. Nielsen, J. *When Search Engines Become Answer Engines*. 2004 [cited; Available from: http://www.useit.com/alertbox/20040816.html.
3. O'hara, K. and N. Shabdolt. *Knowledge Technologies and the semantic web* 2004 [cited; Available from: http://eprints.ecs.soton.ac.uk/12469/.
4. Sullivan, D. *Nielsen NetRatings Search Engine Ratings*. 2006 [cited; Available from: http://searchenginewatch.com/showPage.html?page=2156451.
5. Baeza-Yates, R., C. Castillo, and B. Keith, *Web Searching*, in *Encyclopedia of Language & Linguistics*. 2006, Elsevier: Oxford. p. 527-538.
6. Liaw, S.-S. and H.-M. Huang, *Information retrieval from the World Wide Web: a user-focused approach based on individual experience with search engines.* Computers in Human Behavior, 2006. 22(3): p. 501-517.
7. Baeza-Yates, R., A. and B. Ribeiro-Neto, *Modern Information Retrieval*. 1999: Addison-Wesley Longman Publishing Co., Inc. 513.
8. Jacobson, I., G. Booch, and J. Rumaugh, *. El Proceso Unificado de Desarrollo de Software*. 2000, Madrid: Addison Wesley.
9. Jansen, B.J. and A. Spink, *How are we searching the World Wide Web? A comparison of nine search engine transaction logs.* Information Processing & Management, 2006. 42(1): p. 248-263.
10. Anderson, J.D., *The Turn: Integration of Information Seeking and Retrieval in Context, Peter Ingwersen, Kalervo Jarvelin. Springer, Dordrecht, Netherlands (c2005). xiv, 448 p. .* Information Processing & Management, 2007. 43(3): p. 821-822.
11. Tanudjaja, F. and L. Mui, *Persona: A Contextualized and Personalized Web Search (PDF)* in *35th Annual Hawaii International Conference on System Sciences (HICSS'02)*. 2002: Big Island, Hawaii
12. Speretta, M. and S. Gauch. *Personalized search based on user search histories*. in *Web Intelligence, 2005. Proceedings. The 2005 IEEE/WIC/ACM International Conference on*. 2005.
13. Manning, C., P. Raghavan, and H. Schütze, *An Introduction to Information Retrieval*. 2007, Cambridge University Press: Cambridge, England.
14. Rijsbergen, C.J.V., *Information Retrieval*. 1979: Butterworth-Heinemann. 208.
15. Chakrabarti, S., *Web Search and Information Retrieval*, in *Mining the Web*. 2003, Morgan Kaufmann: San Francisco. p. 45-76.

16. Christopher D. Manning, P.R., Hinrich Schütze, *An Introduction to Information Retrieval.* Cambridge University Press Cambridge, England, 2008.

17. *Text Retrieval Conference*. [cited; Available from: http://trec.nist.gov/.

18. Chen and Kuo, *An information retrieval system based on a user profile* Journal of Systems and Software, 2000. 54(1): p. 3-8.

19. Barry, C.L., *User-Defined Relevance Criteria: An Exploratory Study.* JOURNAL OF THE AMERICAN SOCIETY FOR INFORMATION SCIENCE-A, 1994. 45(1): p. 149-159.

20. Perrault, C.R., F.A. James, and R.C. Philip, *Speech acts as a basis for understanding dialogue coherence*, in *Proceedings of the 1978 workshop on Theoretical issues in natural language processing*. 1978, Association for Computational Linguistics: Urbana-Champaign, Illinois.

21. RICH, E., *User modeling via stereotypes.* 1979: p. 329-354.

22. Haym, H., B. Chumki, and D.D. Brian, *Learning to personalize.* Commun. ACM, 2000. 43(8): p. 102-106.

23. Fink, J. and A. Kobsa, *User Modeling for Personalized City Tours.* Springer Netherlands, 2000. Volume 10(2-3): p. 147-180.

