PROCESO DE EXPANSIÓN DE CONSULTA EN UN META BUSCADOR WEB BASADO EN CO-OCURRENCIA DE TÉRMINOS RELEVANTES Y NO RELEVANTES – ECWEB

EDUARDO ESTÉVEZ MENDOZA

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO MECÁNICA
INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA
2011

PROCESO DE EXPANSIÓN DE CONSULTA EN UN META BUSCADOR WEB BASADO EN CO-OCURRENCIA DE TÉRMINOS RELEVANTES Y NO RELEVANTES – ECWEB

EDUARDO ESTÉVEZ MENDOZA

TRABAJO DE GRADO PARA OPTAR AL TÍTULO DE INGENIERO DE SISTEMAS

DIRECTOR

LUIS CARLOS GÓMEZ FLORÉZ, MSC.

CO-DIRECTOR

CARLOS ALBERTO COBOS LOZADA, PH.D. (C)

UNIVERSIDAD INDUSTRIAL DE SANTANDER

FACULTAD DE INGENIERÍAS FÍSICO MECÁNICA

INGENIERÍA DE SISTEMAS E INFORMÁTICA

BUCARAMANGA

2011

DEDICATORIA

A Dios que me dio la vida y formo mi carácter.

A mi familia que con su apoyo y sacrificio me brindo las herramientas necesarias.

A mi mamá que pase lo que pase siempre me ha brindado su amor incondicional.

A mi hermana que con su trabajo hizo esto posible.

A Jesyca María, Eduardo Andrés y Sebastián que son los motores de mi vida.

A los Ing. Martha, Carlos y Luis Carlos que me ayudaron a llegar a la meta.

A los profesores que me forjaron como profesional.

A mis compañeros de STI por que hicieron un ambiente de trabajo agradable.

A mis amigos que siempre estuvieron en las buenas y en las malas.

A todos los que estando cerca o lejos, que estuvieron de paso o se quedaron en mi vida hicieron de este sueño una realidad.

AGRADECIMIENTOS

A Dios, por permitir mi formación como profesional, por su guía y por su amor incondicional.

A mi familia, que con amor y sacrificio me han acompañado cada momento de vida, dándolo todo de sí para que pueda alcanzar mis metas y cumplir mis sueños.

A los motores de mi vida, especialmente un motorcito pequeñito que con su sonrisa me da las fuerzas suficientes para salir a delante

Al MSc. Luis Carlos Gómez Flórez por su apoyo, orientación, dedicación y compromiso con el grupo de investigación.

Al Ph.D. (c). Carlos Alberto Cobos Lozada por su dedicación, por su tiempo, apoyo y su enorme conocimiento para guiarme en este reto.

A mis compañeros, amigos y educadores, quienes me acompañaron en este proceso, por su ánimo, colaboración y palabras de aliento.

Para finalizar, agradezco a la Universidad Industrial de Santander institución que me forjó como persona y como profesional, a través del programa de Ingeniería de Sistemas y me permitió en conjunto con la Universidad del Cauca, institución en la que realice un semestre de intercambio, realizar mi proyecto de grado, por medio del grupo de investigación en Sistemas y Tecnologías de la Información (STI) y el grupo de investigación y desarrollo en Tecnologías de la Información (GTI) respectivamente.

TABLA DE CONTENIDO

LISTA DE TABLAS

LISTA DE FIGURAS

LISTA DE ANEXOS

RESUMEN

TÍTULO: PROCESO DE EXPANSIÓN DE CONSULTA EN UN META BUSCADOR WEB BASADO EN CO-OCURRENCIA DE TÉRMINOS RELEVANTES Y NO RELEVANTES [*]

AUTOR: Eduardo Estévez Mendoza[**]

PALABRAS CLAVE: Expansión de consulta, Rocchio, Término relevante, IDF, Frecuencia invertida de documento, recuperación de información.

DESCRIPCIÓN

Se ha demostrado que el proceso de expansión de las consultas en el modelo espacio vectorial de representación de documentos en un sistema de recuperación de información, es una técnica útil para mejorar la relevancia medida por la precisión de los resultados entregados a los usuarios, ya que en general reporta mejores niveles de relevancia en los resultados que los obtenidos por otras formas de representación de documentos, como lo son el modelo probabilístico y el modelo booleano que junto con el modelo vectorial son los más destacados. En este documento se presenta un nuevo algoritmo y una variación del mismo para realizar expansión de consultas en un sistema de recuperación de información. Estos algoritmos se basan en una nueva función discreta que define la importancia relativa de un término en una colección de documentos, y en una matriz de co-ocurrencia de términos que representa la relación de términos relevantes y no relevantes, definidos de esta forma por la calificación previa que el usuario le da a los documentos que se le han presentado en consultas anteriores. El algoritmo y su variación se evalúan frente a la búsqueda por similitud de cosenos y el algoritmo de expansión propuesto por Rocchio, obteniendo excelentes resultados sobre la colección de datos CACM (artículos publicados en la revista Communications of the ACM), y la colección de datos LISA (Library & Information Science Abstracts), además se pone a prueba con estudiantes pertenecientes al curso de ingeniería de Sistemas de la Universidad del Cauca.

---

[*] Trabajo de grado. Modalidad: Investigación.

[**] Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingeniería de Sistemas e Informática. Director: Luis Carlos Gómez Flórez.
Co-director: Carlos Alberto Cobos Lozada.

SUMMARY

TÍTLE: PROCESO EXPANSION OF QUERY PROCESS IN A WEB BROWSER-BASED TARGET CO-OCCURRENCE OF NO SIGNIFICANT AND RELEVANT TERMS [*]

AUTHOR: Eduardo Estévez Mendoza[**]

KEYWORDS: Query expansion, Rocchio, Relevant Term, IDF, Inverse document frequency.

DESCRIPTION

It has been shown that the process of expanding queries in the vector space model representation of documents in an information retrieval system is a useful technique to improve the relevance measure for the accuracy of the results delivered to users as generally reported higher levels of relevance in the results obtained by other forms of representation of documents, such as the probabilistic model and the Boolean model with the vector model are the most prominent. This paper presents a new algorithm and a variation of the same for query expansion in information retrieval system. These algorithms are based on a new discrete function that defines the relative importance of a term in a document collection, and an array of co-occurrence of terms represents the ratio of relevant and irrelevant terms defined in this way by the pre-qualify the user gives the documents have been presented in previous consultations. The algorithm and its variations are evaluated against the cosine similarity search algorithm and Rocchio proposed expansion, with excellent results on the CACM collection data (articles published in the journal Communications of the ACM), and data collection LISA (Library & Information Science Abstracts) in addition to being tested with students from the Systems Engineering course at the Universidad del Cauca.

[*] Work degree. Method: Research.

[**] Faculty of Physical-Mechanical Engineering. School of Systems Engineering and Computer Science.
Director: Luis Carlos Gómez Flórez.
Co-director: Carlos Alberto Cobos Lozada.

1. INTRODUCCIÓN

Debido al acelerado crecimiento de internet y a la creciente necesidad de acceso a la información que tienen las personas, se han diseñado e implementado diferentes sistemas de recuperación de información (RI) cada uno con técnicas que de una u otra forma buscan suplir esta necesidad entregándole al usuario mejores resultados a partir de ciertas condiciones de búsqueda (mayor precisión), mejorando igualmente su organización y presentación, de una forma rápida, con interfaces simples y funcionales.

El proceso tradicional de búsqueda en Web se encuentra limitado por los lenguajes de consulta y por la carencia de información semántica sobre el dominio al que se refiere el usuario ya que este no tiene conocimiento de cómo funciona el buscador que está utilizando. Esto provoca que el sistema no recupere todos los resultados relevantes y sí obtenga, por el contrario, resultados que nada tienen que ver con las necesidades del usuario [1].

Todos los buscadores hacen esfuerzos importantes por mejorar continuamente los resultados que entregan a los usuarios, una de las técnicas para lograr este propósito es la expansión de consulta.

La expansión de consultas involucra evaluar una entrada del usuario (las palabras que el usuario ingresa en el área de consulta de búsqueda, y a veces otros tipos de datos) y expandir la consulta de búsqueda para que se ajuste más a las verdaderas necesidades del usuario, el cual, en muchos casos no las expresa adecuadamente en la consulta [2].

La expansión de consultas involucra técnicas como

- Encontrar sinónimos de palabras
- Encontrar las diferentes formas morfológicas de las palabras involucradas en la búsqueda, aplicando técnicas de lematización (stemming).
- Reparar errores tipográficos y buscar automáticamente por la forma corregida o sugerirla
- Ponderar los resultados según la relevancia.

Dicha expansión puede ser realizada explicita (con la complicidad del usuario) o implícitamente (a espaldas del usuario). Pero sin importar si la expansión es explícita o implícita, el sistema de recuperación de información debe usar un método para representar los documentos y las consultas de los usuarios. Los métodos más destacados [3, 4] para dicha representación son: el modelo booleano, el modelo de espacio vectorial y el modelo probabilístico. Existen algunas variaciones de estos tres modelos, entre las más importantes están: el modelo de conjuntos difusos, el modelo booleano extendido, el modelo del espacio vectorial generalizado, el modelo de indexado de semántica latente, el modelo de

redes neuronales, el modelo de las redes bayesianas, el modelo de las redes de inferencia, el modelo de red de creencias, entre otros.

En este proyecto se presenta un método que está dentro del modelo de espacio vectorial debido a que se ha demostrado que es el modelo con el que se obtienen mejores resultados. Este método se basa en una función de evaluación de los documentos que involucra el cálculo de la importancia relativa de cada término (una nueva función de IDF), que representa el peso o la relevancia de ese término en cada documento de la colección. En este documento se explica además, cada paso del método, y se presentaran resultados experimentales que muestran que es una muy buena opción en el campo de la recuperación de información.

1.1 DESCRIPCIÓN DEL PROYECTO

1.1.1 PROBLEMA Y JUSTIFICACIÓN

Hoy en día, los motores de búsqueda web son la página inicial de la gran mayoría de los usuarios de Internet. Sin embargo, los resultados retornados por dichos buscadores, no son los más relevantes a las necesidades de los usuarios. Primero, porque la consulta expresada en unas pocas palabra claves, deja vacíos semánticos al motor de búsqueda, disminuyendo su capacidad de entregar resultados más adecuados. Segundo, porque la Web crece a un ritmo muy elevado y los buscadores no tienen la capacidad de indexar en tiempo real dicha información y mucho menos los índices temáticos. Tercero, porque los motores de búsqueda no registran ni utilizan adecuadamente la información de los usuarios (perfil y retroalimentación), entre otros problemas que se pueden mencionar.

El modelo de espacio vectorial (VSM por sus siglas en inglés, Vector Space Model), comúnmente usado en procesos de recuperación de información y

búsqueda web, ha demostrado que el proceso de ***expansión de consulta*** mejora la relevancia (medida por la precisión) de los resultados entregados a los usuarios[3, 5, 6]. La expansión de la consulta en un sistema de búsqueda web normalmente se hace desde una de dos perspectivas: Realimentación de relevancia del Usuario (URF por sus siglas en inglés, user relevance feedback) o Realimentación automática de relevancia (ARF por sus siglas en inglés, automatic relevance feedback)[3, 5, 6]. URF requiere que el usuario marque los documentos como relevantes o no relevantes y luego, a cada nueva consulta del usuario se le agregan o quitan los términos que el sistema ha encontrado como relevantes o no en los documentos marcado[3, 5, 6]. Rocchio propone la fórmula (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 qe es la consulta expandida[3, 5, 6].

$$q_e = \alpha \times q + \frac{\beta}{|R|}\sum_{d \in R} d - \frac{\gamma}{|R'|}\sum_{d \in R'} d \qquad \textbf{(1)}$$

En contraste con URF, la realimentación automática de relevancia, también conocida como pseudo realimentación (pseudo feedback) expande las consultas automáticamente basados en dos métodos: documentos globales y documentos parciales [3, 5, 6]. En los métodos basados en documentos globales se analizan todos los documentos de la colección y se establecen relaciones entre los términos (palabras), por lo que, estos métodos normalmente se realizan basados en tesauros. La desventaja de este método es que necesita todos los documentos y el proceso de actualización del tesauro puede ser costoso y complejo[3, 5, 6]. En los métodos basados en documentos parciales, se envía originalmente la consulta al motor de búsqueda, con los resultados entregados, se selección un grupo de los

documentos (los primeros resultados, los más relevantes) y con ellos se reformula la consulta (formula de Rocchio con γ=0) y se re-envía al motor. Los resultados de la segunda consulta (o consulta expandida) son los que realmente se le presentan al usuario[3, 5, 6].

En este proyecto se tomó un enfoque de expansión de consulta orientado por la Re-alimentación de relevancia del Usuario, proponiendo un enfoque de co-ocurrencia de términos relevantes y no relevantes, almacenados en el perfil del usuario. Estos términos se calificaran como relevantes o no de acuerdo a la evaluación que el usuario otorgue a los documentos que se le presenten en cada consulta, y se compararon los resultados de la propuesta de expansión con la tradicional propuesta de Rocchio (Estos tópicos se tratan en mayor detalle en la sección del marco teórico).

1.2 OBJETIVOS

Este proyecto tiene lugar en el área de la recuperación de información, y fue realizado bajo la dirección del grupo de investigación en Sistemas y Tecnologías de la Información STI de la Universidad Industrial de Santander y el Grupo de investigación en Tecnologías de la Información GTI de la Universidad del Cauca. A continuación se presentan los objetivos tal y como fueron aprobados en la Escuela de Ingeniería de Sistemas de la Facultad de Ingenierías Físico-Mecánicas junto con el trabajo realizado para dar cumplimiento a los mismos.

1.2.1 OBJETIVO GENERAL

Proponer un proceso de expansión de consulta (query expansión) basado en una matriz de co-ocurrencia de los términos relevantes y no relevantes (matriz que se

obtiene de la calificación que un usuario aporta a los documentos recuperados en consultas anteriores) en el marco de un meta buscador de documentos web (meta web search engine), comparando sus resultados frente a los tradicionalmente propuestos por Rocchio.

1.2.2 OBJETIVOS ESPECÍFICOS

OBJETIVO No.	DESCRIPCIÓN	ESTADO
1	Definir una función de evaluación de la importancia relativa de un término relevante o no en el perfil del usuario de un sistema de meta búsqueda web, basado en la cantidad de documentos evaluados, la cantidad de documentos evaluados como relevantes y la aparición del término en cada uno de los anteriores conjuntos.	Se presenta la función de evaluación que cumple con todos los requisitos expuestos en el objetivo, a partir de la página 39.
2	Establecer una matriz de co-ocurrencia de términos que represente la relación entre los términos relevantes y no relevantes que han sido definidos así, por la calificación previa que el usuario le da a los documentos que se le han presentado en consultas anteriores.	La matriz de co-ocurrencia de términos que se menciona en este objetivo, es indispensable para realizar todo el proceso de expansión de consulta, en especial para facilitar la lista de términos que le permiten autocompletar la consulta a un usuario en ECWEB. Tanto la matriz como la lista de autocompletar hacen parte del proceso de expansión de consulta que se presenta a partir de la página 89.

3	Proponer un proceso de expansión de consulta basado en la matriz de co-ocurrencia de términos y la función de evaluación de la importancia relativa de términos definidos previamente.	El proceso de expansión propuesto en este documento está basado en la función de evaluación y en la matriz de co-ocurrencia de términos. Toda la descripción se puede ver en detalle a partir de la página 88.
4	Modelar e implementar un meta buscador web (aplicación web) para consultas en inglés que use las APIs de Google, Yahoo! y Bing como fuentes iniciales de documentos, el proceso de expansión de consulta previamente propuesto, un esquema de evaluación de la relevancia de los documentos y las funciones básicas de registro, ingreso y salida.	El meta buscador que se implementó para dar cumplimiento a este objetivo tiene por nombre ECWEB, este meta buscador soporta consultas en inglés y en español. La descripción de ECWEB se puede apreciar a partir de la página 75.
5	Realizar una evaluación[1] del proceso de expansión desarrollado para el meta buscador web, basado en una medida común del área de investigación de recuperación de información, la precisión en los primeros k resultados (precisión at k), promediada de un conjunto de consultas realizadas por estudiantes universitarios, a los que adicionalmente se les evaluará el índice kappa, para medir la concordancia de las respuestas.	Esta evaluación se realizó con estudiantes de ingeniería de sistemas de la Universidad del cauca. En total se realizaron 3 pruebas, cada una de ellas constituidas por 3 consultas, además se calculó el grado de concordancia mediante el índice kappa de fleiss, estas pruebas se presentan a partir de la página 103.

[1] Con el objetivo de limitar los alcances de la investigación, las consultas, documentos y fuentes de información usados para la evaluación se limitarán al idioma inglés.

6	Comparar los resultados obtenidos por el proceso de expansión de consulta propuesto en el presente proyecto con los entregados por el método tradicional de Rocchio.	Se muestra como los algoritmos de expansión propuestos son superiores en la relevancia de los resultados recuperados cuando se comparan frente a la propuesta de Rocchio. Los resultados de las evaluaciones se aprecian a partir de la página 48) sobre dos colecciones de datos (CACM y LISA) muy usadas en recuperación de información.

1.3 RESULTADOS OBTENIDOS

La siguiente lista, resume los Productos finales del presente proyecto de investigación:

1. Prototipo funcional del algoritmo propuesto en http://minerva-search.net.

2. Artículo titulado “Algoritmos de Expansión de Consulta basados en una Nueva Función Discreta de Relevancia”, que refleja el algoritmo propuesto en la investigación y resultados experimentales del mismo, el cual será publicado en la Revista UIS Ingenierías a finales del año 2011.

3. Monografía de Trabajo de Grado. Hace referencia al presente documento y a los anexos del mismo, donde se presenta la descripción detallada de la investigación realizada.

2. MARCO TEORICO

2.1 RECUPERACIÓN DE INFORMACIÓN

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 [3]. Para entender mejor esta definición, es necesario pensar en ítems de información como documentos (normalmente desestructurados) que están relacionados con las solicitudes de búsqueda de un usuario [5].

La recuperación de información 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, y un despliegue de documentos ordenados por

relevancia y con una alta posibilidad de equivocarse en el orden de presentación de dichos documentos[3, 4, 6].

La recuperación de información ha tomado gran importancia desde 1940, y con el creciente uso de las computadoras se creó la posibilidad de manejar automáticamente grandes volúmenes de información. En este sentido, se ha definido una estructura general para un sistema de recuperación de información (SRI), compuesto principalmente por (ver Figura 1): Documentos (almacenados en bases de datos o directorios), Usuarios, Consultas (solicitudes), Resultados/Respuestas (documentos relacionados y ordenados por relevancia), Retroalimentación (del usuario al sistema) y el Proceso (software y hardware que realiza el proceso de recuperación de información) [3-5].

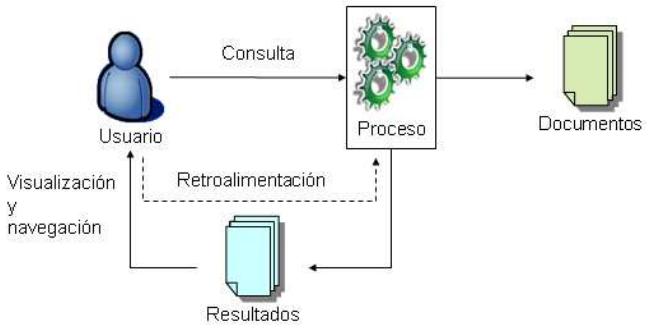


Figura 1. Componentes de un SRI. Adaptado de [3, 4]

Los temas centrales de investigación en recuperación de información iniciaron con la definición de mecanismos eficientes de almacenamiento (índices, índices ponderados, índices invertidos, índices probabilísticas, clasificación automática de palabras claves, discriminación y representación), clasificación automática, estructuras de archivos, estrategias de búsqueda (modelo booleano, modelo de espacio vectorial, funciones de concordancia, búsqueda serial, agrupamiento representativo, realimentación, re-consultas, modelo probabilístico) y evaluación (rendimiento y satisfacción del usuario) del sistema en una colección “controlada”

de documentos[3-5, 7]. Con el tiempo, y específicamente el cambio que ha impreso 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[8, 9]) ha tenido que tomar aportes conceptuales y metodológicos de una mayor cantidad de áreas de conocimiento. A este respecto, 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; mientras que el estudio de datos multimedia, el manejo de múltiples idiomas, la navegación y la visualización de los datos ha tomado mucha mayor importancia [3, 10, 11].

En la actualidad existen varios modelos de recuperación de información (RI), los más destacados [3, 4] son: el modelo booleano, el modelo de espacio vectorial y el modelo probabilístico. Además se encuentran algunas variaciones a estos tres primeros modelos, a saber: el modelo de conjuntos difusos, el modelo booleano extendido, el modelo del espacio vectorial generalizado, el modelo de indexado de semántica latente, el modelo de redes neuronales, el modelo de las redes bayesianas, el modelo de las redes de inferencia, el modelo de red de creencias, entre otros.

El modelo de espacio vectorial [3, 12] (VSM por sus siglas en inglés, Vector Space Model) es el que en general reporta mejores niveles de relevancia en los resultados que se le presentan a los usuarios. En este modelo se conciben los documentos como bolsas de palabras y la colección de documentos se representa con una matriz de M-términos por N-documentos. Cada documento se representa como un vector fila d en el espacio de términos tal que d= {w_1, w_2, …, w_M}(ver

Figura 2), donde $w_{i,j}$ es igual a la frecuencia del término (conocido como TF) normalizado en la colección multiplicado por la inversa de la frecuencia del documento (conocido como IDF) para ese término, en lo que se conoce como el valor TF-IDF que se resume en la (2) o una variación de la misma.

$$w_{i,j} = \frac{frecuencia_{i,j}}{\max(frequecia_i)} \times \log\left(\frac{N}{1+n_j}\right) \qquad \textbf{(2)}$$

	t_1	t_2	...	t_j	...	t_F
d_1						
d_2						
...						
d_i				$W_{i,j}$		
...						
d_N						

Figura 2. Matriz de Términos por Documentos

En este modelo de representación de documentos, se usa la distancia de cósenos para medir el grado de similitud entre dos documentos o entre un documento y la consulta del usuario, calculado por la formula (3) e ilustrado gráficamente en la Figura 3.

$$Sim\ (d,q) = Cos\ (\theta) = \frac{\sum_{i=1}^{M} W_{i,d} \times W_{i,q}}{\sqrt{\sum_{i=1}^{M} W^{2}_{i,d}}\ \sqrt{\sum_{i=1}^{M} W^{2}_{i,q}}} \qquad \textbf{(3)}$$

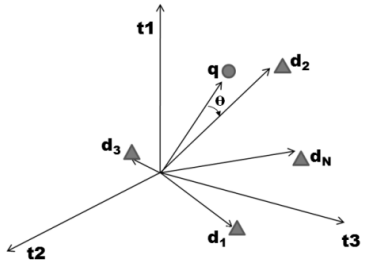


Figura 3. Similitud entre documentos y consultas

2.2 EXPANSIÓN DE CONSULTAS EN SRI

La expansión de la consulta en un sistema de búsqueda web normalmente se hace desde una de dos perspectivas: Realimentación de relevancia del Usuario (URF) o Realimentación automática de relevancia (ARF) [3, 5, 6]. URF requiere que el usuario marque los documentos como relevantes o no relevantes y luego, a cada nueva consulta del usuario se le agregan o quitan los términos que el sistema ha encontrado como relevantes o no en los documentos [3, 5, 6].

Por otro lado con URF, la realimentación automática de relevancia, también conocida como pseudo re-alimentación (pseudo feedback) expande las consultas automáticamente basados en dos métodos: documentos globales y documentos parciales [3, 5, 6], como ya se mencionó anteriormente y de forma más detallada (ver página 20).

Trabajos como los de Robertson & Sparck Jones[13, 14] que re-ponderan los términos de la consulta, o los de Dillon & Desper [15]que abandonan los términos del usuario y usan términos de los documentos inicialmente recuperados, son ejemplos de esta estrategia.

Nuevos enfoques incluyen entre otros: el etiquetado social (social tagging) [16, 17], como una estrategia que aprovecha la creciente popularidad de las redes sociales y los sistemas de etiquetado colaborativo. Estos enfoques extienden la familia de las bien conocidas matrices de co-ocurrencia; El uso de conocimiento semántico representado en ontologías [18, 19], a través del análisis de las relaciones de los conceptos y sus términos, las funciones, las instancias y los axiomas; y métodos que hibridan o mezclan varias técnicas, por ejemplo el uso de ontologías con filtrado colaborativo y redes neuronales artificiales [20].

En este proyecto se hace una propuesta desde el enfoque de realimentación de relevancia del usuario, que es libre de parámetros, contrario a la propuesta de Rocchio y que además es computacionalmente menos costosa, haciéndola una opción viable para la mayoría de aplicaciones y ambientes donde se recupera información usando el proceso de expansión de consulta.

2.3 ALGORITMO DE ROCCHIO

El algoritmo de Rocchio que se resume en la formula (1), se destacan varios elementos, el primero de ellos un conjunto de documentos evaluados como relevantes, un conjunto de documentos evaluados como no relevantes y una consulta expresada como vector de términos con pesos (no como la cadena de texto que usualmente digitan los usuarios en los sistemas de recuperación de información). Los conjuntos de documentos evaluados como relevantes y no relevantes también se deben expresar como vectores de términos con pesos.