24. Alfred, K., *Generic User Modeling Systems.* User Modeling and User-Adapted Interaction, 2001. 11(1-2): p. 49-63.

25. Montaner, M., B. López, and J.L. de la Rosa, *A Taxonomy of Recommender Agents on the Internet.* Artificial Intelligence Review, 2003. 19(4): p. 285-330.

26. Pawan Lingras, R.Y., Chad West, *Interval Set Clustering of Web Users with Rough K-Means*. 2003, Saint Mary's University.

27. Brusilovsky, P., J. Eklund, and E. Schwarz, *Web-based Education for All: A Tool for Development Adaptive Courseware*, in *Seventh International World Wide Web Conference*. 1998, Computer Networks and ISDN Systems. p. 291-300.

28. Paul De, B., et al., *AHA! The adaptive hypermedia architecture*, in *Proceedings of the fourteenth ACM conference on Hypertext and hypermedia*. 2003, ACM: Nottingham, UK.

29. Sheth, B. and P. Maes, *Evolving agents for personalized information filtering*, in *Artificial Intelligence for Applications, 1993. Proceedings., Ninth Conference on*. 1993: Orlando, FL, USA. p. 345-352.

30. Glen, J. and W. Jennifer, *Scaling personalized web search*, in *Proceedings of the 12th international conference on World Wide Web*. 2003, ACM: Budapest, Hungary.

31. Fang, L., Y. Clement, and M. Weiyi, *Personalized Web Search For Improving Retrieval Effectiveness.* IEEE Trans. on Knowl. and Data Eng., 2004. 16(1): p. 28-40.

32. Eisenstein, J., J. Vanderdonckt, and A. Puerta, *Adapting to mobile contexts with user-interface modeling.* Third IEEE Workshop on Mobile Computing Systems and Applications (WMCSA'00), 2000.

33. Hanumansetty, R.G., *Model Based Approach for Context Aware and Adaptive user Interface Generation*, in *Faculty of the Virginia Polytechnic Institute and State University*. 2004.

34. Mitrović, N. and E. Mena, *Adaptive User Interface for Mobile Devices*, in *Interactive Systems:Design, Specification, and Verification*. 2002. p. 29-43.

35. Vassileva, J. and R. Deters, *Dynamic Courseware Generation on the WWW.* British Journal of Educational Technologies. 29(1): p. 4-15.

36. Encarna, L.M., *Multi-level user support through adaptive hypermedia: a highly application-independent help component*, in *Proceedings of the 2nd international conference on Intelligent user interfaces*. 1997, ACM: Orlando, Florida, United States.

37. Terveen, L. and W. Hill, *Beyond Recommender Systems: Helping People Help Each Other* in *In HCI In The New Millennium*, Addison-Wesley, Editor. 2001.

38. Amazon. *Sitio web de Amazon*. [cited; Available from: http://www.amazon.com/.

39. BroadVisionOne-To-One, *Sitio Web BroadVision.*

40. Liliana, A. and G. Anna, *Tailoring the Interaction with Users in Web Stores.* User Modeling and User-Adapted Interaction, 2000. 10(4): p. 251-303.

41. Fink, J. and A. Kobsa, *A Review and Analysis of Commercial User Modeling Servers for Personalization on the World Wide Web. User Modeling and User-Adapted Interaction.* Springer Netherlands, 2000. Volume 10 (2-3): p. 209-249.

42. Krishna, B., K. Tomonari, and A. Michael, *Personalized, interactive news on the Web.* Multimedia Syst., 1998. 6(5): p. 349-358.

43. Dwi, H.W., R.I. Thomas, and Y. John, *An adaptive algorithm for learning changes in user interests*, in *Proceedings of the eighth international conference on Information and knowledge management*. 1999, ACM: Kansas City, Missouri, United States.

44. Callan, J., et al., *Personalisation and Recommender Systems in Digital Libraries Joint NSF-EU DELOS Working Group Report.* 2003.

45. Smeaton, A. and J. Callan, *Proceedings of the Second DELOS Network of Excellence Workshop on Personalisation and Recommender Systems in Digital Libraries.* ERCIM Workshop Proceedings - No. 01/W03, 2001.

46. L. Ardissono, et al., *Personalized recommendation of tourist attractions for desktop and handset devices* Applied Artificial Intelligence, Special Issue on Artificial Intelligence for Cultural Heritage and Digital Libraries, 2003. 17 No. 8-9: p. 687-714.

47. Eurekster. *Sitio Web Eurekster*. [cited; Available from: http://eurekster.com.

48. Salton, G.a.B., C. , *Improving retrieval performance by relevance feedback.* Journal of the American Society for Information, 1999. 41(4): p. 288 - 297.

49. Taher, H.H., *Topic-sensitive PageRank*, in *Proceedings of the 11th international conference on World Wide Web*. 2002, ACM: Honolulu, Hawaii, USA.

50. Chirita, P.-A., D. Olmedilla, and W. Nejdl, *Finding Related Hubs and Authorities.* First Latin American Web Congress (LA-WEB'03), 2003: p. 214.

51. Chakrabarti, S., M. van den Berg, and B. Dom, *Focused crawling: a new approach to topic-specific Web resource discovery.* Computer Networks, 1999. 31: p. 1623-1640.

52. Kerschberg, L., W. Kim, and A. Scime, *A Semantic Taxonomy-Based Personalizable Meta-Search Agent*, in *Second International Conference on Web Information Systems Engineering (WISE'01)*, wise, Editor. 2001: Kyoto, Japan. p. 0041.

53. Philip, Z. and Z. Yi, *Bayesian adaptive user profiling with explicit \& implicit feedback*, in *Proceedings of the 15th ACM international conference on Information and knowledge management*. 2006, ACM: Arlington, Virginia, USA.

54. Paul Alexandru, C., et al., *Using ODP metadata to personalize search*, in *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*. 2005, ACM: Salvador, Brazil.

55. Pitkow, J., Schuatze, H., Cass, T., Cooley, R., Turnbull, D., Edmonds, A., Adar, E. and Breuel, T, *Personalized search.* Commun. ACM. 45(9): p. 50-55.

56. Asnicar, F. and C. Tasso. *ifWeb: a Prototype of User Model-Based Intelligent Agent for Document Filtering and Navigation in the World Wide Web*. in *Proc. of 6th International Conference on User Modelling*. 1997.

57. Liu, S., C.A. McMahon, and S.J. Culley, *A review of structured document retrieval (SDR) technology to improve information access performance in engineering document management.* Computers in Industry, 2008. 59(1): p. 3-16.

58. Wallace, M.S., G. , *Towards a context aware mining of user interests for consumption of multimedia documents*, in *Multimedia and Expo, 2002. ICME '02. Proceedings. 2002 IEEE International Conference* 2002: Lausanne, Switzerland. p. 733- 736.

59. Wallace, M. and G. Stamou. *Towards a context aware mining of user interests for consumption of multimedia documents*. in *Multimedia and Expo, 2002. ICME '02. Proceedings. 2002 IEEE International Conference on*. 2002.

60. Paolo, F. and G. Antonio, *A personalized search engine based on web-snippet hierarchical clustering*, in *Special interest tracks and posters of the 14th international conference on World Wide Web*. 2005, ACM: Chiba, Japan.

61. Decipho. *Sitio Web Decipho*

[cited; Available from: http://www.decipho.com/.

62. Google. *Sitio Web Google Co-op*. [cited; Available from: http://www.google.com/coop.

63. Google. *Google Personalized Web Search*. [cited; Available from: http://labs.google.com/personalized.

64. Google. *iGoogle. Google personalzed search & Home.* [cited; Available from: http://www.google.com/ig.

65. Entopia. *Página Web Entopia*. [cited; Available from: http://www.entopia.com.

66. Yahoo. *My Yahoo! Personal Search engine*. [cited; Available from: http://my.yahoo.com/.

67. Bruce, E., *The Pragmatic Roots of Context*, in *Proceedings of the Second International and Interdisciplinary Conference on Modeling and Using Context*. 1999, Springer-Verlag.