En la práctica, el algoritmo no registra todos los documentos que han sido relevantes y no relevantes para cada usuario del sistema, en lugar de ello, el perfil almacena: un vector de términos representativo del documento relevante promedio, el total de documentos que han sido relevantes (|R|), un vector de términos representativo del documento no relevante promedio y el número total de documentos que han sido evaluados como no relevantes (|R|). En cada celda de los vectores se almacena el valor TF-IDF promedio de todos los documentos del conjunto, según la fórmula (2) o una de sus variaciones. Por ejemplo, en Lucene (framework vectorial para el desarrollo de aplicaciones de recuperación de

información) el valor que se almacena es la frecuencia observada de cada término en el documento y el valor IDF (Inverse Document Frecuency) es igual a $1 + \log\left(\frac{N}{n_i+1}\right)$, donde N es el número de documentos en la colección y n_i es el número de documentos en los que aparece el término.

Cuando se va a expandir una consulta, el texto inicial digitado por el usuario se convierte en un vector de términos similar a los vectores que representan a los documentos en el espacio multidimensional de términos y cada celda almacena el valor TF-IDF para los términos que digitó el usuario.

Luego, el vector de términos que representa la consulta del usuario se multiplica celda a celda por el valor α. Después, al resultado se le suma celda a celda (suma de vectores) el vector de términos representativos del documento relevante promedio previamente multiplicado por el parámetro β. Finalmente, al resultado se le suma celda a celda (suma de vectores) el vector de términos representativos del documento no relevante promedio previamente multiplicado por el parámetro γ. De esta forma la consulta textual digitada por el usuario se expande y el resultado es un vector de términos que representa los términos con sus pesos en el espacio multidimensional, el cual es comparado con los documentos mediante la similitud de cosenos de la formula (3), o una variación de la misma, para obtener el ranking de los documentos. Por ejemplo, en http://lucene.apache.org/java/2_9_0/api/core/org/apache/lucene/search/Similarity.html se puede apreciar la medida de similitud usada por Lucene.

2.4 EVALUACIÓN EN RECUPERACIÓN DE INFORMACIÓN

Los sistemas de recuperación de información, al igual que cualquier sistema software, deben ser evaluados antes de iniciar su funcionamiento en el ambiente real de producción. Dicha evaluación contempla aspectos como: análisis de funcionalidad, de unidad, de integridad, de tolerancia a fallos y de rendimiento (tiempo de respuesta al usuario, espacio requerido para almacenamiento adicional de índices de búsqueda, la velocidad de los canales de comunicación, entre otros). Pero uno de los aspectos más importantes en la evaluación de estos sistemas de recuperación de información es la precisión de la respuesta del sistema, conocida como la evaluación del rendimiento de la recuperación. Las medidas más conocidas y ampliamente usadas para realizar esta evaluación son la precisión y el recuerdo (exhaustividad) [3] y otras medidas derivadas de ellas.

La precisión corresponde a la fracción de los documentos recuperados por el sistema que realmente son relevantes para el usuario, formalmente definida por(4). En colecciones no controladas de documentos se cuenta con una medida derivada Precisión at K, que calcula la precisión de los documentos presentados al usuario en un rango K especifico de documentos, por ejemplo los primeros 5 documentos, 10 documentos y así sucesivamente.

$$precisión = \frac{\left|\{docuemntos\ \_\ relevantes\} \cap \{documentos\ \_\ recuperados\}\right|}{\left|\{documentos\ \_\ recuperados\}\right|} \quad \textbf{(4)}$$

El recuerdo o la exhaustividad corresponden a la fracción de los documentos relevantes que han sido recuperados por el sistema del total de documentos relevantes, formalmente definida por (5).

$$recuerdo = \frac{\left|\{documentos\ \_\ relevantes\} \cap \{documentos\ \_\ recuperado\ s\}\right|}{\left|\{documentos\ \_\ relevantes\}\right|} \quad \textbf{(5)}$$

La medida F es la media armónica de la precisión y la exhaustividad, formalmente definida por (6).

$$F = \frac{2 \times precisión \times recuerdo}{(precisión + recuerdo)} \quad \textbf{(6)}$$

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 como se muestra en (7).

$$kappa = \frac{P(A) - P(E)}{1 - P(E)} \quad \textbf{(7)}$$

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. Por lo anterior, es común usar en recuperación de información el estadístico Kappa de Fleiss, del cual se puede consultar información detallada en

http://en.wikipedia.org/wiki/Fleiss'_kappa. La Tabla 1 muestra el intervalo de valores que puede llegar a tomar el índice Kappa de Fleiss.

κ	Interpretation
< 0	Poor agreement
0.01 – 0.20	Slight agreement
0.21 – 0.40	Fair agreement
0.41 – 0.60	Moderate agreement
0.61 – 0.80	Substantial agreement
0.81 – 1.00	Almost perfect agreement

Tabla 1. Kappa de Fleiss – intervalo de valores

2.5 ÍNDICES TEMÁTICOS, MOTORES DE BÚSQUEDA Y META BUSCADORES WEB

La búsqueda web puede ser vista como un área de aplicación más amplia para los conceptos involucrados en los sistemas de recuperación de información originales. Los componentes de un sistema de búsqueda web son similares a los de un SRI y en esta propuesta es de especial importancia el buscador web, que es el encargado de hacer el proceso automático de representación, organización y recuperación de documentos distribuidos en Internet. Estos buscadores web presentan al usuario una interfaz para que ingresen las peticiones (consultas sobre un tema, normalmente a través de un conjunto de palabras clave), el sistema realiza la búsqueda y devuelve los enlaces para que el usuario los analice, acceda a ellos y decida si le sirven o no. Existen tres tipos principales de buscadores: índices temáticos o directorios web, motores de búsqueda y meta buscadores[21].

Los índices temáticos o directorios son listas de recursos organizadas en jerarquías desde lo más general a lo más específico. Normalmente el proceso de clasificación se hace de forma manual. Los directorios tienen las siguientes ventajas: son fáciles de usar para usuarios no experimentados, la búsqueda se realiza eligiendo la categoría que más se acerca a la consulta y se desciende en la jerarquía hasta encontrar los enlaces de los recursos deseados y hay menos ruido en los recursos. Pero también tiene algunas desventajas: sólo cubren una pequeña parte de los recursos de la web y no existen criterios homogéneos para la clasificación y selección de dichos recursos. Por ejemplo: Yahoo! (www.Yahoo!.com), terra (www.terra.es), galaxy (www.galaxy.com) y dmoz (http://www.dmoz.org).

Los motores de búsqueda recorren la red recolectando e indexando la mayor cantidad de información posible, basados en programas automáticos conocidos como robots (spider o crawler). Las principales ventajas de los motores de búsqueda son: los procesos de recolección e indexación son automáticos, por lo que se recoge gran cantidad de información y además pueden contar con métodos de actualización automática. Entre las principales desventajas están: los robots tienen restricciones de navegación sobre la web profunda [22] ya que sus contenidos son generados dinámicamente mediante consultas que deben ser autenticadas y autorizadas, entre otras cosas, y por esto, ellos solo recorren la web superficial; además estos motores son más complejos de usar para el usuario novato, ya que el usuario debe conocer la sintaxis de búsqueda del motor y debe ser extremadamente cuidadoso cuando realiza una consulta para obtener resultados óptimos (proceso de refinación de la búsqueda); finalmente, no existe un proceso “controlado” de calidad y fiabilidad de los recursos. Por ejemplo: Google (www.google.com) y Altavista (www.altavista.com).

Los meta buscadores web son sistemas de búsqueda que no disponen de bases de datos propias, y por eso buscan en otros buscadores (normalmente en motores de búsqueda web). Ellos recogen la petición del usuario y la envían a los buscadores web, éstos devuelven los resultados y los meta buscadores los clasifican antes de presentarlos al usuario (lo que incluye entre otras cosas un proceso de reordenamiento y filtrado[23]). Entre las ventajas más importantes se pueden mencionar, que la búsqueda es más extensiva, el usuario accede a una sola página para formular la consulta y esta consulta se digita una sola vez. Una desventaja es que al momento de formular la consulta, es posible que la sintaxis no sea la más adecuada para cada uno de los buscadores que se usan en el fondo y además que el proceso de búsqueda es un poco más lento[21]. Por ejemplo: Ixquick (http://www.ixquick.com), DogPile (http://www.dogpile.com), Webferret (http://www.Webferret.com), Copernic (http://www.copernic.com), metacrawler (http://www.metacrawler.com), Monster Crawler (http://monstercrawler.com) y mamma (http://www.mamma.com).

2.6 HERRAMIENTAS DE PROGRAMACIÓN

MICROSOFT VISUAL STUDIO .NET 2010: Es un componente de Microsoft que brinda un completo conjunto de soluciones para la programación de aplicaciones y la ejecución de las mismas. .NET cuenta entre otras cosas, con un marco de trabajo (**framework**) que trabaja con independencia de la plataforma hardware y que permite un rápido desarrollo de aplicaciones [24]. **C#**, es un lenguaje de programación orientado a objetos que hace parte de la plataforma .NET, desarrollado por Microsoft, aceptado a nivel mundial como estándar por la EMAC e ISO y normalizado por la EMAC [25].

ASP.NET: Es un framework para aplicaciones web desarrollado y comercializado por Microsoft. Es usado por programadores para construir sitios web dinámicos, aplicaciones web y servicios XML [26].

MICROSOFT SQL SERVER 2008: Es un sistema (motor) objeto relacional para la gestión de bases de datos producido por Microsoft [27]. Entre las características más importantes de Microsoft SQL Server están:

- Estabilidad, escalabilidad y seguridad.
- Uso de comandos DML y DDL mediante un práctico y potente entorno grafico
- Soporta a procedimientos almacenados
- Soporte a transacciones
- Modo cliente/servidor
- Administración de información de servidores remotos desde una consola gráfica centralizada

Además, la versión 2008 de Microsoft Sql Server, es muy segura, se integra con PowerShell, tiene capacidades de compresión, auditoria y encriptación transparente de datos.

Las herramientas descritas anteriormente fueron utilizadas para la realización del meta buscador web; VS.NET 2010 fue utilizado como entorno de desarrollo, C# como lenguaje de programación, ASP.NET se usó para la programación web y Microsoft Sql Server 2008 como motor de base de datos. Estos recursos están disponibles de forma gratuita para el desarrollo de proyectos de I+D en el marco del programa de Microsoft MSDN Academic Alliance.

3. ALGORITMOS PROPUESTOS

Con base en un estudio detallado de los resultados de un algoritmo de recuperación de información tradicional (basado en TF-IDF y similitud de cosenos), el algoritmo de expansión de consultas propuesto por Rocchio y el análisis de las frecuencias de los términos en los diferentes conjuntos de documentos (relevantes y no relevantes) entregados al usuario, se definió una nueva función para el valor IDF de cada término en el perfil de un usuario de un sistema de recuperación de información o búsqueda web.

3.1 FUNCIÓN IDF

Esta función IDF, ver fórmula (8), define la importancia de un término en relación con el número de documentos evaluados por el usuario (N), el número de documentos relevantes para el usuario (R), el número de documentos en los que aparece el término i (n_i) y el número de documentos relevantes en los que aparece el término i (r_i).

$$idf_i = \begin{Bmatrix} \frac{r_i}{N} \cdots & Si & n_i \le R \\ \frac{r_i * R}{n_i * N} \cdots & Si & n_i > R \end{Bmatrix} \quad \textbf{(8)}$$

La función IDF propuesta en esta investigación, ver Figura 4, tiene un rango de valores discreto entre cero y uno [0,1]. Cero cuando el término no es relevante en absoluto y uno cuando es totalmente relevante. El grado de relevancia esta en relación con el radio de documentos relevantes, es decir, si existen muchos documentos evaluados (por ejemplo en la gráfica, la serie de datos con marcador en forma de rectángulo, N=50) y de ellos el término aparece en sólo unos documentos (por ejemplo 10) y todos son relevantes, la función IDF alcanza un valor de 0.2, en contraste con un número menor de documentos (por ejemplo en la gráfica, la serie de datos con marcador en forma circular, N=10), donde obtendría un valor de 1.0 (valor máximo).

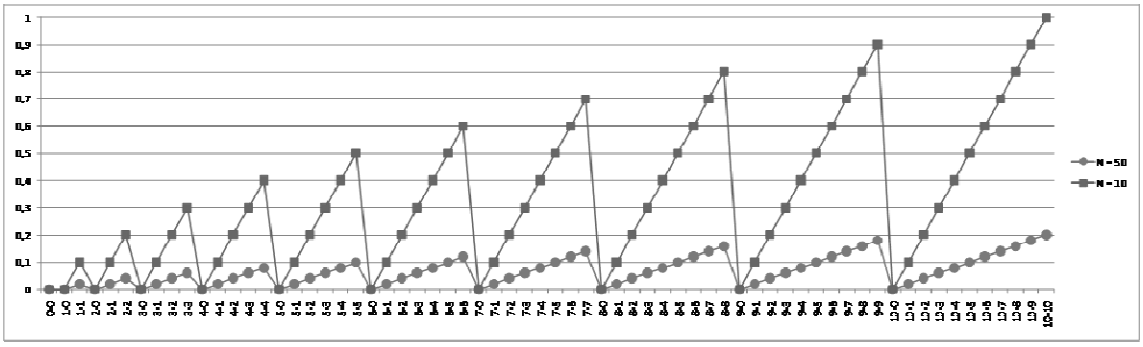


Figura 4. Función IDF

El eje X muestra distintos valores de n_i y r_i, iniciando con (0-0), pasando por ejemplo por (7,3) y finalizando con (10-10). En esta gráfica se muestran valores de n_i y de r_i entre 0 y 10. Para las dos series de datos, se logra el máximo cuando $n_i = r_i$, en este caso (10,10) y el mínimo cuando $r_i = 0$, sin importar el valor de n_i.

3.2 ALGORITMO CON VECTOR PONDERADO (VT-IDF)

Con base en esta función IDF se planteó un algoritmo de expansión de consulta. Este algoritmo recibe como entrada la consulta del usuario y entrega como resultado una consulta expandida con pesos para cada uno de los términos contenidos en dicha consulta. Se parte del hecho, de que cada documento evaluado (relevante o no) por el usuario, modifica la función IDF y otros datos en el perfil del usuario.

El perfil del usuario está compuesto por los elementos que se muestran en la Figura 5, a saber: El número (N) de documentos que el usuario ha evaluado, el número (R) de documentos evaluados como relevantes y una ***lista de términos del usuario***, en la cual se registra por cada término que ha aparecido en los documentos que ha evaluado el usuario, el número (n_i) de veces que el términos especifico ha aparecido en los documentos evaluados, el número (r_i) de veces que el término ha aparecido en los documentos relevantes y el valor IDF para cada término.

Cuando el usuario realiza una consulta ($q_{inicial}$) por palabras claves se realiza un primer paso de pre-procesamiento de dicha consulta, lo que implica: Tokenización (división de la cadena de consulta en términos individuales), remoción de acentos y caracteres especiales, paso a minúsculas, remoción de palabras vacías (basado en una lista de palabras vacías) y lematización (extracción de la raíz léxica del término) con un algoritmo como el propuesto por Porter[28]. Como resultado se obtiene la consulta inicial procesada. Por ejemplo, si la consulta inicial $q_{inicial}$ es igual a “What articles exist which deal with TSS (Time Sharing System), an operating system for IBM computers?”, el resultado, $q_{inicial\text{-}procesada}$ es igual a “articl exist deal tss time share system oper system ibm comput”.

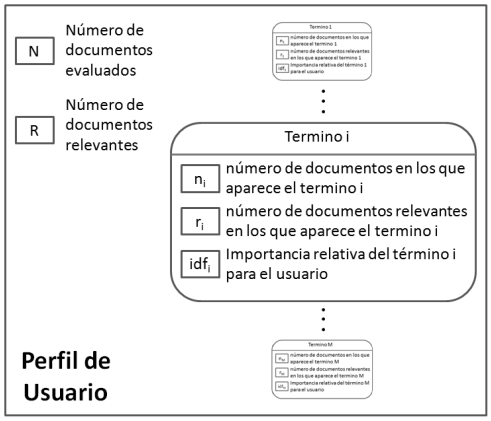


Figura 5. Perfil de Usuario

La consulta inicial procesada, se expande con el algoritmo que se ha denominado, vector ponderado según IDF (VT-IDF) y que se explica a continuación:

1. Se consulta de la ***lista de términos del usuario*** en el perfil de usuario, todos los términos que cumplan con la siguiente condición: $r_i > 0.5 * n_i$. Es decir, todos los términos que hayan aparecido más número de veces en los documentos relevantes que en los documentos no relevantes. Una parte de la lista de términos que cumplen con este criterio para un usuario puede ser, LTC = {{"satisfactori", 1, 1, 0.25},{"tss", 1, 1, 0.25},{"sequenc", 1, 1, 0.111112}, …}. La lista de términos que cumplen con la condición previa se denomina ***lista de términos candidatos*** (LTC).

2. Se divide la cadena de la consulta inicial procesada ($q_{inicial\text{-}procesada}$) en tokens o términos independientes y por cada uno de ellos haga lo siguiente:

   2.1. Si el término se encuentra en la LTC se modifican los valor del término en dicha lista de la siguiente forma: $n_i = n_i + 1$, $r_i = r_i + 1$ y se recalcula el valor IDF de ese término con los nuevos valores de n_i y r_i. Estas

operaciones aumentan la relevancia del término en la consulta expandida final, ya que el término está en el perfil y además está en la consulta inicial.

2.2. De otro modo (es decir, si el término NO se encuentra en la LTC), se inserta un nuevo nodo en la LTC con el texto del término, un valor de uno (1) para n_i y r_i, y se calcula el valor de IDF (con la función propuesta) para este nuevo término. Si el valor IDF es igual a cero (0), es decir no ha aparecido en ningún documento evaluado como relevante, se asigna el valor IDF general de la colección de datos. Si este nuevo IDF sigue siendo cero, el término no se adiciona, ya que, es un término que no existe en la colección de documentos y por esta razón no sirve como término de búsqueda. De esta forma, la lista de términos candidatos consultada del perfil, se complementa con términos de consulta nuevos digitados por el usuario en la consulta especifica.

3. Se recorre la lista de términos candidatos y se genera la consulta expandida, teniendo en cuenta la opción de boosting definida en Lucene, es decir, representando la consulta como un vector en el espacio multidimensional de términos. Cada término se anexa a una cadena de texto de salida usando el formato “término^peso”, donde término es cada uno de los términos de la lista de candidatos que supera el valor IDF y el peso es igual al valor IDF del término multiplicado por n_i (el número de documentos en los que aparece el término). A continuación se presenta un ejemplo de una consulta expandida final o $q_{expandida}$ = “satisfactori^0.25… tss^0.2… sequenc^0.1111111 paper^0.3157895… articl^0.05 exist^0.05 deal^0.05 time^0.05 share^0.05 system^0.2 oper^0.05 ibm^0.05 comput^0.05”.

VT-IDF al igual que Rocchio entrega como resultado un vector de términos ponderados, donde cada término tiene un peso en el espacio multidimensional de términos por documentos. A diferencia de Rocchio que usa todos los términos de la colección de documentos, en VT-IDF se tienen en cuenta sólo los que son más relevantes.

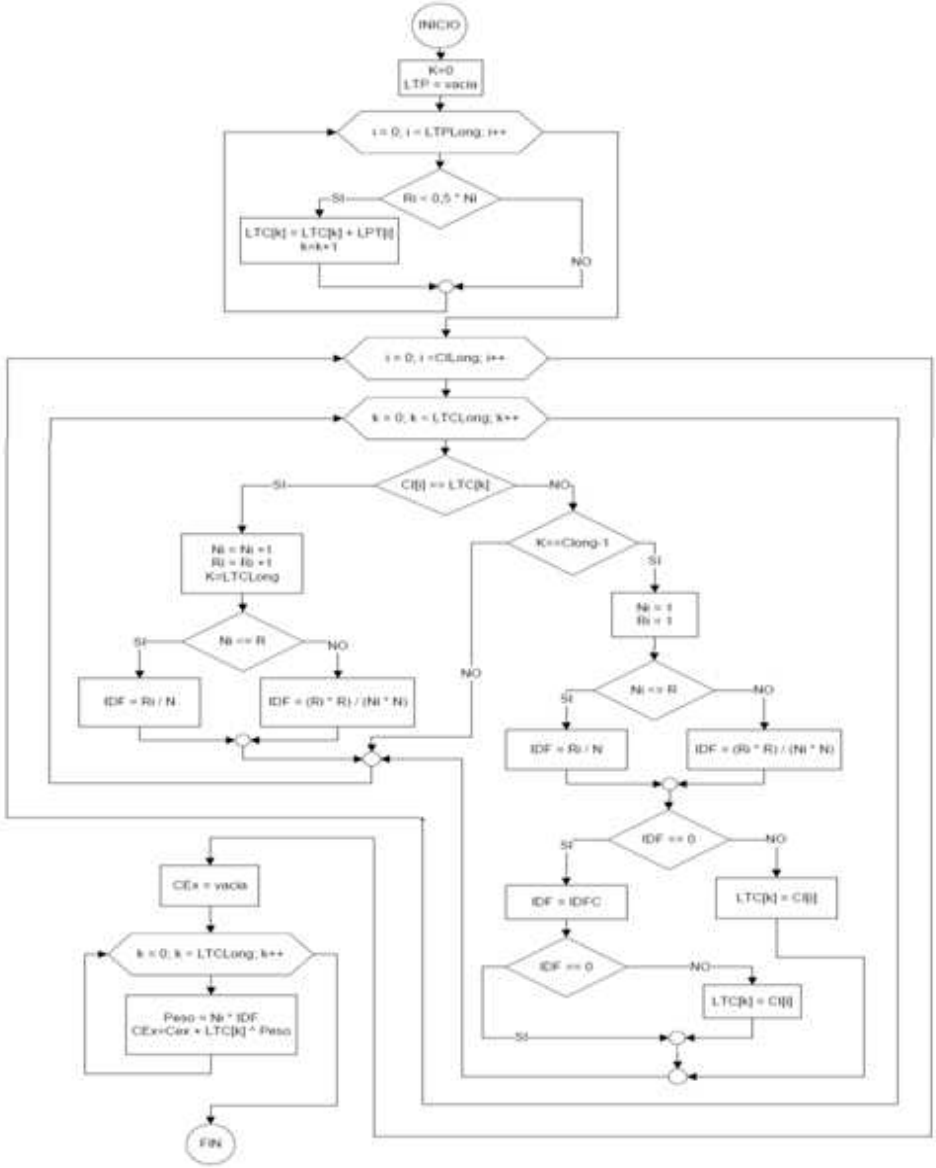


Figura 6 VT-IDF Diagrama de flujo

Dónde:

LTP: Es la lista de términos del perfil del usuario.

LTPLong: Es la longitud de la lista de términos del perfil del usuario (número de términos que contiene).

LTC: Es la lista de términos candidatos para la expansión de búsqueda.

LTCLong: Es la longitud de la lista de términos candidatos.

R: Es el número de documentos relevantes.

N: Es el número total de documentos.

Ri: Es el número de documentos relevantes donde aparece el término i.

Ni: Es el número de documentos en los que aparece el término i.

IDF: Es la importancia del término en relación con el número de documentos evaluados por el usuario.

Ci: Consulta inicial (consulta digitada por el usuario)

CiLong: Es la longitud de la consulta inicial (número de términos que contiene)

CEx: Consulta expandida.

3.3 ALGORITMO CON CADENA EXPANDIDA (CE-IDF)

Teniendo en cuenta las características de la función de IDF previamente definida y buscando generar una consulta expandida compuesta solamente de términos (evitando el boosting) se diseñó una variante del algoritmo VT-IDF, el cual se ha denominado cadena expandida según IDF (CE-IDF), que opera sobre la consulta inicial procesada como sigue:

1. Se consulta de la ***lista de términos del usuario*** en el perfil de usuario, todos los términos que cumplan con la siguiente condición: $r_i > 0.5 * n_{i,}$ formando la LTC. Este paso es igual al primer paso de VT-IDF.

2. La cadena de la consulta final expandida $q_{expandida}$ es igual a la cadena de la consulta inicial, $q_{inicial\text{-}procesada}$ concatenada con los términos de la LTC. En este algoritmo el resultado de la consulta del ejemplo sería igual a “articl exist deal tss time share system oper system ibm comput satisfactory… tss… sequenc… paper…”. Nótese que los términos se repiten como en un documento de texto cualquiera.

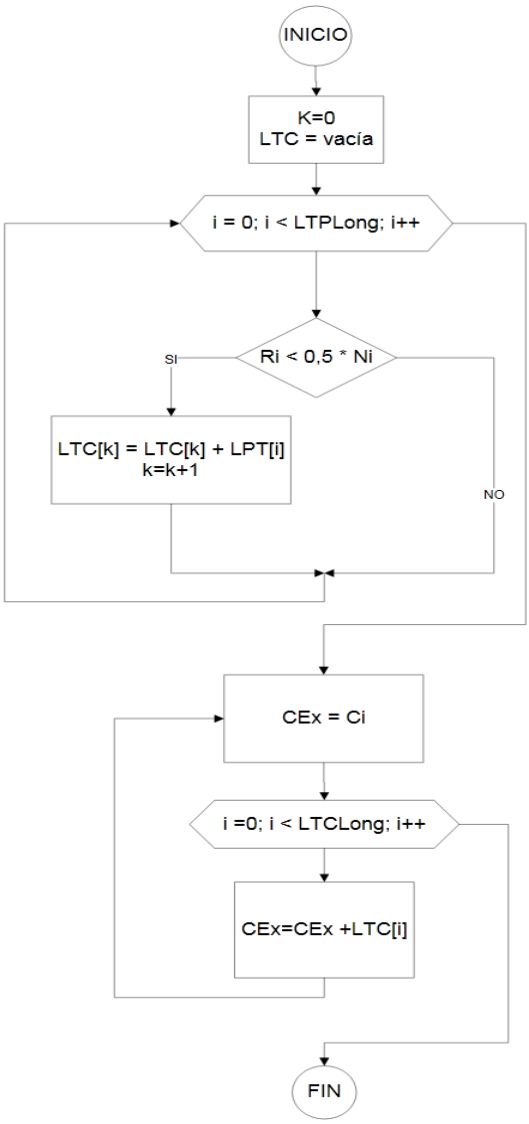


Figura 7 CE-IDF Diagrama de flujo

Dónde:

LTP: Es la lista de términos del perfil del usuario.

LTPLong: Es la longitud de la lista de términos del perfil del usuario (número de términos que contiene).

LTC: Es la lista de términos candidatos para la expansión de búsqueda.

LTCLong: Es la longitud de la lista de términos candidatos.

Ci: Consulta inicial (consulta digitada por el usuario)

CEx: Consulta expandida.

3.4 EVALUACIÓN

Como se puede apreciar anteriormente, las medidas de precisión y recuerdo están basadas en conjuntos cerrados (documentos, consultas y documentos relevantes para cada consulta); con el paso del tiempo ellas fueron adaptadas para evaluar sistemas que muestran resultados en una lista ordenada de documentos (como la mayoría de buscadores web de hoy en día), donde se espera que los primeros documentos estén más relacionados (sean más relevantes) con las necesidades del usuario. Una de estas adecuaciones es la curva de precisión-recuerdo, que en forma gráfica representa el valor de la precisión a diferentes niveles de recuerdo [3, 5, 29]. Esta curva permite comparar visualmente el rendimiento de dos o más sistemas de recuperación de información. La Figura 8 muestra un ejemplo de un gráfico de una curva de precisión recuerdo. En ella se puede ver que el sistema presenta una precisión aproximada de 50% cuando obtiene el 10% de recuerdo (cuando ha recuperado el 10% del total de los documentos relevantes a la consulta del usuario). Además muestra que esta curva, en general, es descendente; es decir a mayor valor de recuerdo se obtienen menores valores de precisión.

Con el objetivo de verificar el rendimiento de los algoritmos propuestos en el presente trabajo, se compararon los resultados de los mismos frente a la medida básica de ranking usada por Lucene (basada en similitud de cosenos) y el algoritmo de re-alimentación de relevancia del usuario propuesto por Rocchio [3, 5, 29]. Para este último se toman los siguientes valores para los parámetros de este algoritmo: α = 50%, β = 50% y γ = 0%. Valores que reportaron los mejores resultados en cuatro de los cinco experimentos realizados.

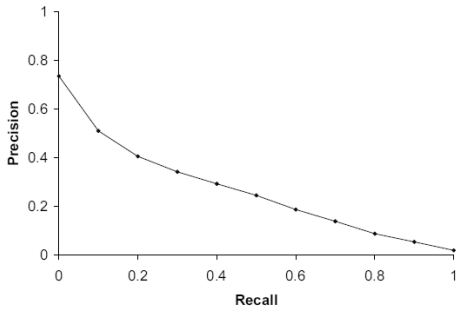


Figura 8. Curva de Precisión-Recuerdo (Tomada de [5])

3.5 EXPERIMENTO CACM

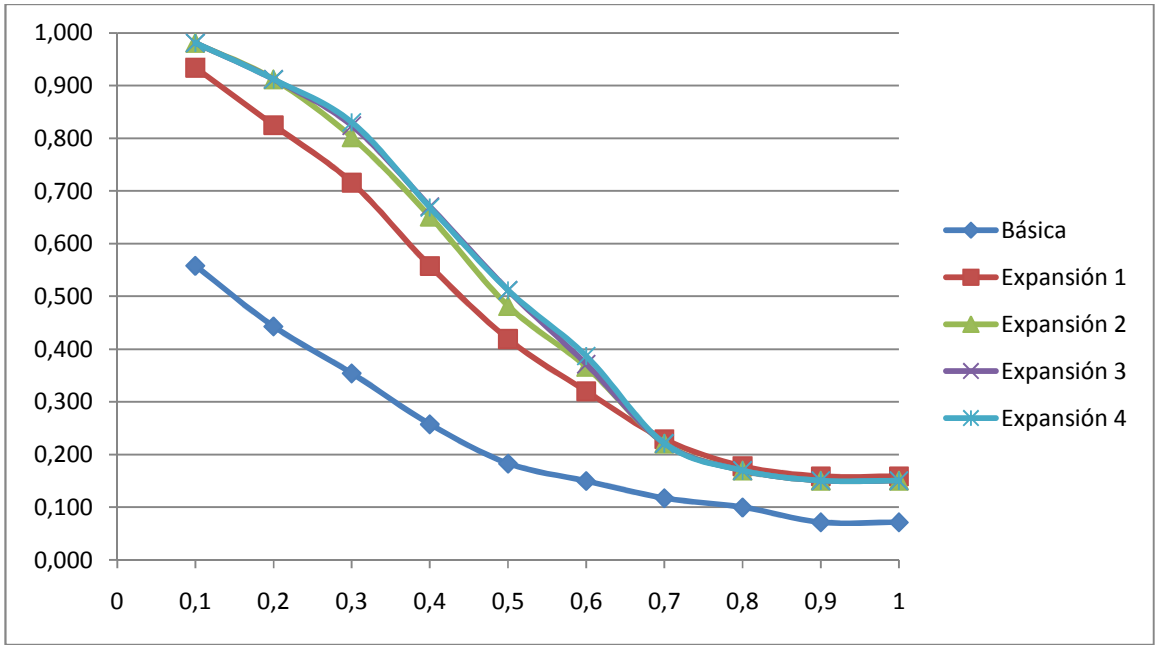
El conjunto de datos (dataset) usado para el primer experimento fue la colección de textos CACM disponible en forma gratuita en http://ir.dcs.gla.ac.uk/resources/test_collections (Colecciones de prueba del Grupo de I+D en Recuperación de Información de la Universidad de Glasgow en Escocia, Reino Unido). 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.

Con estos datos se simuló la ejecución de cada consulta cinco veces, la primera, denominada “Básica” que 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, 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 específico. Es de notar que la memoria del perfil del usuario en este caso sólo dura de una solicitud de consulta a otra (en adelante se denomina, sin memoria del perfil de usuario). Los resultados de este experimento se presentan en la Figura 9.

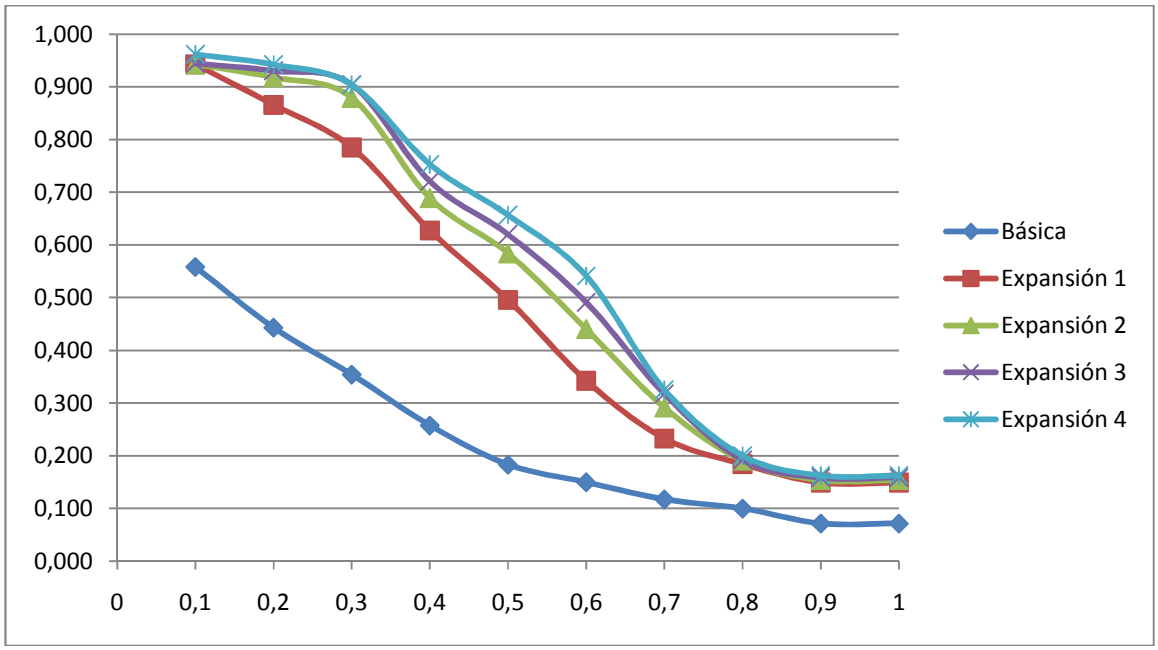
En las tres gráficas de la Figura 9 (a, b y c) se muestra el resultado de la consulta básica usando Lucene, que inicia en un 56% de precisión para un nivel de recuerdo de 10%, y decrece hasta un 7% cuando el nivel de recuerdo es del 100%. Luego en las líneas con marcador rectangular se muestra el resultado de la expansión 1, mostrando una mejora apreciable en los tres algoritmos, llegando a un promedio de 94% de precisión en el primer nivel de recuerdo y cayendo a un promedio de 16% en el último nivel de recuerdo. Este primer proceso de expansión, muestra una curva de precisión-recuerdo que es muy superior en todos los niveles de recuerdo en la consulta básica. Además muestra como los tres algoritmos siguen mejorando poco a poco en la expansión 2, 3 y 4.

En la Tabla 2 se muestran en detalle los valores de la Figura 9. Se muestra como VT-IDF logra desde la expansión 1 una precisión de 94% en el 10% de recuerdo y como en la expansión 4 alcanza un 96%. Mientras que Rocchio logra un 93% en la primer expansión y un máximo de 98% en la expansión 4. Finalmente, muestra

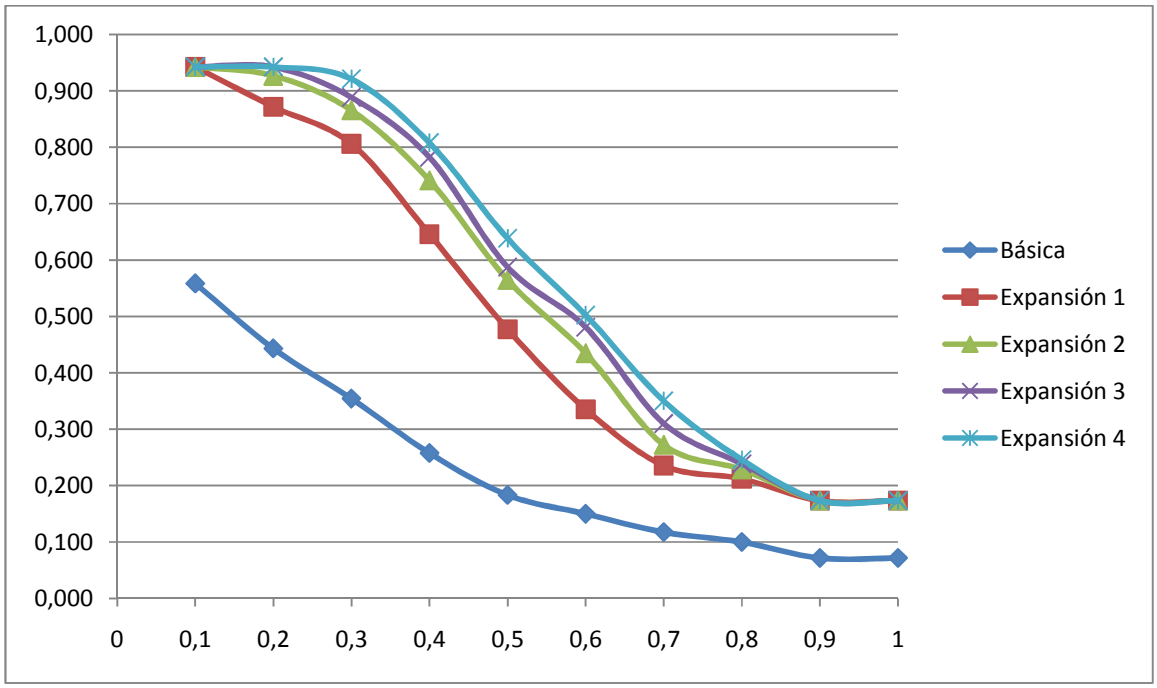
que CE-IDF logra un valor inicial y final de 94% en las 4 expansiones, pero logrando mejores resultados en los niveles de recuerdo del 20%, 30% y 40%.



(a) Rocchio



(b) VT-IDF



(c) CE-IDF

Figura 9. Curva de precisión-recuerdo para Rocchio (a), VT-IDF (b) y CE-IDF (c) sobre CACM sin memoria del perfil

Tabla 2. Valores de precisión -recuerdo para Rocchio, VT-IDF y CE-IDF sobre CACM sin memoria del perfil

	Recuerdo	**10%**	**20%**	**30%**	**40%**	**50%**	**60%**	**70%**	**80%**	**90%**	**100%**
Básica	Lucene	0,56	0,44	0,35	0,26	0,18	0,15	0,12	0,10	0,07	0,07
Expansión 1	CE-IDF	**0,94**	**0,87**	**0,81**	**0,65**	0,48	**0,34**	**0,24**	**0,21**	**0,17**	**0,17**
	VT-IDF	**0,94**	**0,87**	0,78	0,63	**0,50**	**0,34**	0,23	0,18	0,15	0,15
	Rocchio	0,93	0,82	0,72	0,56	0,42	0,32	0,23	0,18	0,16	0,16
Expansión 2	CE-IDF	0,94	**0,93**	0,87	**0,74**	0,56	0,43	0,27	**0,23**	**0,17**	**0,17**
	VT-IDF	0,94	0,92	**0,88**	0,69	**0,58**	**0,44**	**0,29**	0,19	0,15	0,15
	Rocchio	**0,98**	0,91	0,80	0,65	0,48	0,37	0,22	0,17	0,15	0,15
Expansión 3	CE-IDF	0,94	**0,94**	0,89	**0,78**	0,59	0,48	0,31	**0,24**	**0,17**	**0,17**
	VT-IDF	0,94	0,93	**0,90**	0,72	**0,62**	**0,49**	**0,32**	0,19	0,16	0,16
	Rocchio	**0,98**	0,91	0,82	0,67	0,51	0,37	0,22	0,17	0,15	0,15

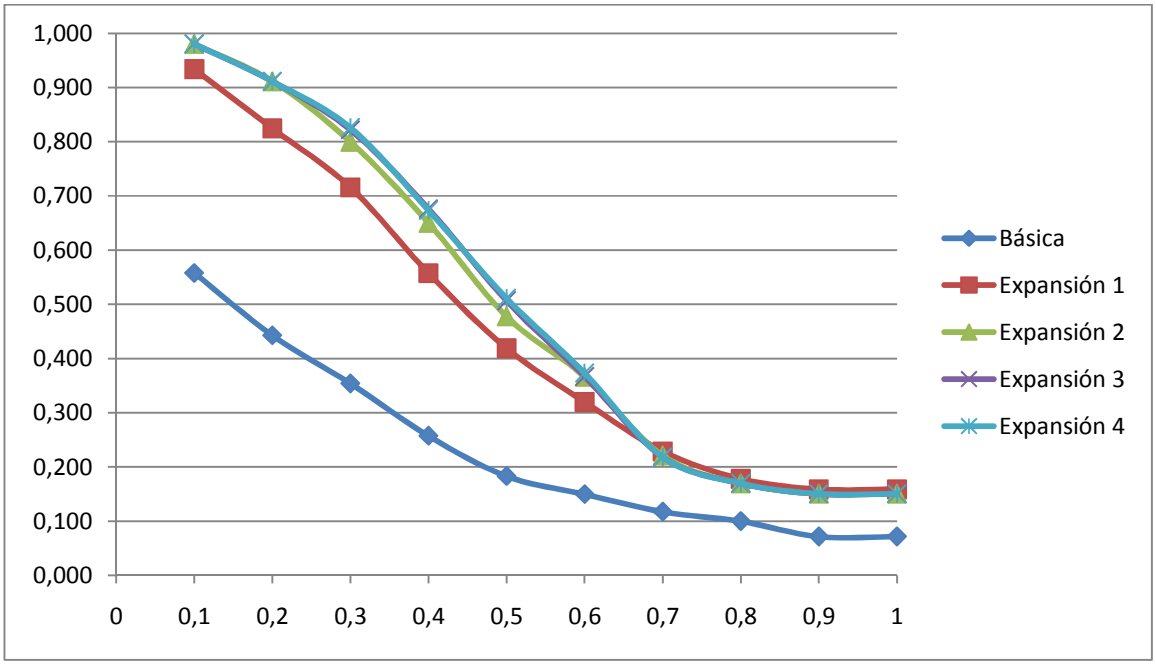
Expansión 4	CE-IDF	0,94	**0,94**	**0,92**	**0,81**	0,64	0,50	**0,35**	**0,25**	**0,17**	**0,17**
	VT-IDF	0,96	**0,94**	0,90	0,75	**0,66**	**0,54**	0,33	0,20	0,16	0,16
	Rocchio	**0,98**	0,91	0,83	0,67	0,51	0,39	0,22	0,17	0,15	0,15

En este primer experimento queda demostrado como el uso de la expansión de una consulta con base en la relevancia de los resultados previamente presentados al usuario, puede mejorar ostensiblemente los resultados del sistema. También muestra que para la colección de datos seleccionada el algoritmo de Rocchio obtiene mejores resultados en los primeros niveles de recuerdo, pero que VT-IDF y CE-IDF obtienen resultados muy similares.

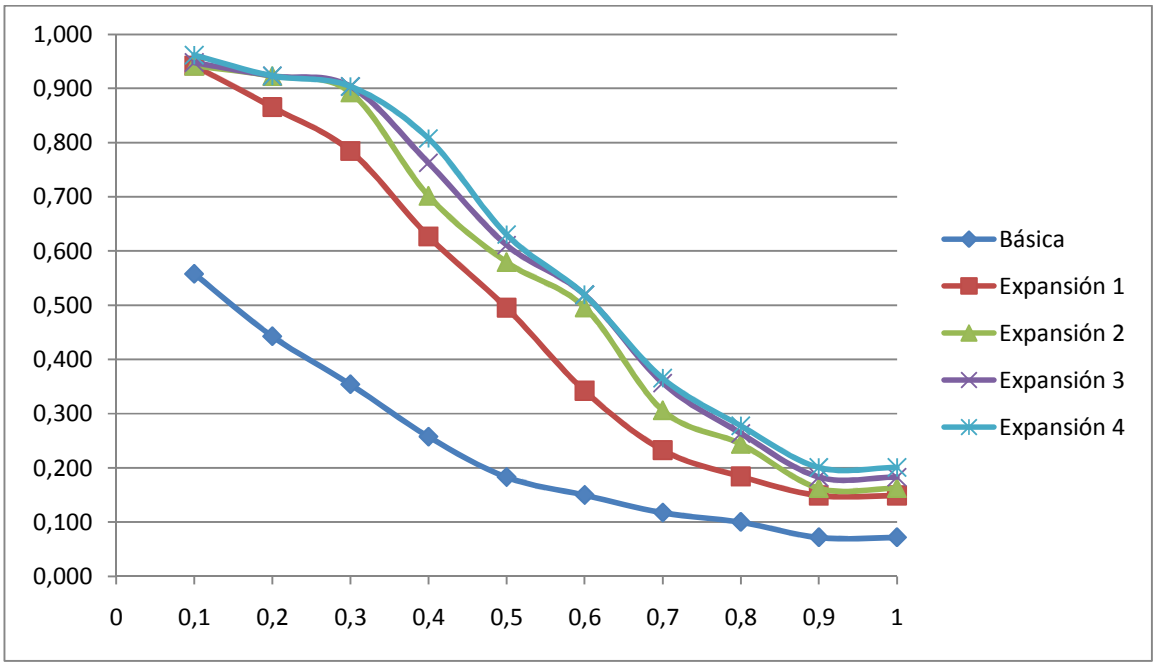
Un segundo experimento se llevó a cabo sobre la misma colección de datos. El proceso seguido fue el mismo del experimento anterior, pero en este caso el perfil del usuario mantuvo memoria en las cinco ejecuciones de la misma consulta. Este proceso simula el almacenamiento del perfil de un usuario durante una sesión de consulta de un tema. Los resultados de este experimento se presentan en la Figura 10.

En las tres gráficas de la Figura 10 (a, b y c), se muestra el resultado de la consulta básica usando Lucene, luego en las líneas con marcador rectangular se muestra el resultado de la expansión 1, mostrando una mejora apreciable en los tres algoritmos, llegando a un promedio de 94% de precisión en el primer nivel de recuerdo y cayendo a un promedio de 16% en el último nivel de recuerdo. Este primer proceso de expansión muestra una curva de precisión-recuerdo que es evidentemente muy superior en todos los niveles de recuerdo a la consulta básica. Además se muestra como Rocchio, VT-IDF y CE-IDF aprovechan la mayor información del perfil para mejorar la precisión de los resultados, expansión tras expansión, en los diferentes niveles de recuerdo.

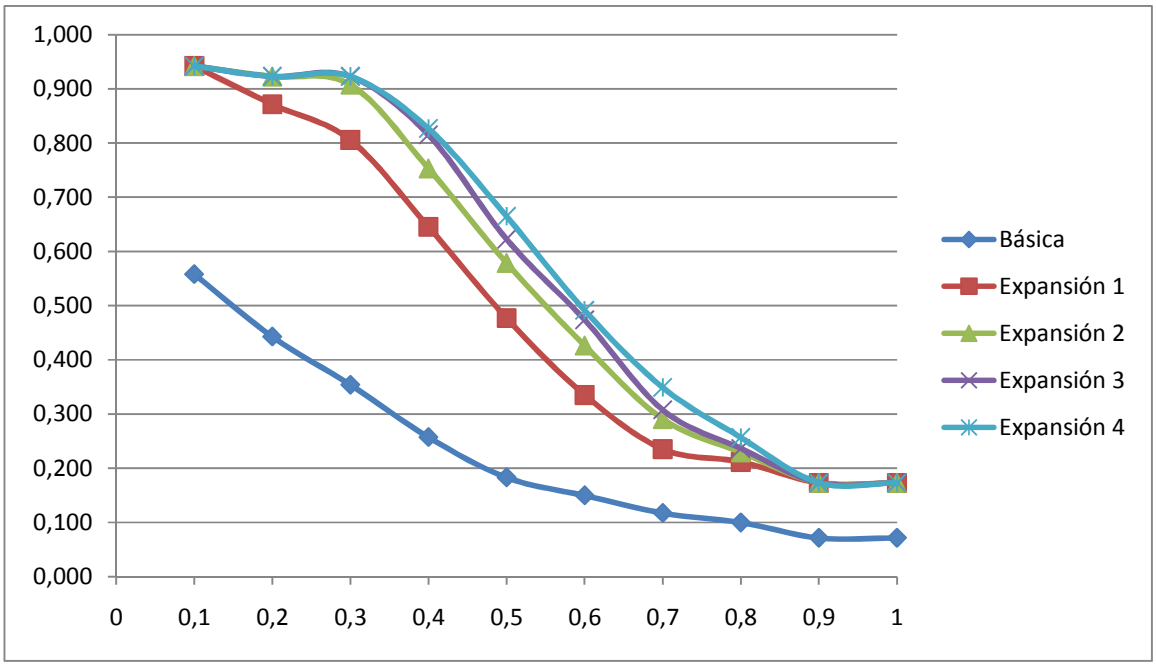
En la Tabla 3 se muestran en detalle los valores de la Figura 10. Se muestra como VT-IDF logra desde la expansión 1 una precisión de 94% en el 10% de recuerdo y como en la expansión 4 alcanza un 96%. Mientras que Rocchio logra un 93% en la primer expansión y un máximo de 98% en la expansión 4. Finalmente, muestra que CE-IDF logra un 94% en el primer nivel de recuerdo en todas las expansiones. De igual forma que en el experimento anterior, este algoritmo obtiene consistentemente mejores niveles de precisión en los niveles 20%, 30% y 40% de recuerdo.