68. Paavo, A., et al., *Generalized contextualization method for XML information retrieval*, in *Proceedings of the 14th ACM international conference on Information and knowledge management*. 2005, ACM: Bremen, Germany.

69. Daniel, B., M.H. David, and M.-A. Dan, *Improving proactive information systems*, in *Proceedings of the 10th international conference on Intelligent user interfaces*. 2005, ACM: San Diego, California, USA.

70. Travis, B. and B.L. David, *Real time user context modeling for information retrieval agents*, in *Proceedings of the tenth international conference on Information and knowledge management*. 2001, ACM: Atlanta, Georgia, USA.

71. Kazunari, S., H. Kenji, and Y. Masatoshi, *Adaptive web search based on user profile constructed without any effort from users*, in *Proceedings of the 13th international conference on World Wide Web*. 2004, ACM: New York, NY, USA.

72. Bharat, K., *SearchPad: explicit capture of search context to support Web search.* Computer Networks, 2000. 33(1-6): p. 493-501.

73. Dou, Z., Song, R. and Wen, J., A, *A Large-scale Evaluation and Analysis of Personalized Search Strategies.pdf* in *16th international World Wide Web conference (WWW2007)*. 2007: Banff, Alberta, Canada. p. 572-581.

74. Xuehua, S., T. Bin, and Z. ChengXiang, *Implicit user modeling for personalized search*, in *Proceedings of the 14th ACM international conference on Information and knowledge management*. 2005, ACM: Bremen, Germany.

75. Finkelstein, L., et al., *Placing Search in Context: The Concept Revisited. in World Wide Web.* 2001: p. 406-414.

76. Reiner, K., et al., *Searching with context*, in *Proceedings of the 15th international conference on World Wide Web*. 2006, ACM: Edinburgh, Scotland.

77. Rocchio, J.a.S., G., *Relevance feedback in information retrieval*, ed. Prentice-Hall. 1971.

78. Melucci, M., *Context modeling and discovery using vector space bases*, in *In Proceedings of the ACM Conference on Information and Knowledge Management (CIKM)*. 2005, ACM Press: Bremen, Germany. p. 808–815.

79. Apostol , T.M., *Calculus*. 1969: New York.

80. Valsan, Z. and M. Emele. *Thematic text clustering for domain specific language model adaptation*. in *Automatic Speech Recognition and Understanding, 2003. ASRU '03. 2003 IEEE Workshop on*. 2003.

81. Jong-Seok, L. and O. Sigurdur, *Two-way cooperative prediction for collaborative filtering recommendations.* Expert Syst. Appl., 2009. 36(3): p. 5353-5361.

82. Linden, G., B. Smith, and J. York, *Amazon.com recommendations: item-to-item collaborative filtering.* Internet Computing, IEEE, 2003. 7(1): p. 76-80.

83. Badrul, S., et al., *Item-based collaborative filtering recommendation algorithms*, in *Proceedings of the 10th international conference on World Wide Web*. 2001, ACM: Hong Kong, Hong Kong.

84. Marlin, B., *Collaborative Filtering: A Machine Learning Perspective*, in *Computer Science*. 2004, University of Toronto: Toronto, Canada. p. 137.

85. Schickel-Zuber, V., *Ontology filtering*, in *FACULTÉ INFORMATIQUE ET COMMUNICATIONS*. 2007, ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE: Suisse. p. 214.

86. Konstan, J.A., et al. *Recommender Systems: A GroupLens perspective*. in *Recommender Systems: Papers from the 1998 Workshop (AAAI Technical Report WS-00-04)*. 1998. Menlo Park, CA, Estados Unidos.

87. Paul, H., K. Georgia, and G.-M. Hector, *Can social bookmarking improve web search?*, in *Proceedings of the international conference on Web search and web data mining*. 2008, ACM: Palo Alto, California, USA.

88. Lee, Y.-J., *VisSearch: A collaborative Web searching environment.* Computers & Education, 2005. 44(4): p. 423-439.

89. Chen, L.-C., C.-J. Luh, and C. Jou, *Generating page clippings from web search results using a dynamically terminated genetic algorithm.* Information Systems, 2005. 30(4): p. 299-316.