(a) Rocchio



(b) VT-IDF



(c) CE-IDF

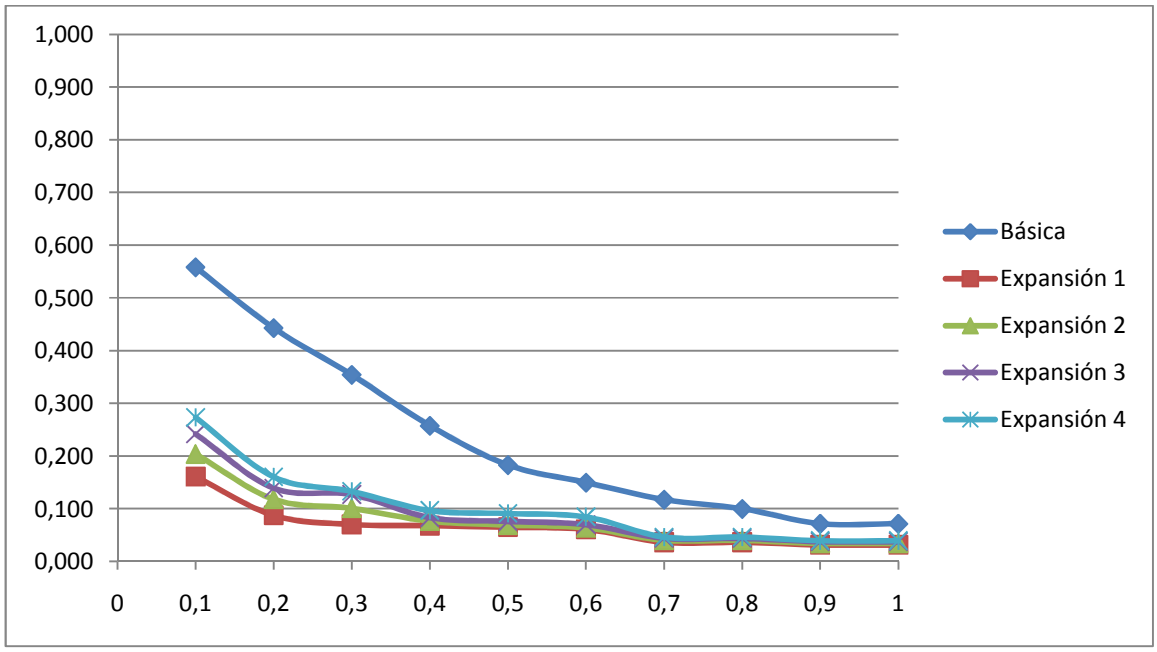
Figura 10. Curva de precisión-recuerdo para Rocchio (a), VT-IDF (b) y CE-IDF (c) sobre CACM con memoria de sesión

Tabla 3. Valores de precisión -recuerdo para Rocchio, VT-IDF y CE-IDF sobre CACM con memoria de sesión

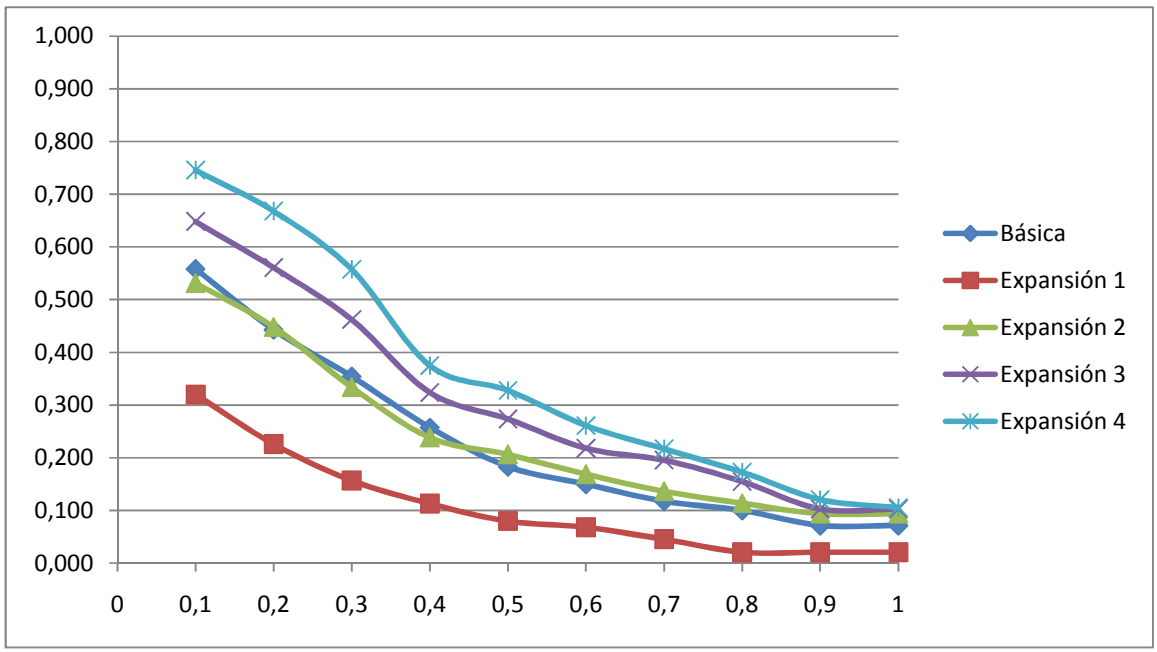
	Recuerdo	**10%**	**20%**	**30%**	**40%**	**50%**	**60%**	**70%**	**80%**	**90%**	**100%**
Básica	Lucene	0,56	0,44	0,35	0,26	0,18	0,15	0,12	0,10	0,07	0,07
Expansión 1	CE-IDF	**0,94**	**0,87**	**0,81**	**0,65**	0,48	**0,34**	**0,24**	**0,21**	**0,17**	**0,17**
	VT-IDF	**0,94**	**0,87**	0,78	0,63	**0,50**	**0,34**	0,23	0,18	0,15	0,15
	Rocchio	0,93	0,82	0,72	0,56	0,42	0,32	0,23	0,18	0,16	0,16
Expansión 2	CE-IDF	0,94	**0,92**	**0,91**	**0,75**	**0,58**	0,43	0,29	0,23	**0,17**	**0,17**
	VT-IDF	0,94	**0,92**	0,89	0,70	**0,58**	**0,50**	**0,31**	**0,24**	0,16	0,16
	Rocchio	**0,98**	0,91	0,80	0,65	0,48	0,37	0,22	0,17	0,15	0,15
Expansión 3	CE-IDF	0,94	**0,92**	**0,92**	**0,82**	**0,62**	0,47	0,31	0,24	0,17	0,17
	VT-IDF	0,95	**0,92**	0,90	0,76	0,61	**0,52**	**0,36**	**0,26**	**0,18**	**0,18**
	Rocchio	**0,98**	0,91	0,82	0,68	0,51	0,37	0,22	0,17	0,15	0,15
Expansión 4	CE-IDF	0,94	**0,92**	**0,92**	**0,83**	**0,67**	0,49	0,35	0,26	0,17	0,17
	VT-IDF	0,96	**0,92**	0,90	0,81	0,63	**0,52**	**0,37**	**0,28**	**0,20**	**0,20**
	Rocchio	**0,98**	0,91	0,83	0,67	0,51	0,37	0,22	0,17	0,15	0,15

En general los estos resultados no son muy diferentes a los obtenidos en el experimento anterior.

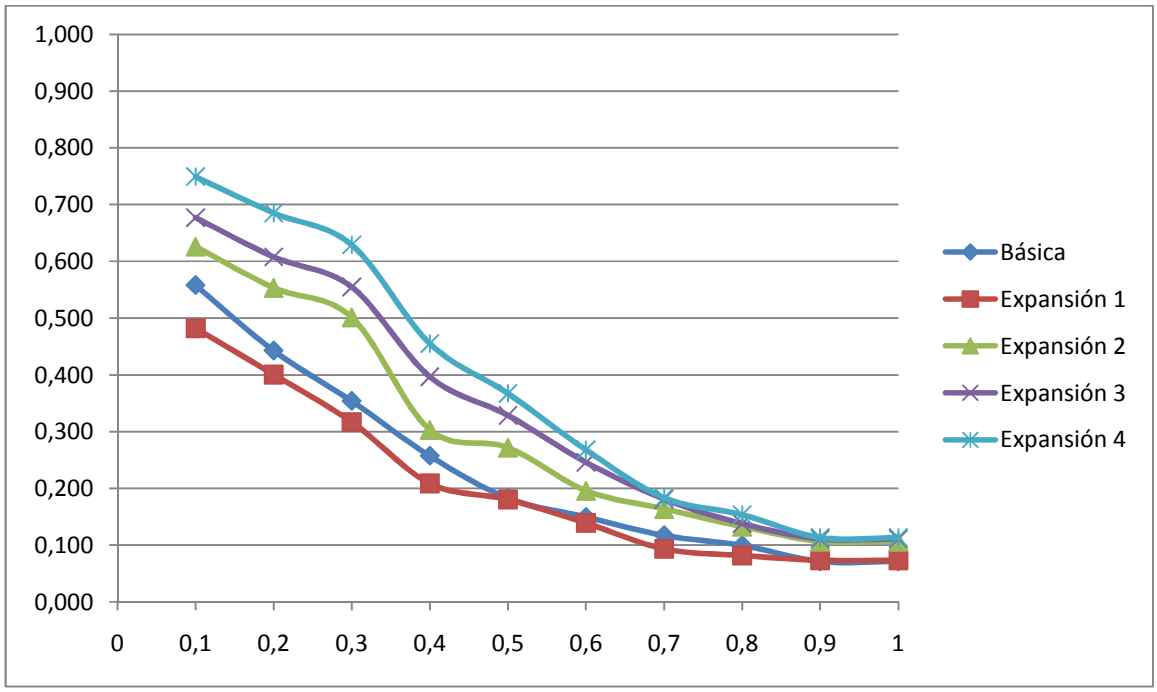
Finalmente, se realizó un tercer experimento sobre la misma colección de datos. El proceso seguido fue el mismo del experimento anterior, pero en este caso el perfil del usuario se mantuvo durante todas las consultas. Este proceso simula el almacenamiento del perfil de un usuario durante toda su vida en el sistema. Se considera el experimento más importante, debido a que en general, los sistemas de recuperación de información o búsqueda web deben mantener un perfil del usuario durante todo el tiempo que el usuario use el sistema y que este perfil se adapte a las cambiantes necesidades de búsqueda de los usuarios. Los resultados de este experimento se presentan en la Figura 11.



(a) Rocchio



(b) VT-IDF



(c) CE-IDF

Figura 11. Curva de precisión-recuerdo para Rocchio (a), VT-IDF (b) y CE-IDF (c) sobre CACM con memoria de largo plazo

En las tres gráficas de la Figura 11 (a, b y c), se muestra el resultado de la consulta básica usando Lucene (serie de datos con marcador en forma de rombo), luego en las líneas con marcador rectangular se muestra el resultado de la expansión 1. En este caso los tres algoritmos obtienen precisiones más bajas que las logradas con la expansión básica, esto debido al peso del perfil del usuario (historia de las consultas pasadas) sobre la consulta que se está realizando. Pero en este caso el algoritmo CE-IDF obtiene un mayor valor de precisión, mostrando que este algoritmo es menos sensible a la historia del usuario o dicho de otro modo, que CE-IDF se adapta más rápidamente a los cambios en los requerimientos de las consultas del usuario.

Tabla 4. Valores de precisión -recuerdo para Rocchio, VT-IDF y CE-IDF sobre CACM con memoria de largo plazo

	Recuerdo	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
Básica	Lucene	0,56	0,44	0,35	0,26	0,18	0,15	0,12	0,10	0,07	0,07
Expansión 1	CE-IDF	**0,48**	**0,40**	**0,32**	**0,21**	**0,18**	**0,14**	**0,09**	**0,08**	**0,07**	**0,07**
	VT-IDF	0,32	0,23	0,16	0,11	0,08	0,07	0,05	0,02	0,02	0,02
	Rocchio	0,16	0,09	0,07	0,07	0,07	0,06	0,04	0,04	0,03	0,03
Expansión 2	CE-IDF	**0,63**	**0,55**	**0,50**	**0,30**	**0,27**	**0,20**	**0,16**	**0,13**	**0,11**	**0,11**
	VT-IDF	0,53	0,45	0,33	0,24	0,21	0,17	0,14	0,11	0,09	0,09
	Rocchio	0,20	0,12	0,10	0,08	0,07	0,06	0,04	0,04	0,03	0,03
Expansión 3	CE-IDF	**0,68**	**0,61**	**0,55**	**0,40**	**0,33**	**0,25**	0,18	0,14	**0,11**	**0,11**
	VT-IDF	0,65	0,56	0,46	0,32	0,27	0,22	**0,20**	**0,15**	0,10	0,10
	Rocchio	0,24	0,14	0,13	0,08	0,08	0,07	0,04	0,04	0,04	0,04
Expansión 4	CE-IDF	**0,75**	**0,69**	**0,63**	**0,46**	**0,37**	**0,27**	0,18	0,15	0,11	**0,11**
	VT-IDF	**0,75**	0,67	0,56	0,38	0,33	0,26	**0,22**	**0,17**	**0,12**	**0,11**
	Rocchio	0,27	0,16	0,13	0,10	0,09	0,08	0,05	0,05	0,04	0,04

En la expansión 2 (serie de datos con marcador triangular), se muestra como los tres algoritmos mejoran la precisión, pero sólo CE-IDF mejora la consulta básica. Para la expansión 3 y 4 todos los algoritmos mejoran sus resultados progresivamente, pero sólo CE-IDF y VT-IDF obtienen mejores resultados a la consulta básica, llegando a una diferencia hasta de 20% en el primer nivel de recuerdo. En todos los casos CE-IDF obtiene mejores resultados, reafirmando con esto, la idea de que es un método que se adapta más rápidamente a las nuevas necesidades del usuario.

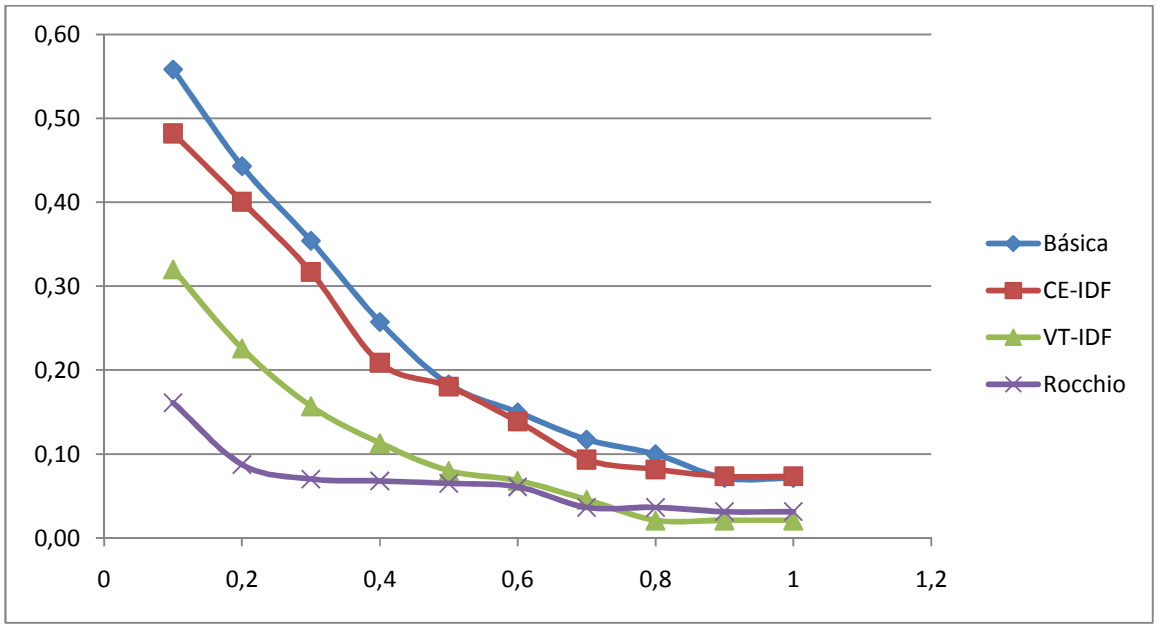
En la gráfica de Rocchio de la Figura 11, además se observa que el proceso de mejora es más lento que el obtenido con los otros dos algoritmos. Evaluaciones adicionales, mostraron que Rocchio puede obtener mejores resultados de

precisión en este tercer experimento cuando α = 90%, β = 10% y γ = 0%. En este caso la precisión oscila entre 55% y 62% en el primer nivel de recuerdo durante las cuatro expansiones. Desafortunadamente, con estos parámetros los valores de precisión para los dos primeros experimentos disminuyen a 91% y 94% en el primer nivel de recuerdo en las cuatros expansiones. Con estos nuevos valores para los parámetros se logra disminuir el peso del historial sobre la consulta inicial del usuario en el algoritmo de Rocchio. Además con esto se evidencia la dificultad que puede presentar la definición apropiada de estos valores en este algoritmo.

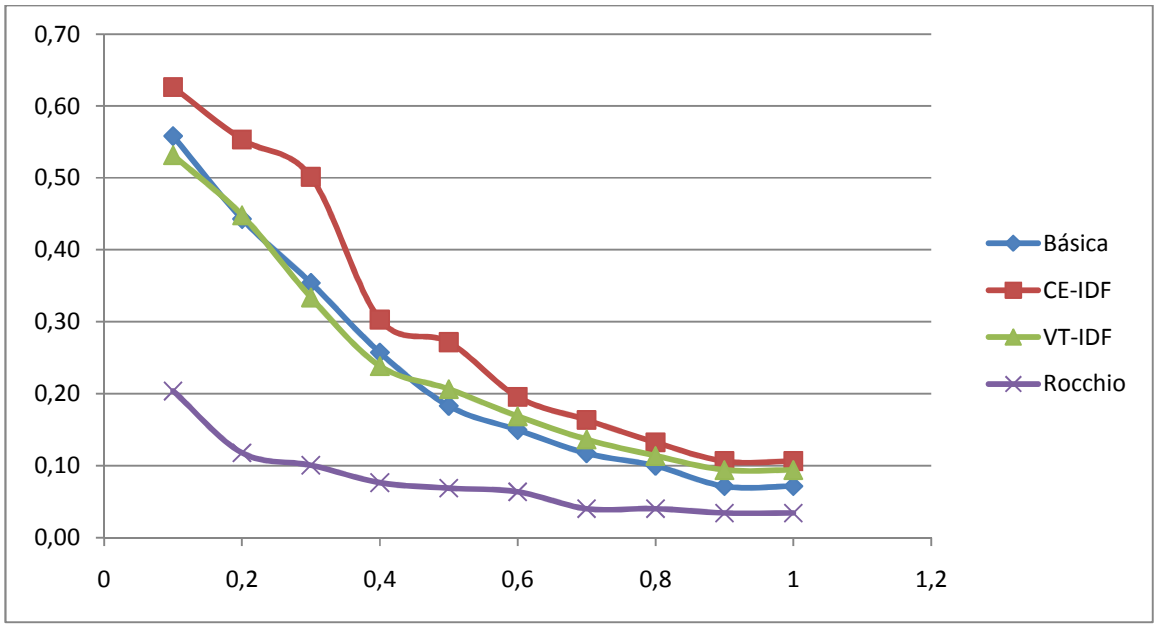
En la Tabla 4 se muestran en detalle los valores de la Figura 11. Se muestra como CE-IDF logra desde la expansión 1 una precisión de 48% en el 10% de recuerdo y como en la expansión 4 alcanza un 75%. Mientras que Rocchio logra tan sólo un 16% en la primer expansión y un máximo de 27% en la expansión 4. Finalmente, muestra que VT-IDF a pesar de empezar con un 32% en la primera expansión, alcanza a igualar a CE-IDF en la expansión 4 con un 75% de precisión

En la Figura 11 se muestra la curva de precisión-recuerdo de tres expansiones y permite comparar visualmente los resultados obtenidos con los tres algoritmos. En general los resultados muestran que CE-IDF es un mejor algoritmo cuando se tiene en cuenta un perfil de largo plazo, seguido de VT-IDF y por último de Rocchio.

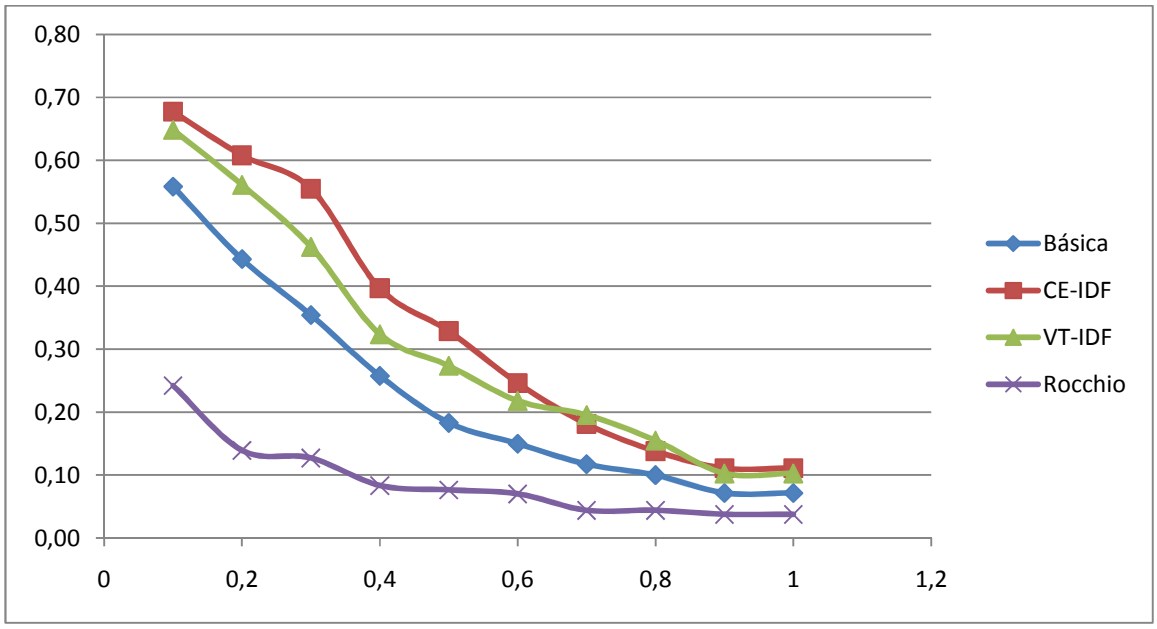
Por último vamos a comparar cada algoritmo expansión por expansión para apreciar la forma en que evolucionan una respecto a la otra cuando la memoria es a largo plazo, esto lo podemos a preciar en la Figura 12.



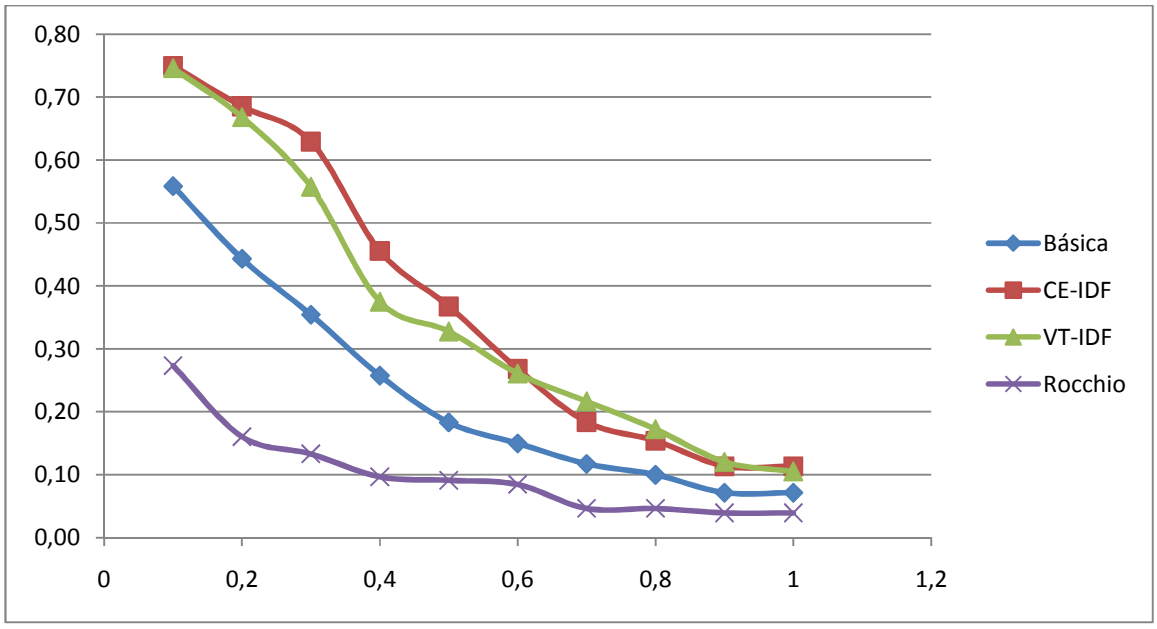
(a) Expansión 1



(b) Expansión 2



(c) Expansión 3



(d) Expansión 4

Figura 12. Comparación de Rocchio, VT-IDF y CE-IDF en cuatro expansiones con CACM

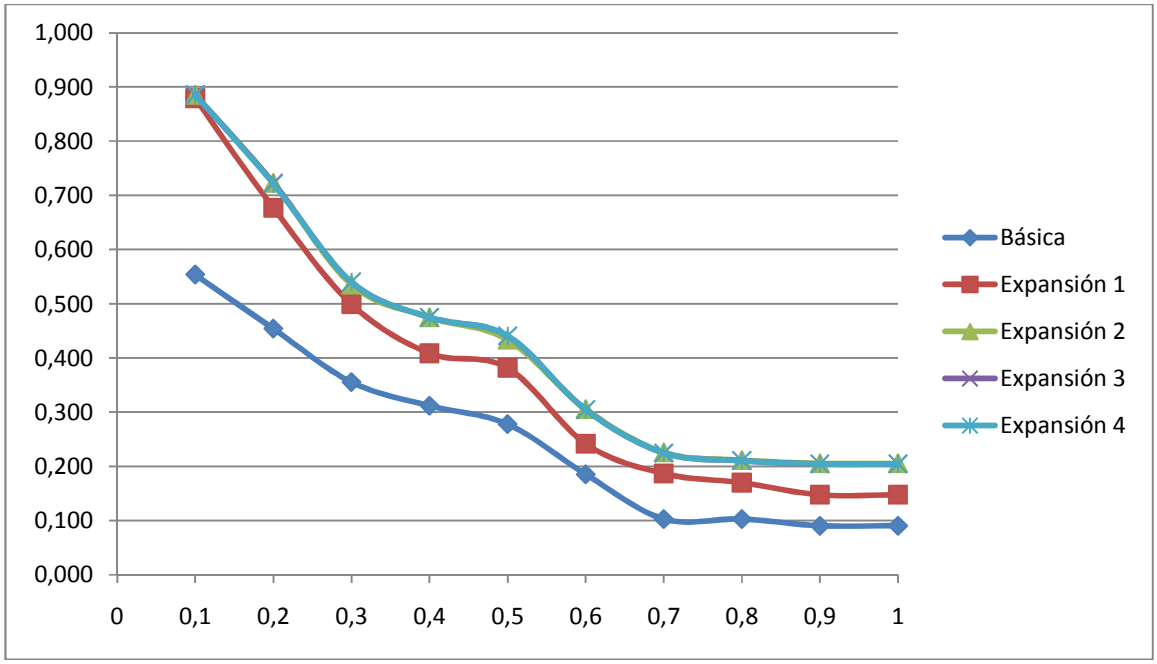
3.6 EXPERIMENTO LISA

Se realizó un segundo experimento con una colección de textos diferente denominada Library & Information Science Abstracts (LISA), Disponible gratuitamente 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). En la colección se encuentran 6004 documentos y 35 consultas. Para cada consulta, asesores humanos leyeron todos los documentos y evaluaron cuáles de ellos son relevantes. En la presente investigación se tomaron las 35 consultas.

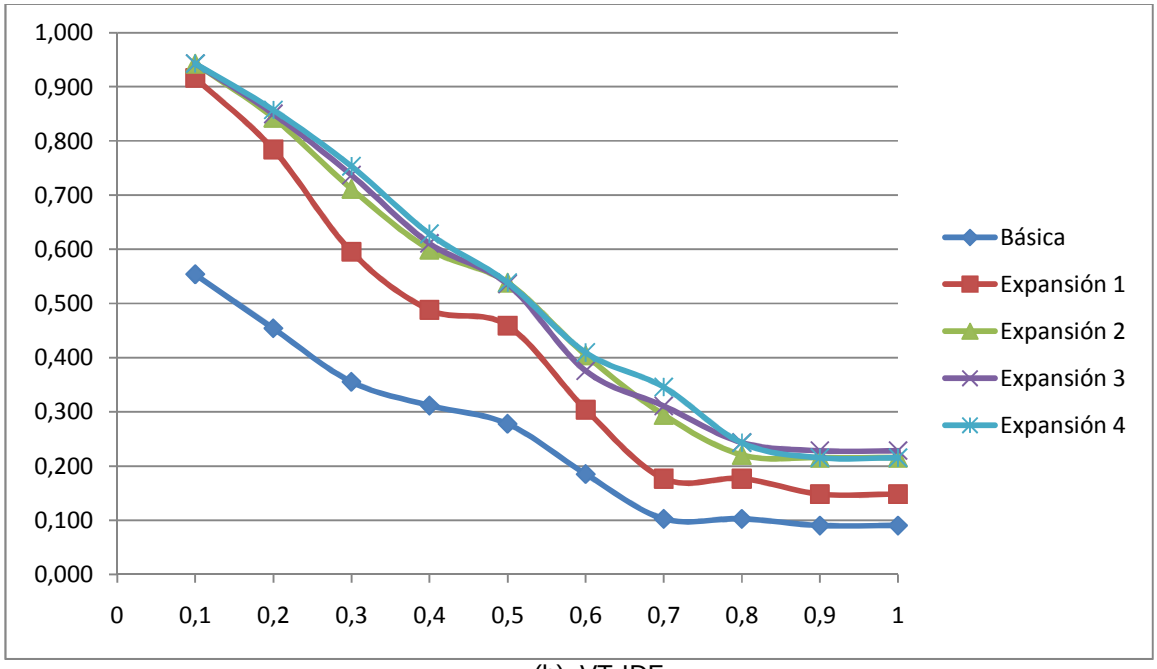
En las tres gráficas de la Figura 13 (a, b y c) se muestra el resultado de la consulta básica usando Lucene, que inicia en un 55% de precisión para un nivel de recuerdo de 10%, y decrece hasta un 9% cuando el nivel de recuerdo es del 100%. Luego en las líneas con marcador rectangular se muestra el resultado de la expansión 1, mostrando una mejora apreciable en los tres algoritmos, llegando a un promedio de 91,3% de precisión en el primer nivel de recuerdo y cayendo a un promedio de 20,3% en el último nivel de recuerdo. Este primer proceso de expansión, muestra una curva de precisión-recuerdo que es muy superior en todos los niveles de recuerdo en la consulta básica. Además muestra como los tres algoritmos siguen mejorando poco a poco en la expansión 2, 3 y 4.

En la Tabla 5 se muestran en detalle los valores de la Figura 13. Se muestra como VT-IDF logra desde la expansión 1 una precisión de 92% en el 10% de recuerdo y como en la expansión 4 alcanza un 94%. Mientras que Rocchio logra un 88% en la primer expansión y un máximo de 89% en la expansión 4. Finalmente, muestra que CE-IDF logra un valor inicial y final de 91% en las 4 expansiones, pero

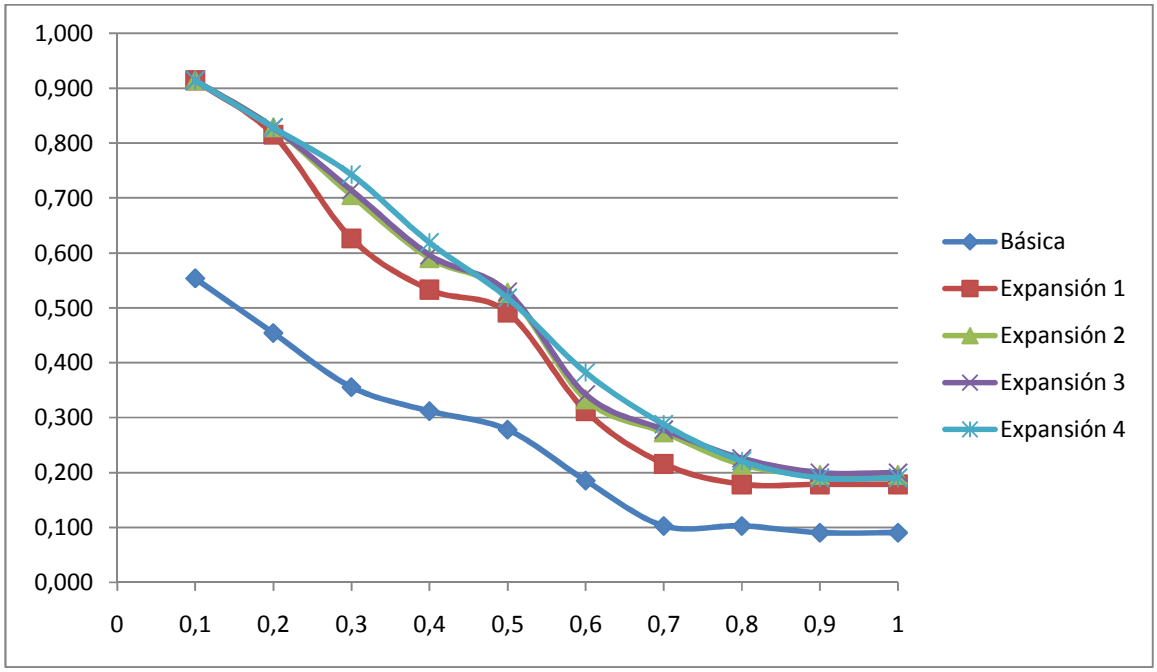
logrando mejoras en los niveles de recuerdo 20% al 100% desde la primera expansión.



(a) Rocchio



(b) VT-IDF



(c) EC-IDF

Figura 13. Curva de precisión-recuerdo para Rocchio (a), VT-IDF (b) y CE-IDF (c) sobre LISA sin memoria del perfil

Tabla 5. Valores de precisión -recuerdo para Rocchio, VT-IDF y CE-IDF sobre LISA sin memoria del perfil

	Recuerdo	**10%**	**20%**	**30%**	**40%**	**50%**	**60%**	**70%**	**80%**	**90%**	**100%**
Básica	Lucene	0,55	0,45	0,36	0,31	0,28	0,19	0,10	0,10	0,09	0,09
Expansión 1	CE-IDF	0,91	**0,82**	**0,63**	**0,53**	**0,49**	**0,31**	**0,22**	**0,18**	**0,18**	**0,18**
	VT-IDF	**0,92**	0,78	0,60	0,49	0,46	0,30	0,18	**0,18**	0,15	0,15
	Rocchio	0,88	0,68	0,50	0,41	0,38	0,24	0,19	0,17	0,15	0,15
Expansión 2	CE-IDF	0,91	0,83	0,70	0,59	0,53	0,33	0,27	0,21	0,20	0,20
	VT-IDF	**0,94**	**0,84**	**0,71**	**0,60**	**0,54**	**0,40**	**0,29**	**0,22**	**0,22**	**0,22**
	Rocchio	0,89	0,72	0,54	0,47	0,43	0,31	0,23	0,21	0,21	0,21

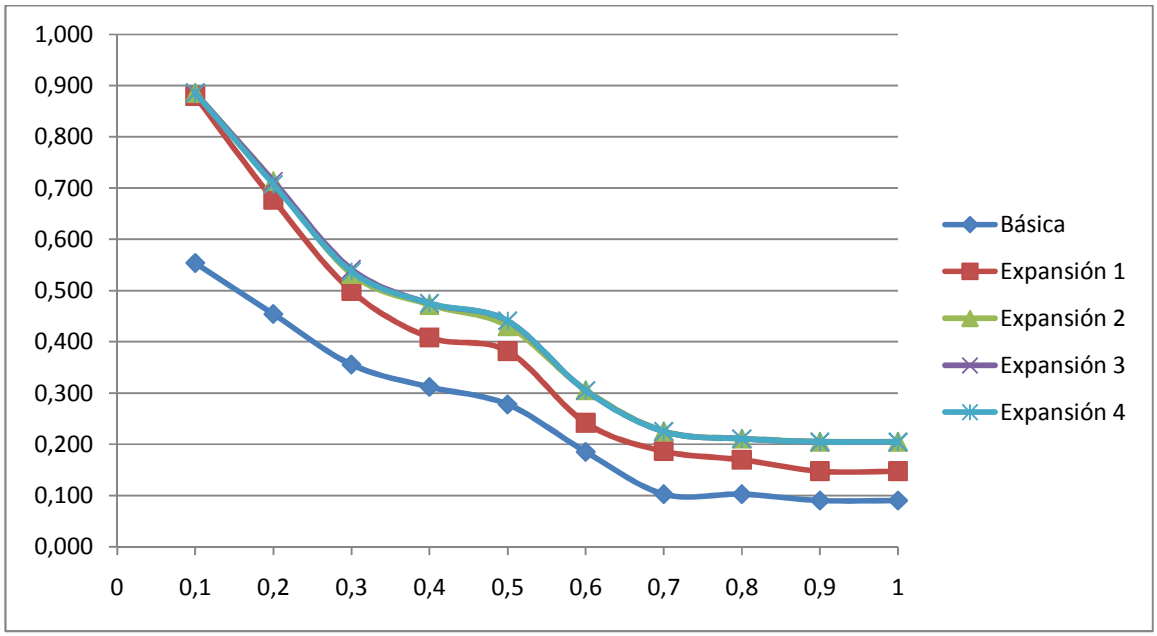
Expansión 3	CE-IDF	0,91	0,83	0,71	0,60	0,53	0,34	0,28	0,23	0,20	0,20
	VT-IDF	**0,94**	**0,85**	**0,74**	**0,61**	**0,54**	**0,38**	**0,31**	**0,24**	**0,23**	**0,23**
	Rocchio	0,89	0,72	0,54	0,47	0,44	0,30	0,22	0,21	0,20	0,20
Expansión 4	CE-IDF	0,91	0,83	0,74	0,62	0,52	0,38	0,29	0,22	0,19	0,19
	VT-IDF	**0,94**	**0,86**	**0,75**	**0,63**	**0,54**	**0,41**	**0,35**	**0,24**	**0,22**	**0,22**
	Rocchio	0,89	0,72	0,54	0,47	0,44	0,30	0,22	0,21	0,20	0,20

En este experimento muestra que para la colección de datos seleccionada el algoritmo VT-IDF obtiene mejores resultados en todos los niveles de recuerdo para las expansiones 2, 3 y 4, seguido por CE-IDF, pero CE-IDF en general obtiene los mejores resultados para la expansión 1, seguido por VT-IDF, dejando a Rocchio en último lugar.

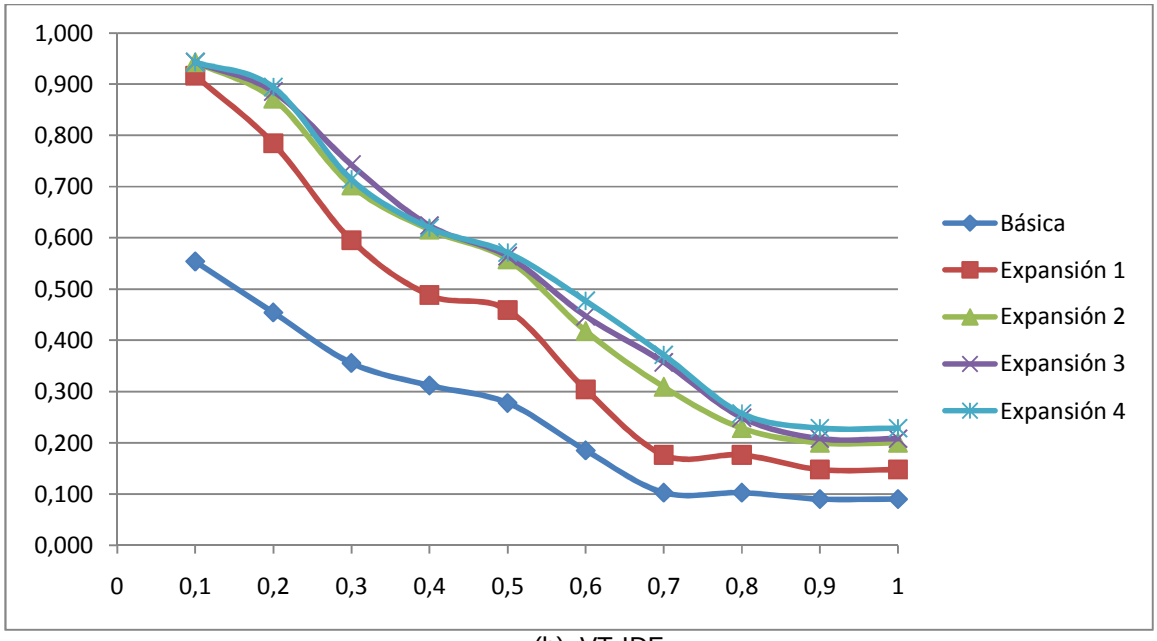
Un segundo experimento se llevó a cabo sobre LISA. El proceso seguido fue el mismo del experimento anterior, pero en este caso el perfil del usuario mantuvo memoria en las cinco ejecuciones de la misma consulta. Este proceso simula el almacenamiento del perfil de un usuario durante una sesión de consulta de un tema. Los resultados de este experimento se presentan en la Figura 14.

En las tres gráficas de la Figura 14 (a, b y c), se muestra el resultado de la consulta básica usando Lucene, luego en las líneas con marcador rectangular se muestra el resultado de la expansión 1, mostrando una mejora apreciable en los tres algoritmos, llegando a un promedio de 90% de precisión en el primer nivel de recuerdo y cayendo a un promedio de 16% en el último nivel de recuerdo. Este primer proceso de expansión muestra una curva de precisión-recuerdo que es evidentemente muy superior en todos los niveles de recuerdo a la consulta básica. Además se muestra como Rocchio, VT-IDF y CE-IDF aprovechan la mayor

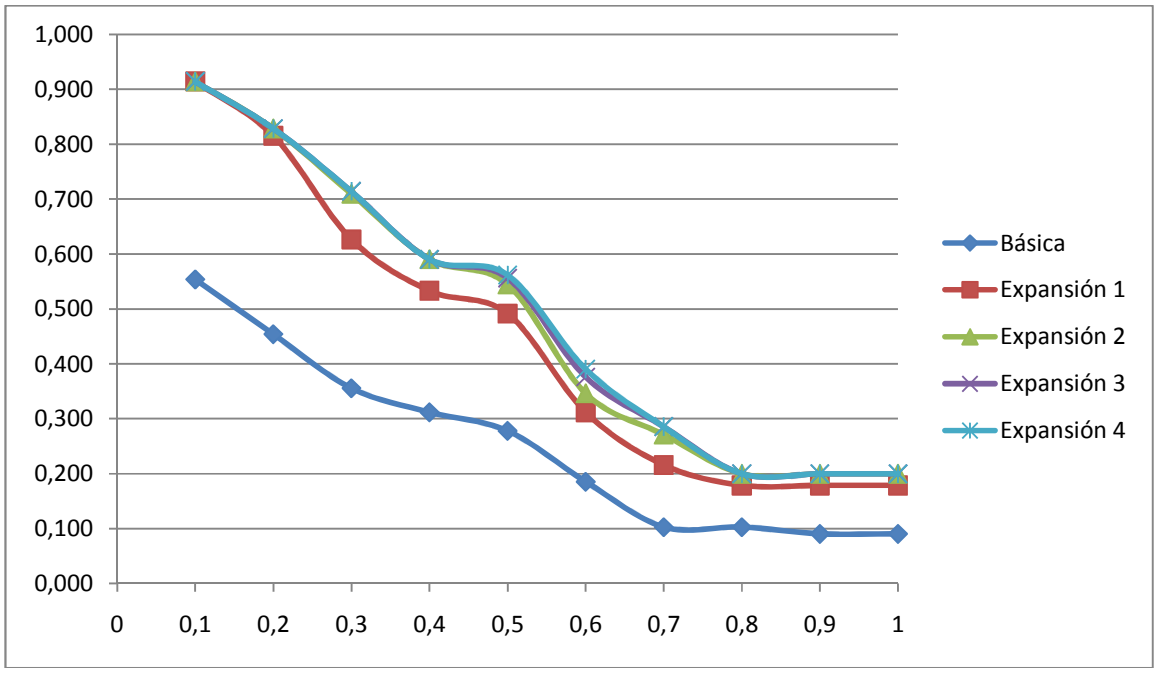
información del perfil para mejorar la precisión de los resultados, expansión tras expansión, en los diferentes niveles de recuerdo.



(a) Rocchio



(b) VT-IDF



(c) CE-IDF

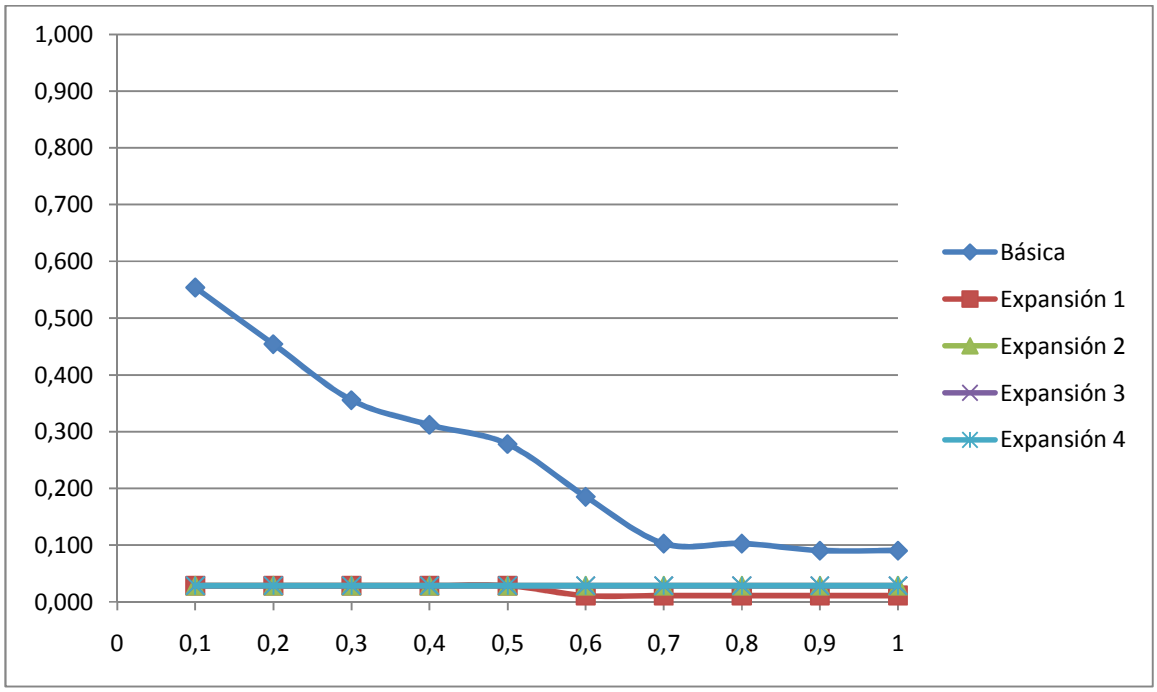
Figura 14. Curva de precisión-recuerdo para Rocchio (a), VT-IDF (b) y CE-IDF (c) sobre LISA con memoria de sesión

En la Tabla 6 se muestran en detalle los valores de la Figura 14. Se muestra como VT-IDF logra desde la expansión 1 una precisión de 92% en el 10% de recuerdo y como en la expansión 4 alcanza un 94%. Mientras que Rocchio logra un 88% en la primer expansión y un máximo de 89% en la expansión 4. Finalmente, muestra que CE-IDF logra un 91% en el primer nivel de recuerdo en todas las expansiones. De igual forma que en el experimento anterior en general obtiene los mejores resultados para la expansión 1, pero se mantiene el comportamiento de los 3 algoritmos dejando de nuevo a Rocchio en último lugar.