90. Godoy, D. and A. Amandi, *Modeling user interests by conceptual clustering.* Information Systems, 2006. 31(4-5): p. 247-265.

91. Jaime, T., *Supporting finding and re-finding through personalization*. 2007, Massachusetts Institute of Technology. p. 1.

92. Billerbeck, B. and J. Zobel, *Efficient query expansion with auxiliary data structures.* Information Systems, 2006. 31(7): p. 573-584.

93. Lin, H.-C., L.-H. Wang, and S.-M. Chen, *Query expansion for document retrieval based on fuzzy rules and user relevance feedback techniques.* Expert Systems with Applications, 2006. 31(2): p. 397-405.

94. Joachims, T. and F. Radlinski, *Search Engines that Learn from Implicit Feedback.* Computer, 2007. 40(8): p. 34-40.

95. Vozalis, M.G. and K.G. Margaritis, *Using SVD and demographic data for the enhancement of generalized Collaborative Filtering.* Information Sciences, 2007. 177(15): p. 3017-3037.

96. de Campos, L.M., J.M. Fernández-Luna, and J.F. Huete, *A collaborative recommender system based on probabilistic inference from fuzzy observations.* Fuzzy Sets and Systems, 2008. 159(12): p. 1554-1576.

97. Zhang, Z. and O. Nasraoui, *Mining search engine query logs for social filtering-based query recommendation.* Applied Soft Computing, 2008. 8(4): p. 1326-1334.

98. Liang, T.-P., et al., *A semantic-expansion approach to personalized knowledge recommendation.* Decision Support Systems, 2008. 45(3): p. 401-412.

99. Ko, Y., H. An, and J. Seo, *Pseudo-relevance feedback and statistical query expansion for web snippet generation.* Information Processing Letters, 2008. 109(1): p. 18-22.

100. Clements, M., A.P. de Vries, and M.J.T. Reinders, *The influence of personalization on tag query length in social media search.* Information Processing & Management. 46(4): p. 403-412.

101. Hernández del Olmo, F., E. Gaudioso, and E.H. Martin, *The task of guiding in adaptive recommender systems.* Expert Systems with Applications, 2009. 36(2, Part 1): p. 1972-1977.

102. Trias i Mansilla, A. and J.L. de la Rosa i Esteva, *Asknext: An agent protocol for social search.* Information Sciences, 2011. 190(0): p. 144-161.

103. da Costa Pereira, C.l., M. Dragoni, and G. Pasi, *Multidimensional relevance: Prioritized aggregation in a personalized Information Retrieval setting.* Information Processing & Management, 2011. 48(2): p. 340-357.

104. Jones, S., et al., *Relevance feedback for real-world human action retrieval.* Pattern Recognition Letters, 2011. 33(4): p. 446-452.

105. Lou, Y., Z. Li, and Q. Chen, *Semantic relevance ranking for XML keyword search.* Information Sciences, 2012. 190(0): p. 127-143.

106. Martínez, F., *Propuesta y desarrollo de un modelo para la evaluación de la recuperación de información en Internet*, in *Información y Documentación*. 2002, Universidad de Murcia: Murcia, España. p. 283.

107. Cacheda, F., V. Formoso, and V. Carneiro, *Performance Analysis of Distributed Web Information Retrieval Systems.* Latin America Transactions, IEEE (Revista IEEE America Latina), 2007. 5(6): p. 479-485.

108. Can, F., R. Nuray, and A.B. Sevdik, *Automatic performance evaluation of Web search engines.* Information Processing & Management, 2004. 40(3): p. 495-514.

109. Wang, Y.D. and G. Forgionne, *A decision-theoretic approach to the evaluation of information retrieval systems.* Information Processing & Management, 2006. 42(4): p. 863-874.

110. Petrelli, D., *On the role of user-centred evaluation in the advancement of interactive information retrieval.* Information Processing & Management, 2008. 44(1): p. 22-38.