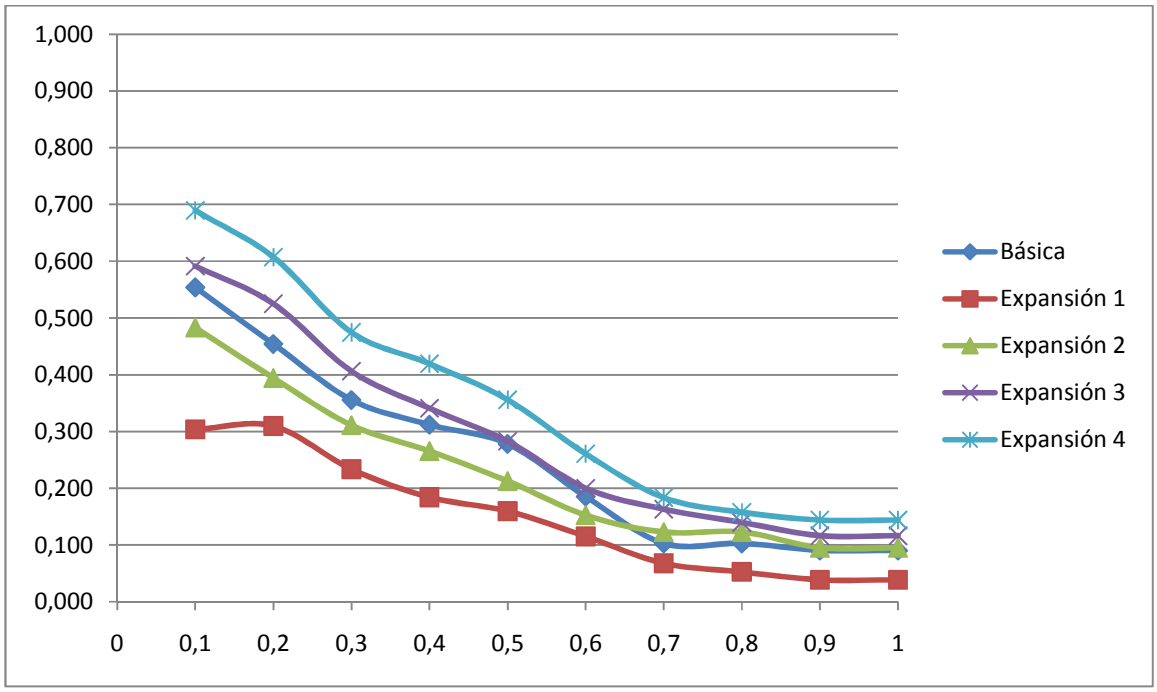
Tabla 6. Valores de precisión -recuerdo para Rocchio, VT-IDF y CE-IDF sobre LISA con memoria de sesión

	Recuerdo	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
Básica	Lucene	0,55	0,45	0,36	0,31	0,28	0,19	0,10	0,10	0,09	0,09
Expansión 1	CE-IDF	0,91	**0,82**	**0,63**	**0,53**	**0,49**	**0,31**	**0,22**	**0,18**	**0,18**	**0,18**
	VT-IDF	**0,92**	0,78	0,60	0,49	0,46	0,30	0,18	**0,18**	0,15	0,15
	Rocchio	0,88	0,68	0,50	0,41	0,38	0,24	0,19	0,17	0,15	0,15
Expansión 2	CE-IDF	0,91	0,83	**0,71**	0,59	0,54	0,35	0,27	0,20	0,20	0,20
	VT-IDF	**0,94**	**0,87**	0,70	**0,62**	**0,56**	**0,42**	**0,31**	**0,23**	0,20	0,20
	Rocchio	0,89	0,71	0,53	0,47	0,43	0,31	0,23	0,21	**0,21**	**0,21**
Expansión 3	CE-IDF	0,91	0,83	0,71	0,59	**0,56**	0,38	0,29	0,20	0,20	0,20
	VT-IDF	**0,94**	**0,89**	**0,74**	**0,62**	**0,56**	**0,45**	**0,36**	**0,25**	**0,21**	**0,21**
	Rocchio	0,89	0,71	0,54	0,47	0,44	0,30	0,22	0,21	0,20	0,20
Expansión 4	CE-IDF	0,91	0,83	**0,71**	0,59	0,56	0,39	0,29	0,20	0,20	0,20
	VT-IDF	**0,94**	**0,89**	**0,71**	**0,62**	**0,57**	**0,48**	**0,37**	**0,26**	**0,23**	**0,23**
	Rocchio	0,89	0,71	0,54	0,47	0,44	0,30	0,22	0,21	0,20	0,20

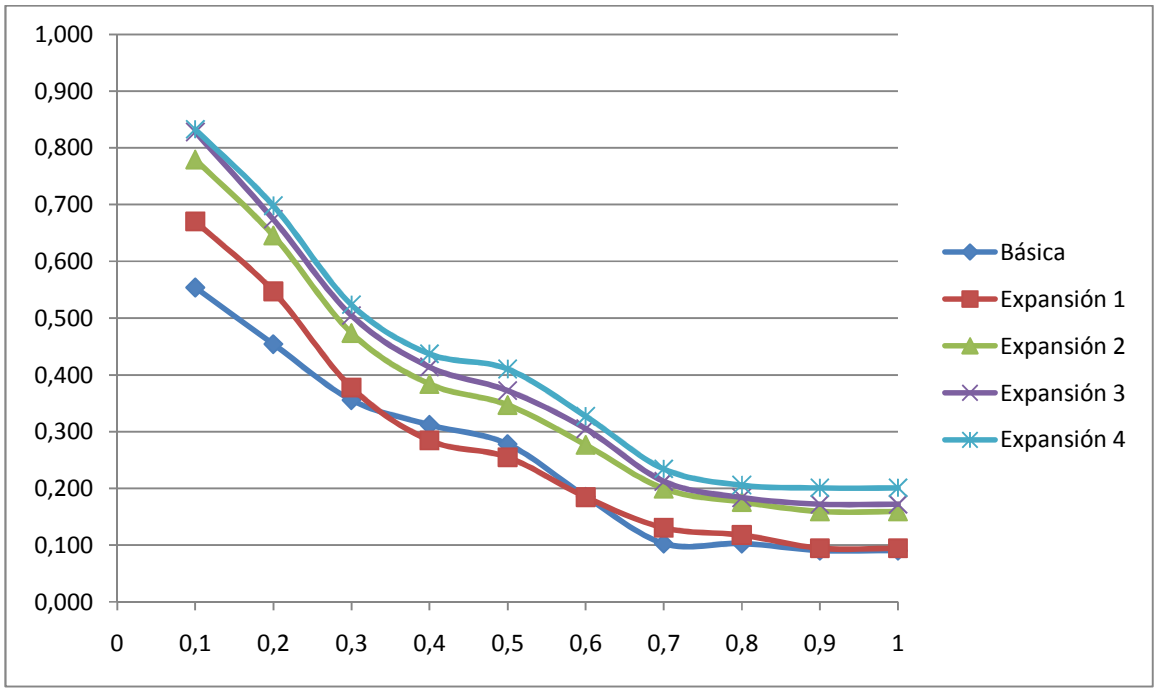
Por último y al igual que en el experimento con CACM, se realizó un tercer experimento sobre el mismo data set. El proceso seguido fue el mismo del experimento anterior, pero en este caso el perfil del usuario se mantuvo durante todas las consultas. Este proceso simula el almacenamiento del perfil de un usuario durante toda su vida en el sistema. Se considera el experimento más importante, debido a que en general, los sistemas de recuperación de información o búsqueda web deben mantener un perfil del usuario durante todo el tiempo que el usuario use el sistema y que este perfil se adapte a las cambiantes necesidades de búsqueda de los usuarios. Los resultados de este experimento se presentan en la Figura 15 (a, b y c).



(a) Rocchio



(b) VT-IDF



(c) CE-IDF

Figura 15. Curva de precisión-recuerdo para Rocchio (a), VT-IDF (b) y CE-IDF (c) sobre LISA con memoria de largo plazo

En las tres gráficas de la Figura 15 (a, b y c), se muestra el resultado de la consulta básica usando Lucene (serie de datos con marcador en forma de rombo), luego en las líneas con marcador rectangular se muestra el resultado de la expansión 1. En este caso VT-IDF y Rocchio obtienen precisiones más bajas que las logradas con la expansión básica y en especial Rocchio cuyo valor de precisión es demasiado bajo aun con respecto a VT-IDF, esto debido al peso del perfil del usuario (historia de las consultas pasadas) sobre la consulta que se está realizando. Pero en este caso el algoritmo CE-IDF obtiene un mayor valor de precisión, mostrando que este algoritmo es menos sensible a la historia del usuario o dicho de otro modo, que CE-IDF se adapta más rápidamente a los cambios en los requerimientos de las consultas del usuario.

En la expansión 1 (serie de datos con marcador rectangular de color rojo), se muestra como CE-IDF mejora la consulta básica. Para la expansión 3 VT-IDF alcanza un valor por encima de la básica y para la expansión 4 Rocchio sigue lejos de la básica ya que su mejora con respecto a las expansiones anteriores es muy poca obteniendo en general valores demasiado bajos, sigue sin superar el 3% de precisión en el nivel de recuerdo del 10%. En todos los casos CE-IDF obtiene mejores resultados, reafirmando con esto, la idea de que es un método que se adapta más rápidamente a las nuevas necesidades del usuario.

En la gráfica de Rocchio de la Figura 15, además se observa que el proceso de mejora es muchísimo más lento que el obtenido con los otros dos algoritmos. Evaluaciones adicionales, mostraron que Rocchio puede obtener mejores resultados de precisión en este tercer experimento cuando $\alpha = 90\%$, $\beta = 10\%$ y $\gamma = 0\%$. En este caso la precisión oscila entre 4,3% y 5,7% en el primer nivel de recuerdo durante las cuatro expansiones. Desafortunadamente, con estos parámetros los valores de precisión para los dos primeros experimentos disminuyen a 71% en el primer nivel de recuerdo en las cuatro expansiones. Con estos nuevos valores para los parámetros se logra disminuir el peso del historial sobre la consulta inicial del usuario en el algoritmo de Rocchio. Además con esto se confirma la evidente dificultad que puede presentar la definición apropiada de estos valores en este algoritmo.

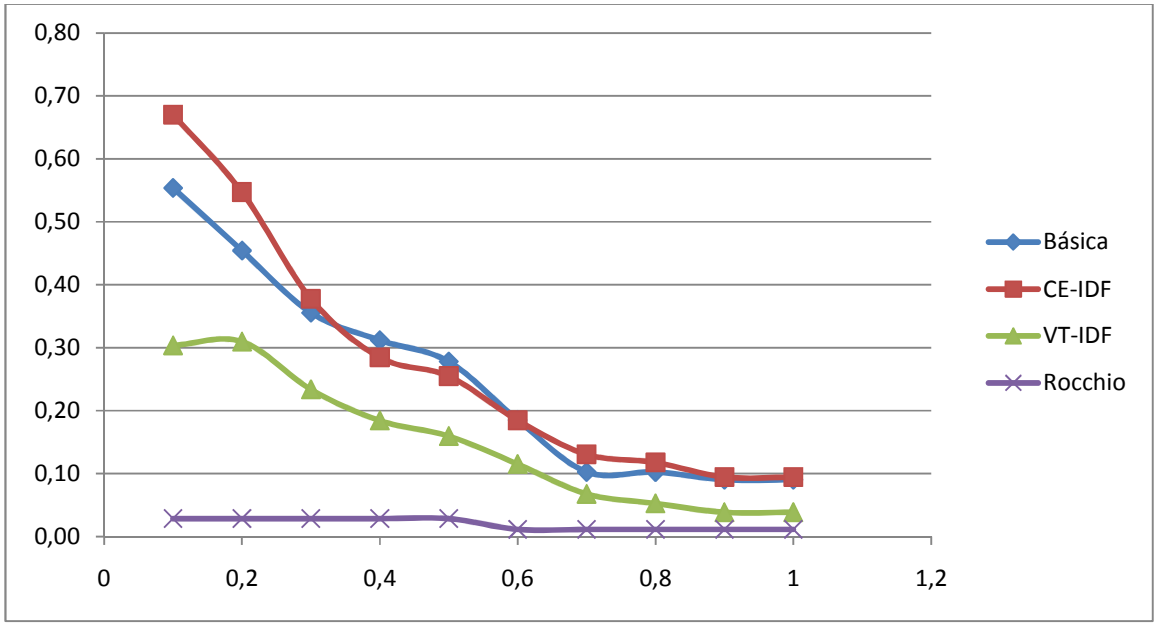
En la Tabla 7 se muestran en detalle los valores de la Figura 15. Se muestra como CE-IDF logra desde la expansión 1 una precisión de 67% en el 10% de recuerdo y como en la expansión 4 alcanza un 83%. Mientras que Rocchio logra tan sólo un 3% en todas las expansiones. Finalmente, muestra que VT-IDF a pesar de empezar con un 30% en la primera expansión, alcanza un 69% de precisión en la expansión 4.

Tabla 7. Valores de precisión -recuerdo para Rocchio, VT-IDF y CE-IDF sobre LISA con memoria de largo plazo

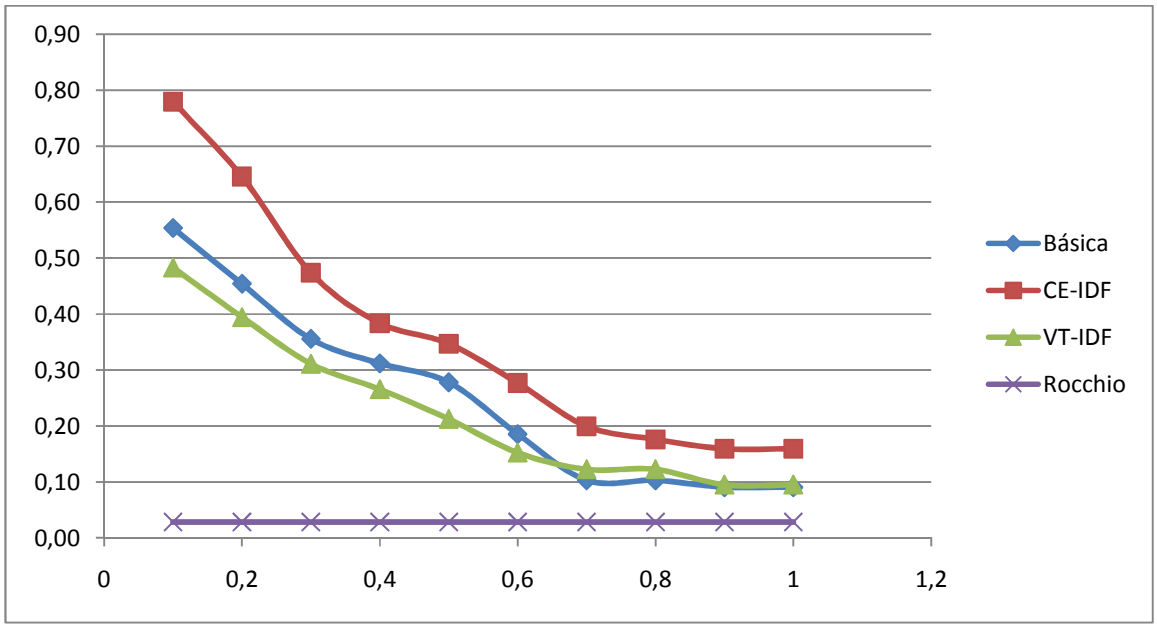
	Recuerdo	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
Básica	Lucene	0,55	0,45	0,36	0,31	0,28	0,19	0,10	0,10	0,09	0,09
Expansión 1	CE-IDF	**0,67**	**0,55**	**0,38**	**0,28**	**0,26**	**0,18**	**0,13**	**0,12**	**0,09**	**0,09**
	VT-IDF	0,30	0,31	0,23	0,18	0,16	0,12	0,07	0,05	0,04	0,04
	Rocchio	0,03	0,03	0,03	0,03	0,03	0,01	0,01	0,01	0,01	0,01
Expansión 2	CE-IDF	**0,78**	**0,65**	**0,47**	**0,38**	**0,35**	**0,28**	**0,20**	**0,18**	**0,16**	**0,16**
	VT-IDF	0,48	0,39	0,31	0,27	0,21	0,15	0,12	0,12	0,10	0,10
	Rocchio	0,03	0,03	0,03	0,03	0,03	0,03	0,03	0,03	0,03	0,03
Expansión 3	CE-IDF	**0,83**	**0,67**	**0,50**	**0,41**	**0,37**	**0,31**	**0,21**	**0,18**	**0,17**	**0,17**
	VT-IDF	0,59	0,53	0,41	0,34	0,28	0,20	0,16	0,14	0,12	0,12
	Rocchio	0,03	0,03	0,03	0,03	0,03	0,03	0,03	0,03	0,03	0,03
Expansión 4	CE-IDF	**0,83**	**0,70**	**0,52**	**0,44**	**0,41**	**0,33**	**0,23**	**0,21**	**0,20**	**0,20**
	VT-IDF	0,69	0,61	0,47	0,42	0,36	0,26	0,18	0,16	0,14	0,14
	Rocchio	0,03	0,03	0,03	0,03	0,03	0,03	0,03	0,03	0,03	0,03

En la Figura 15 se muestra la curva de precisión-recuerdo de las tres expansiones y permite comparar visualmente los resultados obtenidos con los tres algoritmos. CE-IDF demuestra su superioridad cuando se tiene en cuenta un perfil de largo plazo, seguido de VT-IDF y por último de Rocchio con valores muy por debajo aun de la consulta básica en todas las expansiones realizadas.

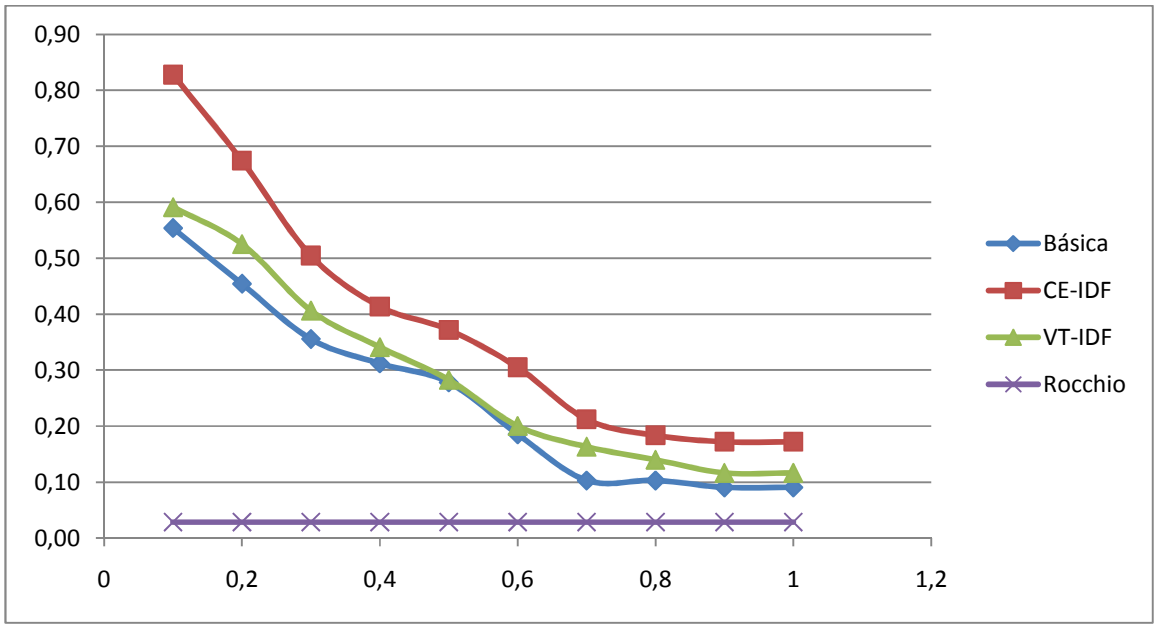
Por último al igual que se realizó con CACM se comparó cada algoritmo, expansión por expansión para apreciar la forma en que evolucionan una respecto a la otra cuando la memoria es a largo plazo, esto se puede apreciar en la Figura 16.



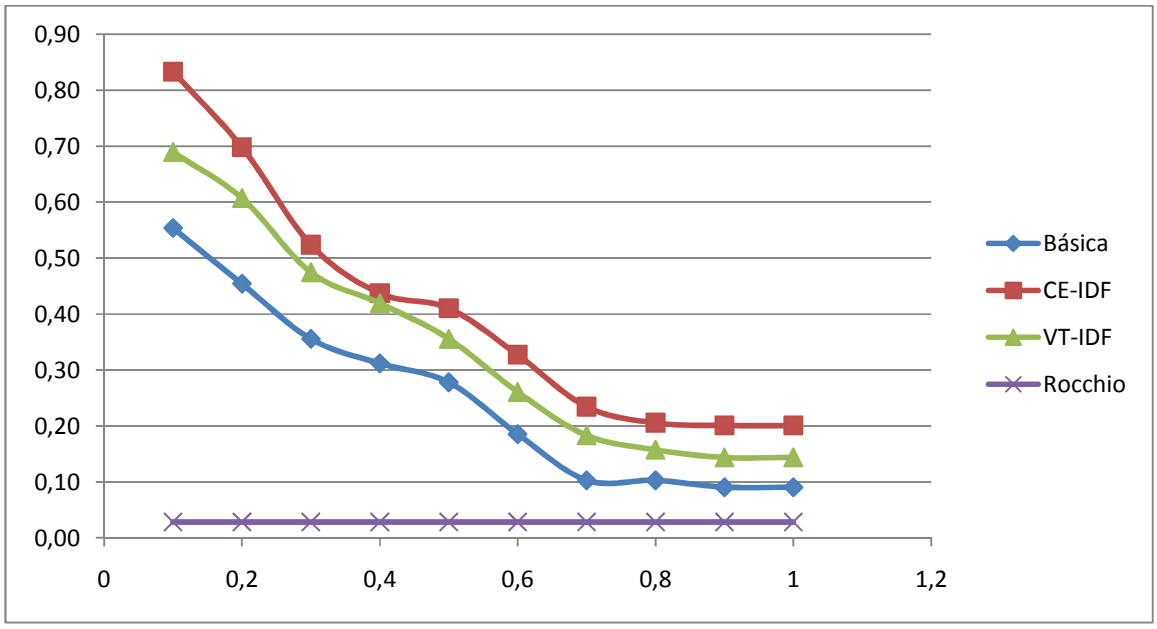
(a) Expansión 1



(b) Expansión 2



(c) Expansión 3



(d) Expansión 4

Figura 16. Comparación de Rocchio, VT-IDF y CE-IDF en cuatro expansiones con LISA

4. DESCRIPCIÓN DEL META BUSCADOR WEB

Para dar cumplimiento al cuarto objetivo específico, en este capítulo se describe el meta buscador web desarrollado con la propuesta completa de expansión de consulta, el cual fue denominado ECWEB, y fue usado como herramienta de evaluación de los algoritmos con propuestos.

4.1 INTERFAZ DE ECWEB

La interfaz de búsqueda en ECWEB es sencilla e intuitiva. En la Figura 17 se puede apreciar la caja de texto donde se digita la consulta y el botón con el que se realiza la búsqueda, éste último inhabilitado ya que en ese momento no hay un usuario autenticado en el sistema. Más hacia la derecha se encuentran las opciones básicas, que son:

1. Home: Ir a la página inicial del sistema (Figura 17).
2. Help: Muestra brevemente el funcionamiento de CEWEB.
3. Register: Donde el usuario crea su cuenta para poder acceder al meta buscador.
4. Login: página donde se realiza la autenticación del usuario, previamente registrado.

Figura 17. Página principal de ECWEB

A continuación se presenta un breve recorrido por las opciones de ECWEB.

4.2 AYUDA

La ayuda para los usuarios de ECWEB está compuesta por 5 partes que explican el funcionamiento de la aplicación (ver Figura 18), y son:

1. Qué es ECWEB
2. Como realizar las búsquedas
3. Estructura de un documento recuperado
4. Marcar un documento como relevante
5. Marcar un documento como no relevante

Help Home Register Login

HOW WORK ECWEB

WHAT IS ECWEB

It is a meta-search web, simple and intuitive graphical interface with a friendly and flexible to the needs of the user performing the search using Google, Yahoo and Bing and permits through an easy user interaction with the system, take advantage of personally the benefits provided by the EC-IDF algorithm as a method of query expansion to improve the relevance of retrieved documents. For this iteration of the user system is necessary for the user is registered .

HOW TO SEARCH

Enter the query in the text box and press the Search button, see Figure 1.

Figure 1.

STRUCTURE OF A DOCUMENT RETRIEVED

The number that appears on the left indicates the order in which they are presented to the user, the title is a little more to the right as a hiperlynk, which opens the document in the same window, a little to the right is cleared and a cross, used to mark the document as relevant and irrelevant as, respectively, and with them a magnifying glass with which to open the document in another window. It also shows a brief summary of the document, its URL, the source of search that retrieves the document and a value that represents the importance in relation to the consultation. See Figure 1.

Figure 2.

MARK A DOCUMENT AS RELEVANT

When a retrieved document is important for the query typed, need to mark it as relevant by the approval (☑) that shown in Figure 2, which changes to *GREEN*

Figure 3.

MARK A DOCUMENT AS NOT RELEVANT

When a retrieved document is not important for the query typed, you must mark it as not relevant, by the approval (☒) that shown in Figure 3, which changes to *RED*

Figure 4.

Figura 18 Ayuda - ECWEB

4.3 REGISTRO

Este formulario se puede apreciar en la Figura 19.

Los siguientes datos son necesarios para crear una cuenta:

1. User name: Puede ser cualquiera, no tiene restricciones.
2. Password: Tiene una única condición y es que al menos un carácter debe ser no alfanumérico.
3. Confirm password: Esto se hace como regla mínima de seguridad, para verificar que el usuario haya digitado una contraseña que recuerde y que este bien escrita.
4. E-mail: Campo necesario para la recuperación de la contraseña en caso de pérdida.
5. Security question y security answer: Campos necesarios para poder realizar todo el proceso de recuperación de contraseña, la pegunta puede ser cualquiera y la respuesta igualmente.
6. Create user: Al dar click en este botón, si los datos del registro son válidos, se mostrara un mensaje indicando que fue exitoso, se enviara un mensaje de bienvenida al correo electrónico proporcionado en este formulario, indicando que el usuario ya está dentro del sistema. Si los datos no son válidos el sistema indica el error.

Figura 19. Formulario de registro ECWEB

4.4 AUTENTICAR

El formulario de autenticación se puede apreciar en la Figura 20 y cuenta con los siguientes campos:

1. User Name: Nombre de usuario digitado en el formulario de registro.
2. Password: Contraseña digitada anteriormente en el formulario de registro.
3. Remember me next time: Esta casilla se marca en caso de que el usuario quiera ser recordado para futuras autenticaciones.
4. Log In: Al dar click en este botón, si los datos son correctos aparece un mensaje indicándolo y se habilita la búsqueda, si no lo son el mensaje indica que algo anda mal y el usuario deberá corregir el error.
5. Register: Enlace que re-direcciona el meta buscador al formulario de registro.
6. ¿Forgot your password?: Enlace que re-direcciona al meta buscador a un formulario donde se le pedirán algunos datos al usuario, si los datos son correctos se enviara al correo electrónico una nueva contraseña generada automáticamente por el sistema.

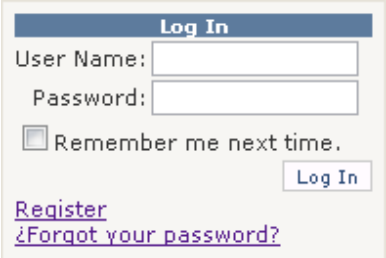


Figura 20. Formulario de autenticación ECWEB

Una vez dentro del sistema se habilitan las opciones que se pueden apreciar en la parte superior Izquierda de la Figura 21 y aparece una etiqueta en letras rojas al lado derecho del botón de búsqueda que indica el método de expansión de consulta que se está utilizando.

Figura 21. Interfaz de usuario autenticado ECWEB

4.5 RECUPERAR CONTRASEÑA

Para la recuperación de contraseña se le solicita al usuario cierta información digitada en el registro e involucra los formularios de la Figura 22 y Figura 23. En el primer formulario, sólo se solicita el nombre de usuario, al presionar el botón submit, si el nombre de usuario es correcto, se re-direccionara al formulario de la Figura 23.

Figura 22. Formulario recuperación de contraseña 1 ECWEB

En el segundo formulario se muestra el nombre de usuario y la pregunta secreta digitada en el registro. El usuario debe contestar esa pregunta tal y como lo hizo en el formulario de registro, al dar click en el botón Submit, si los datos son correctos, el sistema le envía una contraseña generada automáticamente al correo electrónico proporcionado en el formulario de registro.

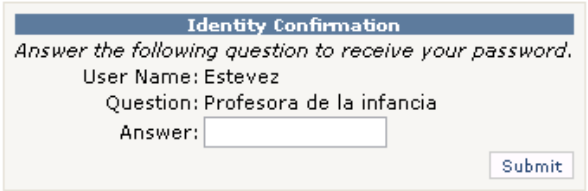


Figura 23. Formulario recuperación de contraseña 2 ECWEB

4.6 CAMBIO DE CONTRASEÑA

Una vez el usuario se ha autenticado, tiene la opción de cambiar la contraseña cuando desee, a través del Enlace “Change Password” que se encuentra en la parte superior izquierda, (ver Figura 21). Este formulario se puede apreciar en la Figura 24 y la información que en éste se solicita es:

1. Password: Contraseña actual
2. New Password: Nueva contraseña
3. Confirm New Password: Confirmación de la nueva contraseña
4. Change Password: Al dar click en este botón se ejecuta el cambio de la contraseña, si los datos digitados en 1,2 y 3 son correctos el cambio será exitoso, aparecerá un mensaje confirmándolo y se envía un mensaje al correo electrónico proporcionado en el registro, en caso contrario el usuario deberá introducir nuevamente la información pedida en 1, 2 y 3.
5. Cancel: Al dar click en este botón se cancela la operación de cambio de contraseña y se re-direccionara a la página principal.

Figura 24. Formulario cambio de contraseña ECWEB

4.7 CONFIGURAR BÚSQUEDA

ECWEB brinda la oportunidad de personalizar las búsquedas, esta configuración se realiza en el enlace “Search options” que se encuentra en la parte superior izquierda justo arriba del Enlace “Change Password”, ver (Figura 21). Las

opciones de configuración que ofrece este formulario se pueden apreciar en la Figura 25.

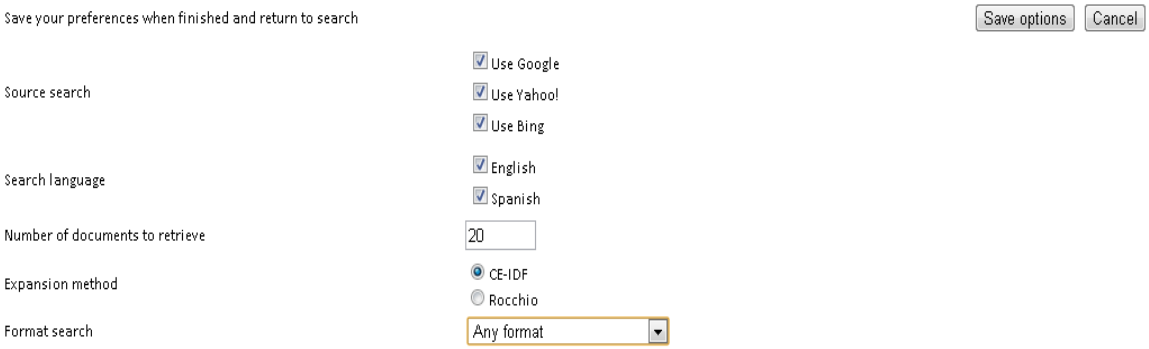


Figura 25. Formulario opciones de búsqueda ECWEB

4.7.1 Fuentes de búsqueda:

Por defecto se usan los tres (3) motores de búsqueda (ver Figura 26). Si no se selecciona ninguna fuente el sistema indica el error.

Figura 26. Formulario opciones de búsqueda - Fuentes de búsqueda ECWEB

4.7.2 Idioma de búsqueda:

Inicialmente se especificó en el plan de proyecto que se trabajaría solo con el idioma inglés, así que esta opción es un aporte adicional, ya que el usuario puede realizar sus búsquedas también en español, por defecto las casillas están seleccionadas, como lo muestra la Figura 27. Si no se selecciona ningún idioma el sistema indica el error.

Figura 27. Formulario opciones de búsqueda – Idioma de búsqueda

4.7.3 Número de documentos a recuperar:

Esta opción permite al usuario especificar cuantos documentos espera que el sistema recupere de los sistemas de búsqueda tradicionales, mediante una caja de texto como lo indica la Figura 28, por defecto el número de documentos a recuperar es de 20. Esta caja de texto sólo acepta números enteros entre 1 y 100. Si el usuario deja la casilla vacía o digita caracteres no validos el sistema indica el error.

Figura 28. Formulario opciones de búsqueda – Número de documentos a recuperar

4.7.4 Método de expansión:

Aunque no estaba originalmente contemplado en el plan de proyecto, se incluyó en ECWEB el método de expansión de consulta basado en Rocchio, método que se usó como punto de comparación en las pruebas realizadas sobre la colección de datos CACM y LISA. De esta forma el usuario puede decidir con que método desea que se realice la expansión de la búsqueda (ver Figura 29).

Figura 29. Formulario opciones de búsqueda – Método de expansión.

4.7.5 Formato de búsqueda:

Esta opción permite buscar documentos en un formato específico seleccionándolo de una lista desplegable como la que muestra la Figura 30, donde se encuentran los formatos más conocidos y usados. Por defecto la búsqueda se hace en cualquier formato (Any format).

Figura 30. Formulario opciones de búsqueda – Formato de búsqueda

Después de que el usuario ha configurado su búsqueda puede guardar los cambios mediante el botón “Save options” o cancelar la configuración hecha mediante el botón “Cancel”, que se encuentran en la parte superior derecha del formulario (ver Figura 25).

4.8 PROCESO DE BÚSQUEDA

4.8.1 AUTOCOMPLETAR

Cuando el usuario está digitando la consulta se despliega una lista de términos para autocompletar la consulta, que ayuda al usuario a formular su consulta brindando sugerencias acorde a lo que está digitando. Esta lista de autocompletar puede ser el resultado de la primera expansión que realiza ECWEB (si ya se tiene un perfil de usuario en el sistema) o puede ser el resultado de un servicio externo, que en este caso se trata del servicio de autocompletar de Google.

4.8.2 PRIMERA EXPANSIÓN DE CONSULTA (AUTOCOMPLETAR ECWEB)

En caso de que el usuario ya tenga perfil, se desplegara una lista propia de ECWEB que se forma de la matriz de coocurrencia de términos, la cual está

constituida por los términos de los documentos evaluados por el usuario (más adelante se muestra la forma como el usuario hace esta evaluación), por lo tanto esta lista de autocompletar es construida de acuerdo a las necesidades de búsqueda del usuario, lo que implica que esta lista varia de usuario a usuario. Para diferenciar la lista de autocompletar ECWEB de la lista autocompletar de Google se colocó un punto al final de cada sugerencia, esta es la primera expansión de consulta que se realiza en ECWEB (ver Figura 31).

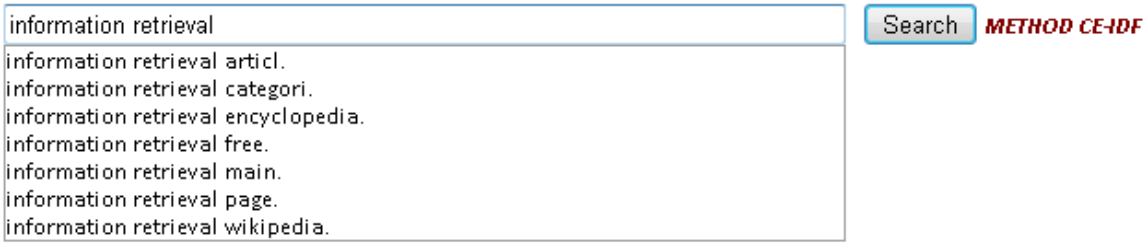


Figura 31. Primera expansión de consulta ECWEB (Autocompletar)

De otro modo si el usuario no tiene aún un perfil se usa un servicio externo, en este caso y como se mencionó anteriormente es el servicio de autocompletar de Google, cuya lista muestra sugerencias sin importar que usuario este registrado (ver Figura 32).

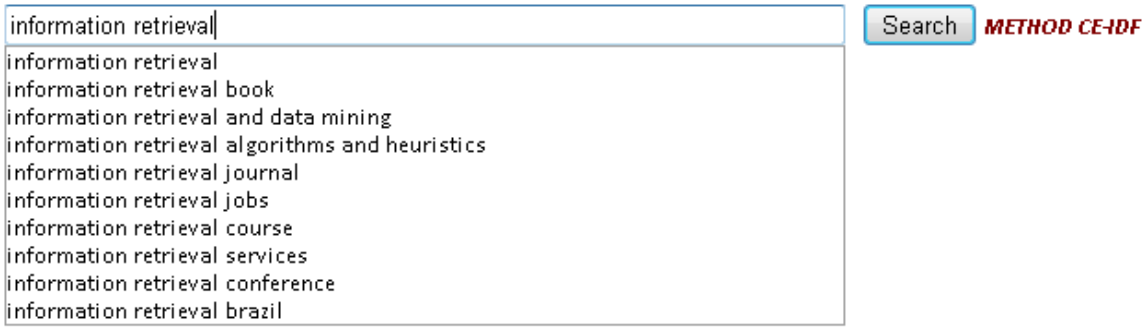


Figura 32. Servicio autocompletar de Google

4.8.3 BÚSQUEDA

Al dar click en el botón buscar una vez que se haya digitado la consulta se despliega una lista de resultados como la que se aprecia en la Figura 33, donde cada resultado tiene un número consecutivo comenzando con el número 1 hasta el número de documentos que este especificado en la configuración de búsqueda (ver Figura 28), además se muestra el título del documento como un enlace, un resumen del mismo (snippet), la url que tiene asignada, el motor de búsqueda de donde se obtuvo ese resultado, y un número que representa el peso de ese documento en relación con la consulta. Para ver cualquiera de los documentos el usuario deberá dar click en el título del documento y se abrirá la página en esa misma ventana o dar click en el icono que está al lado derecho del título del documento que tiene forma de lupa ( ) y se abrirá la página en otra ventana. Un poco más a la derecha se pueden apreciar dos iconos uno es un visto bueno ( ) y el otro una equis ( ), el primero le indica a ECWEB que el documento es relevante (visto bueno) a la búsqueda que está haciendo y el segundo que el documento no es relevante (equis).

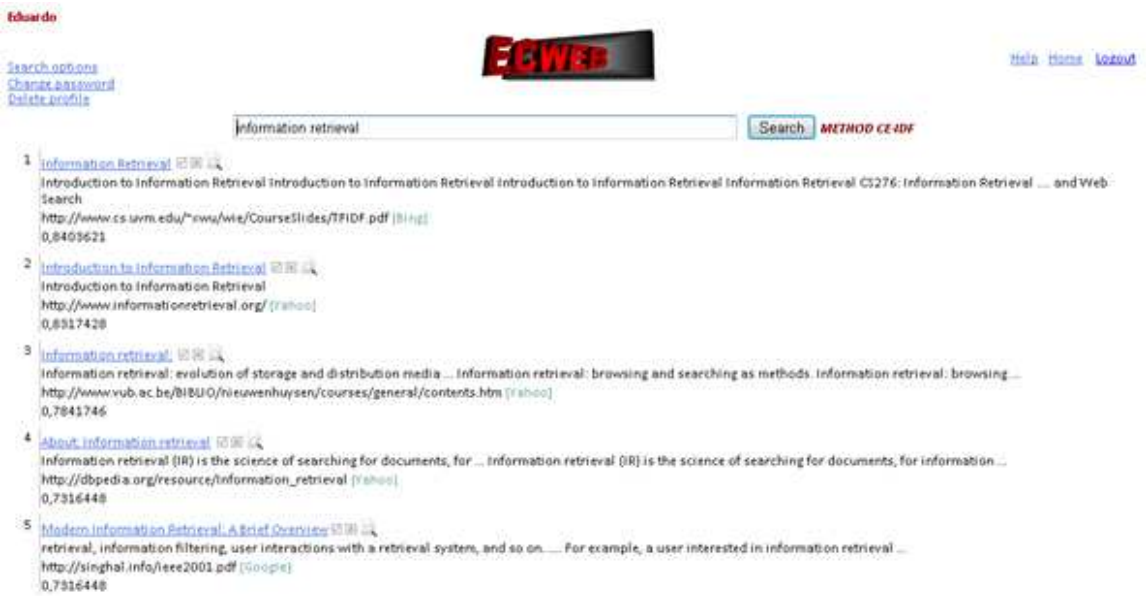


Figura 33. Ejemplo - Búsqueda ECWEB

La Figura 34 muestra de cerca la estructura de un documento.

Figura 34. Ejemplo - Estructura de un resultado

Para este caso en particular los resultados son muy buenos, por lo tanto se va a suponer que el documento número 15 no es relevante mientras que el documento 16 si es relevante, al evaluar los documentos dando click sobre los iconos correspondientes, como se muestra en la Figura 35.

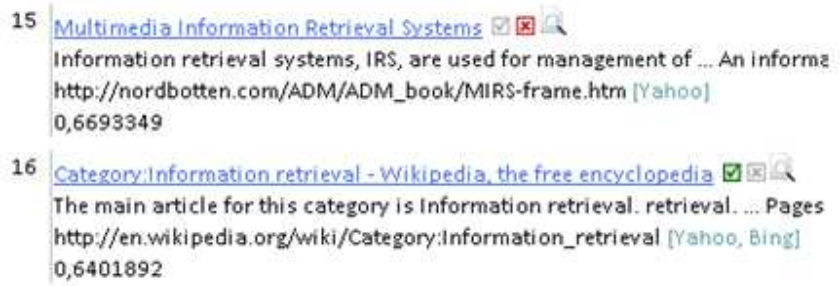


Figura 35. Ejemplo - Evaluación de resultados

Nótese que la equis cambia a color rojo mientras que el visto bueno cambia a color verde, para efectos del ejemplo, luego de evaluar esos dos documentos se repite la búsqueda colocando exactamente lo mismo en la caja de texto, esta vez la lista de documentos recuperados cambió (ver Figura 36), es aquí donde se aprecia la segunda expansión que realiza ECWEB que será presentada a continuación.

4.8.4 SEGUNDA EXPANSIÓN DE CONSULTA (METODO CE-IDF)

Gracias al procedimiento interno que hace el algoritmo CE-IDF al adicionar a la consulta original términos relevantes provenientes de la evaluación que da el usuario a los documentos es posible presentar resultados más precisos en relación al perfil de cada usuario. En el ejemplo que se está presentando, y teniendo en cuenta que se marcó un documento como relevante y uno como no relevante (ver Figura 35), al volver a realizar la misma búsqueda, el documento marcado como relevante ya no ocupa el puesto 16 ahora se encuentra de primero en la lista y su peso también cambio, ahora es de 2,333403 lo que quiere decir que ahora es mucho más importante en relación a la consulta digitada, como lo indica la Figura 36. En cuanto al documento que se marcó como no relevante (ver Figura 35), para este ejemplo en particular no fue recuperado entre los primeros 20 resultados.

Figura 36. Ejemplo - Segunda expansión ECWEB

Este proceso de evaluación de los documentos crea el perfil del usuario aportando los términos para realizar la primera expansión formando la lista autocompletar ECWEB y de igual manera son usados para la segunda expansión de consulta, agregando términos correlacionados a la consulta original digitada por el usuario y de esta forma mejorar los resultados de las búsquedas. La construcción de este perfil se realiza automáticamente y es usado para búsquedas futuras, tanto para la recuperación de documentos, como para construir la lista de autocompletar en ECWEB, cuya construcción se explica a continuación.

4.8.5 DESCRIPCIÓN DE LA MATRIZ DE CORRELACIÓN DE TÉRMINOS

Cada vez que el usuario marca un documento recuperado como relevante o como no relevante, el sistema actualiza los valores que representan el número total de documentos evaluados (N), el número total de documentos relevantes R, el número total de documentos que contienen el término i (ni), el número de documentos relevantes que contienen el término i (ri), y el valor IDF de cada uno de los términos que componen el documento que está siendo evaluado, esto con el objetivo de crear o actualizar el perfil del usuario. Luego de crear o actualizar el perfil se procede a crear la matriz de correlación, para esto se toma cada uno de los términos del documento y se realizan los cálculos con base en la proximidad de los mismos delimitado a un total de 5 términos a la izquierda y 5 a la derecha, el valor de la correlación de cada uno de los términos se registra en la base de datos, de esta forma cada vez que el usuario está digitando la consulta, el sistema accede a la base de datos y recupera los términos que estén correlacionados con el término que el usuario está digitando, mediante la lista autocompletar ECWEB (ver Figura 42). Para ilustrar y detallar el proceso del cálculo de la correlación se presenta a continuación un ejemplo:

Al buscar por “information retrieval” como se muestra en la Figura 37, se despliega una lista de documentos recuperados de los cuales se marca como relevante el quinto documento como se aprecia en la Figura 38.

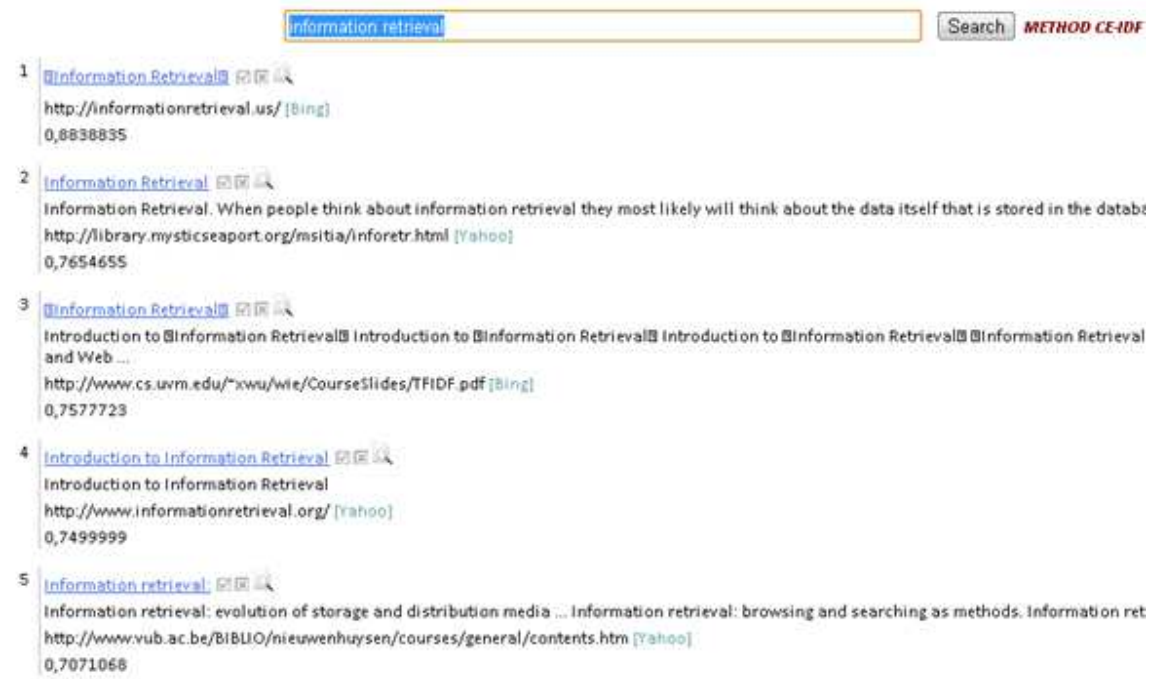


Figura 37. Ejemplo – Búsqueda "information retrieval"

Figura 38. Ejemplo – Documento evaluado "information retrieval"

Internamente el texto con el que se procede a crear la matriz de correlación correspondiente al documento número 5, es el siguiente: "**inform retriev inform retriev evolut storag distribut media inform retriev brows search method inform retriev brows"**, lo primero que se hace es tokenizar el texto y eliminar palabras repetidas, convirtiendo el texto a una lista de palabras, como se aprecia en la Figura 39,

"inform"
"retriev"
"evolut"
"storag"
"distribut"
"media"
"brow"
"search"
"method"

Figura 39. Ejemplo - Términos del documento evaluado "information retrieval"

Luego, de tener la lista de términos se siguen los siguientes pasos:

01: **Repita**

02: **Repita**

03: Para cada término de la lista seleccionar de forma bidireccional el vecino más próximo.

04: Se calcula la correlación entre los dos términos mediante la fórmula de la Figura 40.

05 Se almacena la correlación en el perfil del usuario, específicamente en la matriz de correlación de términos" representada en la Figura 41, que evaluó el documento

06: **Hasta que** se complete un total de 10 (tamaño de la ventana) vecinos o hasta que se recorran todos los términos de la lista de términos del documento.

07: **Hasta que** se haya recorrido toda la lista de términos en el documento evaluado.

$$c_{i,l} = \frac{n_{i,l}}{n_i + n_l - n_{i,l}}$$

Figura 40. Fórmula – Cálculo de la correlación de términos

De la Figura 40 se tiene que: $c_{i,l}$ es el valor de la correlación entre el término i y el término l, donde n_i es el número de documentos que contienen el término i, (n_l) es el número de documentos que contienen el término l, y $n_{i,l}$ es el número de documentos donde aparecen los dos términos al mismo tiempo y en la misma ventana. La correlación toma valores entre 0 y 1, si la correlación es 1 indica que los términos están totalmente correlacionados, por el contrario si este valor es 0 indica que no existe correlación entre ellos.

Eduardo	inform	retriev	evolut	storag	distribut	media	brow	search	method
inform	X	1	1	1	1	1	1	1	1
retriev	1	X	1	1	1	1	1	1	1
evolut	1	1	X	1	1	1	1	1	1
storag	1	1	1	X	1	1	1	1	1
distribut	1	1	1	1	X	1	1	1	1
media	1	1	1	1	1	X	1	1	1
brow	1	1	1	1	1	1	X	1	1
search	1	1	1	1	1	1	1	X	1
method	1	1	1	1	1	1	1	1	X

Figura 41. Ejemplo - Matriz de correlación co-ocurrencia de términos

Para este caso específico, los valores de la correlación entre cada uno de los términos es 1, porque es el primer documento que se evalúa y además es relevante, así que el valor de la correlación entre el término **"information" ("inform")** y el término **"retrieval" ("retriev")** es 1, entre el término **"information" ("inform")** y el término **"evolution" ("evolut")** es 1 e igualmente para los valores de la correlación de los demás términos como se aprecia en la Figura 41.

Una vez el sistema termina de realizar los cálculos, al realizar la consulta nuevamente la lista de autocompletar refleja los cálculos de la correlación, trayendo los términos correlacionados con el término **"retrieval" ("retriev"),** cuyo

valor de correlación es mayor a cero (ver Figura 42), estos términos son mostrados en la lista, ordenados alfabéticamente.

Figura 42. Ejemplo - Lista autocompletar ECWEB

4.9 ELIMINAR PERFIL

La Figura 43 muestra las opciones que cada usuario tiene a su disposición, las cuales son:

1. Delete all profile: Elimina tanto el perfil CE-IDF, como el perfil de Rocchio.
2. Delete profile CE-IDF: Sólo elimina el perfil CE-IDF.
3. Delete profile Rocchio: Sólo elimina el perfil de Rocchio.
4. Save: Al dar click en este botón se lleva a cabo la opción que fue seleccionada en 1, 2 o 3.
5. Cancel: Al dar click en este botón se cancela la operación, lo que implica que se conservan los perfiles.

Figura 43. Formulario eliminar perfil ECWEB

Por último el usuario puede cerrar sesión a través del enlace “logout”, como se muestra en la Figura 44.

Figura 44. Cerrar sesión en ECWEB

4.10 CASO DE USO ECWEB

La Figura 45 muestra un resumen de la funcionalidad del sistema, representando la interacción descrita anteriormente con el usuario de una forma muy sencilla.

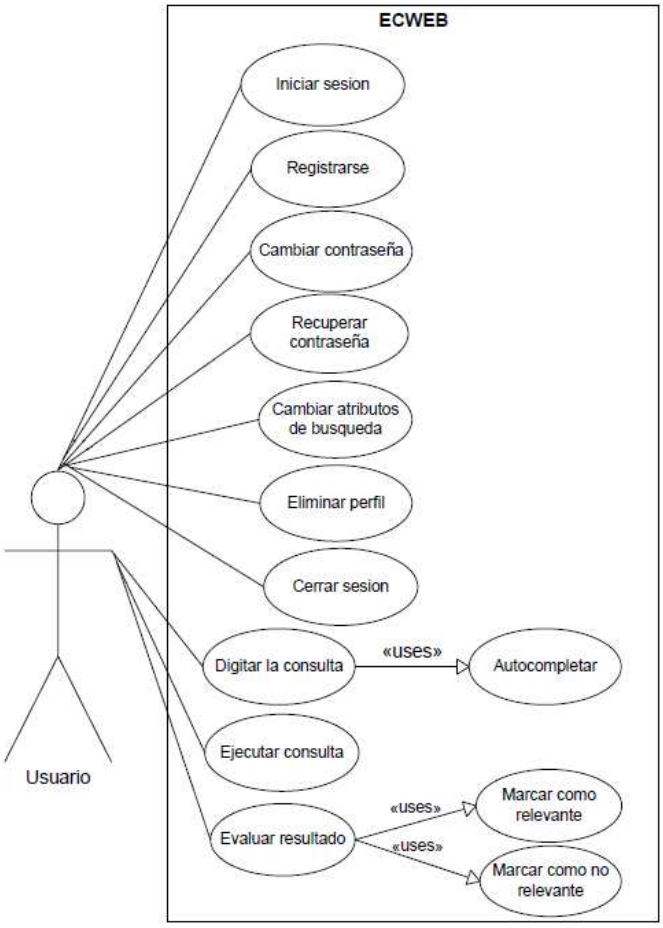


Figura 45 Casos de uso – ECWEB

4.11 ARQUITECTURA DEL SOFTWARE

La arquitectura usada para el desarrollo del proyecto es la arquitectura por capas o arquitectura multi-capa. En esta arquitectura se le confía a cada nivel una misión simple, lo que permite el diseño de sistemas/aplicaciones escalables (que pueden extenderse con facilidad en caso de que las necesidades aumenten). La ventaja principal de esta arquitectura es que el desarrollo se puede llevar a cabo en varios niveles y, en caso de que sobrevenga algún cambio, sólo se afecta al nivel requerido sin tener que revisar código mezclado de diversas capas. En este proyecto se definieron 3 capas, a saber:

Capa de presentación: Esta capa reúne todos los aspectos del software que tienen que ver con las interfaces y la interacción con los usuarios. Estos aspectos típicamente incluyen el manejo de las ventanas, menús y gráficos. Su principal responsabilidad es mostrar información al usuario, interpretar los comandos de este y realizar algunas validaciones simples de los datos ingresados [30]. Esta capa se comunica únicamente con la capa de negocio. En esta capa se presenta la interfaz del meta buscador web, una interfaz amigable, entendible y fácil de usar para el usuario.

Capa de reglas de negocio: Esta capa reúne todos los aspectos del software que tienen que automatizar o apoyar los aspectos del negocio que llevan a cabo los usuarios. Estos aspectos típicamente incluyen las tareas que forman parte de los procesos, las reglas y restricciones que aplican. Esta capa se comunica con la capa de presentación para recibir las peticiones del usuario y enviar las respuestas tras el proceso, y con la capa de datos para solicitar al gestor de bases de datos almacenar o recuperar datos de él [31]. En esta capa se desarrolló la programación de los algoritmos propuestos, la implementación de la función de

evaluación y la matriz de correlación de términos y en general, la lógica que hace posible el buen funcionamiento de ECWEB.

Capa de Servicios: Esta capa reúne los aspectos del software que tienen que ver con la persistencia de los datos y el llamado a servicios remotos. Es la encargada de responder a las solicitudes de almacenamiento o recuperación de datos que realiza la capa de negocio. Está formada por un gestor de bases de datos que realiza el almacenamiento de los datos [31], en este caso, Microsoft Sql Server es el encargado del almacenamiento de los datos del usuario y de la matriz de correlación de términos (matriz única para cada usuario). Adicionalmente, en esta capa se realiza el llamado de los servicios externos de Google, Yahoo! y Live para obtener los resultados iniciales de las búsquedas y el llamado del servicio de autocompletar de Google.

Figura 46 Arquitectura Tres capas adatado de [30]

4.12 DIAGRAMA DE CLASES – ECWEB

A continuación se presenta el diagrama de clases de la aplicación, dividida en 3 figuras, una por cada capa de la arquitectura.

En la Figura 47 se aprecian las clases que conforman la capa de presentación.

A continuación una breve descripción:

MasterPage: Se encarga de la comunicación entre todas las clases de la capa.
FrmCreateUser: Se encarga del registro del usuario.
FrmLogin: Se encarga de realizar la autenticación del usuario en el sistema.
Usuarios_FrmCambiarContraseña: Se encarga de realizar el cambio de contraseña.
Usuarios_FrmRestablecerContraseña: Se encarga de generar una contraseña en caso de pérdida u olvido.
FrmOpciones: Se encarga de gestionar todas las opciones de búsqueda.
FrmEliminarPerfil: Se encarga de limpiar el historial que el usuario haya creado mediante las evaluaciones hechas sobre los documentos.
_Default: Se encarga de gestionar todo lo referente a las consultas, recibe y ejecuta la consulta, presenta los resultados de la consulta al usuario y recibe las evaluaciones que el usuario imparte a los documentos, además es el enlace a la capa de lógica de negocio, se comunica con la clase Autocompletar, Parametros, Procesamiento, PerfildelUsuarioRocchio, PerfilDelUsuarioIDF.

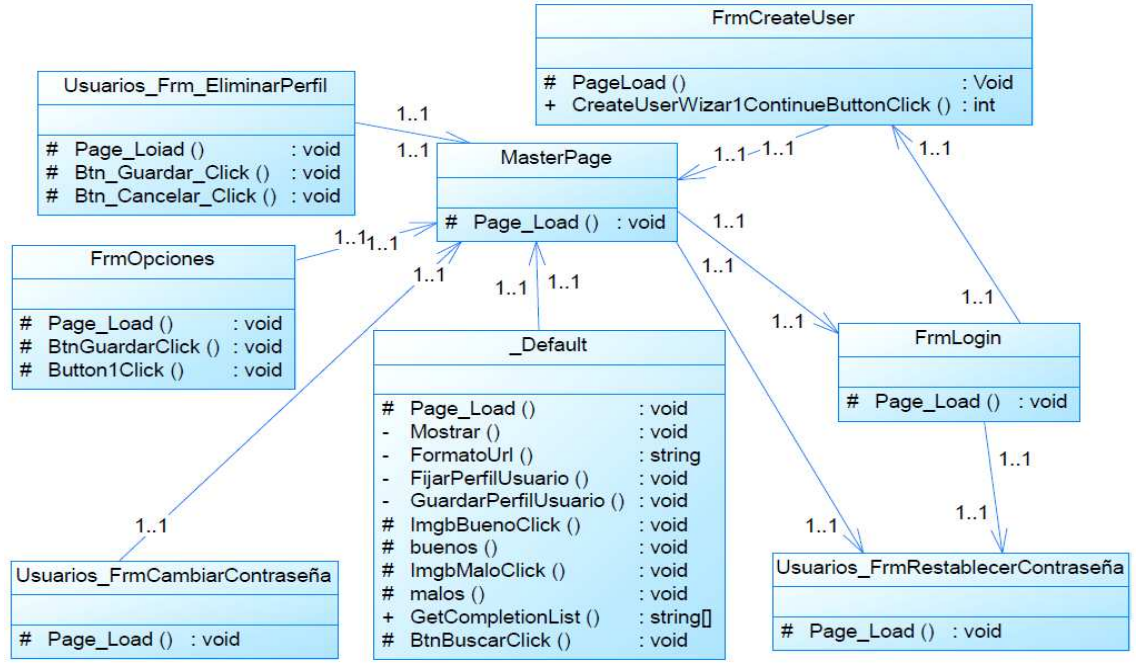


Figura 47 Diagrama de clases - primera parte

La Figura 48 muestra las clases que componen la capa de reglas del negocio.

A continuación una breve descripción:

Procesamiento: Esta clase es la encargada de gestionar todo el proceso de búsqueda, desde que se ejecuta la consulta hasta que los resultados son presentados al usuario.

Busqueda: Se encarga de centralizar el proceso de búsqueda en los diferentes motores tradicionales. Usa hilos para hacer más rápida y eficiente la consulta de resultados en Internet.

Parametros: Se encarga de comunicarle a la clase búsqueda cuales parámetros debe tener en cuenta para realizar la búsqueda.

Indexardocumentos: Se encarga de tomar los documentos obtenidos con los diferentes motores e indexarlos.

ExpansionRocchio: Se encarga de realizar el proceso de expansión de consulta implementando el algoritmo de Rocchio.

ExpansionCEIDF: Se encarga de realizar el proceso de expansión de consulta implementando el algoritmo CE-IDF.

PerfilDelUsuarioRocchio: Se encarga de actualizar todos los valores que están involucrados en el proceso de expansión Rocchio.

PerfilDelUsuarioCEIDF: Se encarga de actualizar todos los valores que están involucrados en el proceso de expansión CEIDF y realiza la actualización de las variables involucradas para construir o actualizar la matriz de co-ocurrencia de términos.

RecuperarDcumentos: Se encargar de recuperar los documentos luego de haberse realizado el proceso de expansión de consulta.

ASP.NETmembership: Se encarga de comunicar a las clases de la capa presentación con la capa de servicios, con el fin de poder acceder a los datos del usuario.

Autocompletar: Esta clase se encarga de gestionar que lista autocompletar ejecutara.

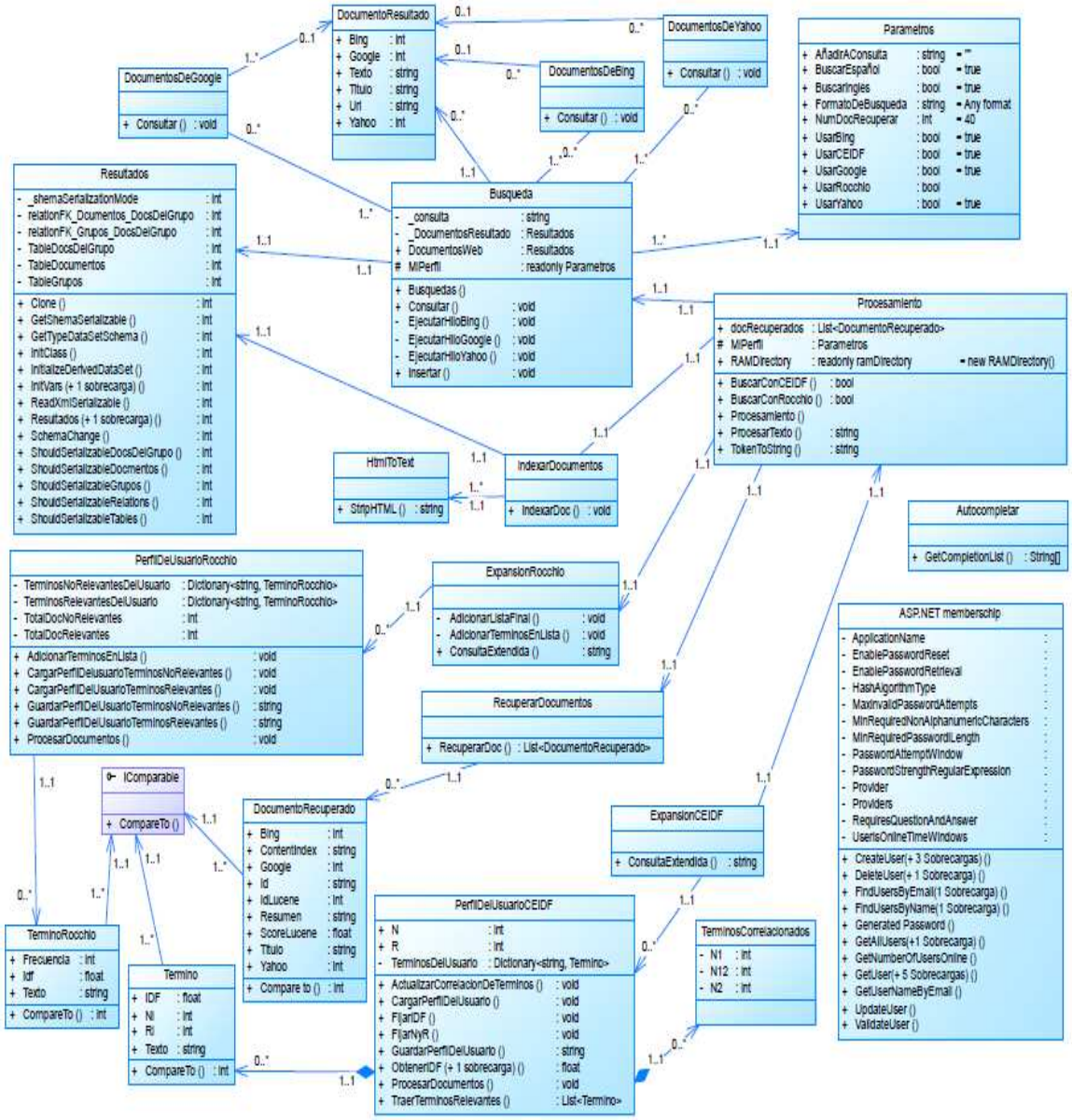


Figura 48 Diagrama de clases – segunda parte

En la Figura 49, se encuentran las clases que componen la capa de servicio.

A continuación una breve descripción:

BaseDeDatos: Se encarga de almacenar todos los datos del usuario incluidos la matriz de correlación de términos en la base de datos.
Autocompletar: Se encarga de acceder al servicio de autocompletar de Google.
StopWordList: Se encarga de eliminar las palabras vacías de las consultas y documentos.
Texto: Procesa el texto de los documentos recuperados filtrando los caracteres especiales para presentar el texto lo más limpio posible al usuario.
Las clases **SpanishStemmer**, **SnoballProgram**, **Among**, son las encargadas de llevar el texto en español a su raíz léxica.
Las clases **Idioma** e **IdiomaResultado** se encargan de detectar el idioma de la consulta.
Las clase **YahooSearchService**, **CacheType**, **ResultType**, **ResulSet**, acceden al servicio de búsqueda de Yahoo, para ejecutar la consulta con este motor.
Las clases **GoogleSearchService**, **GoogleSearchResults** y **GoogleSearchResult**, acceden al servicio de búsqueda de Google, para ejecutar la consulta con este motor.

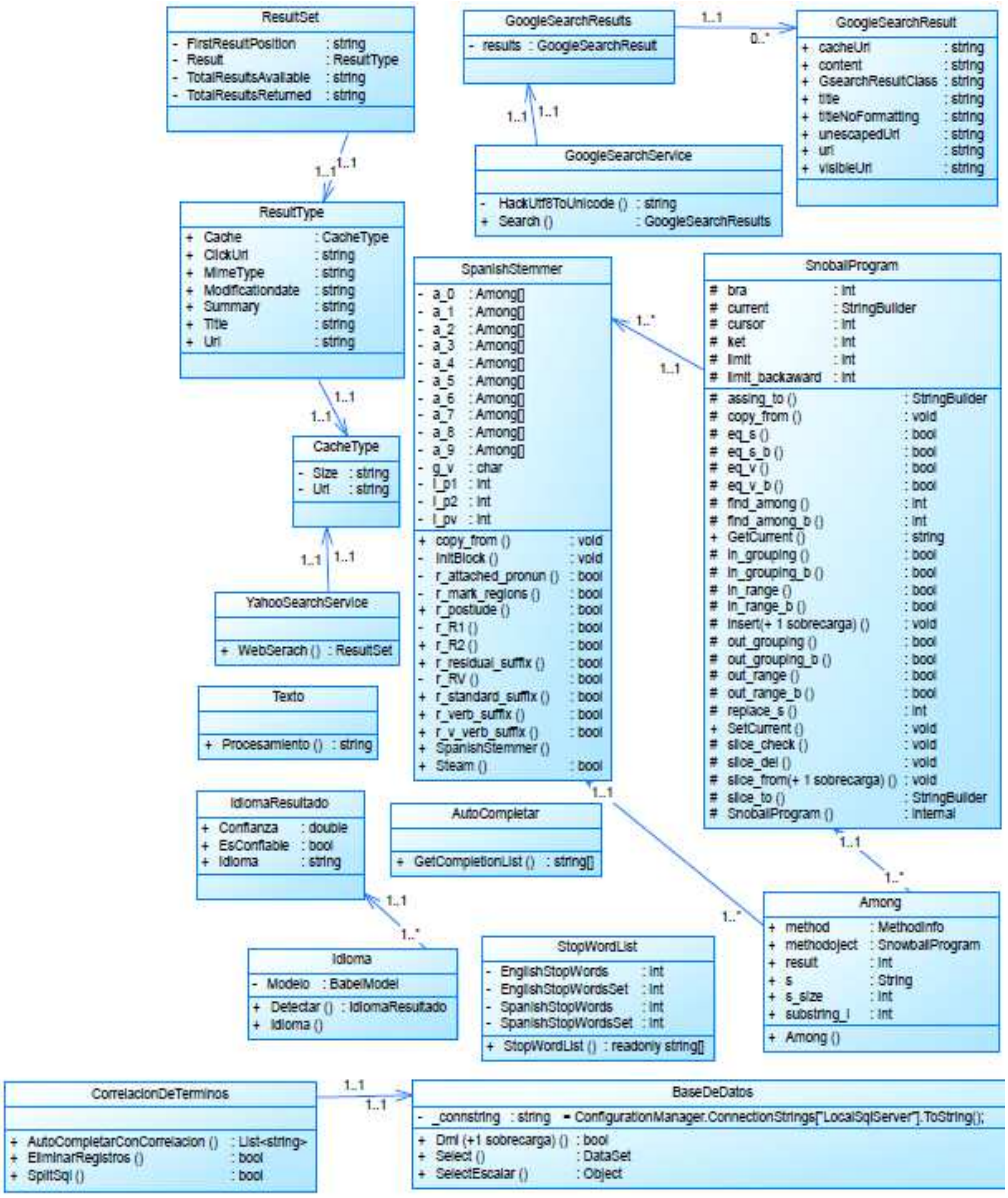


Figura 49 Diagrama de clases – tercera parte

4.13 PRUEBAS CON LOS USUARIOS

Para dar cumplimiento al objetivo 5 se realizaron pruebas con estudiantes de diferentes niveles del programa de ingeniería de sistemas de la Universidad del Cauca, Popayán.

En total se hicieron 3 pruebas sobre ECWEB, todas hechas de la misma forma, para cada prueba se seleccionaron 3 consultas de acuerdo al conocimiento de cada grupo de usuarios, luego para cada consulta seleccionada se realizan 3 iteraciones de búsqueda y para cada iteración se evalúan los primeros 6 documentos, es decir, se realiza la primera consulta (primera iteración) y se evalúan los primeros 6 documentos, luego se repite la consulta (segunda iteración) y se realiza nuevamente la evaluación de los primeros 6 documentos, esa misma consulta se digita por tercera vez (tercera iteración) y se realiza nuevamente la evaluación de los primeros 6 documentos, lo mismo se hace para la segunda y tercera consulta.

4.13.1 PRUEBA 1

Esta prueba se realizó con diecisiete (17) estudiantes de Ingeniería de Sistemas de primer semestre de la universidad del Cauca, pertenecientes al curso “Introducción a la Ingeniería de Sistemas”, orientado por la Ing. Jimena Timaná. Las consultas utilizadas fueron:

1. “Que es un motor de búsqueda”
2. “Que es un meta buscador”
3. “Que es un índice de búsqueda”

La Tabla 8 muestra las evaluaciones que dieron los usuarios (u1, u2,….u17) a cada uno de los primeros 6 documentos recuperados, para cada una de las 3 iteraciones que se hicieron con la primer consulta. Los estudiantes evaluaban cada resultado 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 (ver Tabla 9) y tercer consulta (Tabla 10).

Tabla 8. Prueba 1 "Que es un motor de búsqueda"– Resultados de la consulta

		Que es un motor de búsqueda														
ITERACIÓN	RESULTADO	U1	U3	U4	U5	U6	U7	U8	U10	U11	U12	U13	U14	U15	U16	U17
1	1	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
	2	N	N	N	N	N	N	N	N	N	N	N	N	N	R	N
	3	X	X	X	N	N	N	X	X	X	X	X	X	X	X	X
	4	N	N	N	N	R	N	X	N	R	N	N	X	N	N	N
	5	R	R	R	R	R	R	X	R	R	R	R	N	N	N	R
	6	R	R	N	R	R	N	N	R	N	N	N	R	R	X	R
2	1	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
	2	R	R	R	R	R	R	N	R	R	R	R	N	R	R	R
	3	R	X	N	R	R	X	X	R	X	R	N	R	N	N	R
	4	R	R	R	X	R	N	R	R	R	N	R	R	N	N	N
	5	N	N	N	N	R	R	N	R	N	R	N	N	R	N	R
	6	R	R	N	N	R	N	N	R	N	N	N	R	N	X	X
3	1	R	R	R	R	R	R	R	R	R	R		R	R	R	R
	2	R	N	R	R	R	R	N	R	N	R		N	N	R	R
	3	R	R	R	R	R	R	X	R	R	R		R	R	R	R
	4	R	R	N	R	R	R	R	R	R	R		N	R	N	N
	5	N	R	R	N	R	N	N	R	R	N		R	N	N	R
	6	N	N	R	X	R	N	N	R	N	R		R	R	N	R

En la Tabla 8 se puede apreciar que el usuario 2 (U2) y el usuario (U9) no fueron tenidos en cuenta para realizar los cálculos ya que presentaron desde la primera

iteración varias evaluaciones como inaccesibles (X), debido a problemas de conexión a Internet que tuvieron los estudiantes en la sala de cómputo.

Tabla 9. Prueba 1 "Que es un meta buscador" – Resultados de la consulta

		Que es un meta buscador													
ITERACIÓN	RESULTADO	U1	U2	U3	U4	U5	U6	U7	U9	U10	U11	U12	U13	U14	U15
1	1	R	N	R	N	N	N	N	R	R	N	R	R	R	R
	2	N	N	R	N	N	R	N	N	R	N	R	R	N	N
	3	X	R	N	N	N	R	N	R	N	R	N	N	R	R
	4	N	R	R	R	R	R	N	X	N	R	X	R	R	R
	5	R	N	N	R	R	N	R	R	R	N	R	R	R	R
	6	R	R	R	R	R	N	R	N	N	R	R	N	R	R
2	1	R	R	R	R	R	R	R	R	R	R	R	R		
	2	R	R	R	R	N	N	R	R	R	R	R	R		
	3	X	R	R	R	N	R	R	R	R	R	R	N		
	4	N	R	R	R	R	R	R	R	R	R	R	R		
	5	R	R	N	R	R	N	R	R	R	R	R	N		
	6	R	R	R	R	R	N	N	R	R	R	N	R		
3	1	R	R	R	R	R	R	R	R	R	R	R			
	2	R	R	R	R	R	R	R	R	R	R	R			
	3	X	R	R	R	R	R	R	R	R	R	R			
	4	R	R	R	R	N	R	R	R	R	N	N			
	5	N	N	R	R	N	N	R	R	R	N	R			
	6	R	R	R	R	N	R	R	R	R	R	R			

En la Tabla 9 se excluyó al usuario 8 (u8) debido a que o evaluó todos los documentos en la primera iteración. Además los usuarios 16 y 17 (u16 y u17) no respondieron esta parte de la prueba, solo respondieron lo concerniente a la primera consulta. Además se puede observar que los usuarios 13, 14 y 15 no terminaron toda la evaluación de esta consulta (celdas vacías en la tabla).

Tabla 10. Prueba 1 Que es un índice de búsqueda – Resultados de la consulta

		Que es un índice de búsqueda								
ITERACIÓN	RESULTADO	U2	U3	U4	U5	U6	U7	U8	U9	U10
1	1	R	R	N	N	N	N	R	R	N
	2	R	R	N	R	N	R	R	N	R
	3	N	N	N	N	N	N	N	R	N
	4	R	R	N	R	R	R	R	N	R
	5	N	N	N	N	R	N	X	N	R
	6	N	N	N	N	N	N	N	R	N
2	1	R	R	N	R	R	R	R	R	
	2	R	R	R	R	R	R	R	R	
	3	R	R	R	N	N	N	N	N	
	4	R	R	N	R	N	R	R	R	
	5	N	N	R	N	R	N	N	N	
	6	N	N	R	N	N	N	X	R	
3	1	R	R	R	R	R	R	R	R	
	2	R	R	R	R	R	R	R	R	
	3	R	R	R	R	R	R	N	R	
	4	N	N	N	R	R	R	R	N	
	5	N	N	R	N	N	N	X	R	
	6	N	N	R	N	N	N	N	N	

Para la tercer consulta (ver Tabla 10) se excluyó al usuario 1 (U1) ya que no evaluó todos los documentos de la iteración 1 lo que afecta todos los documentos recuperados y las evaluaciones hechas posteriormente. Además no se tuvieron en cuenta los usuarios 11, 12, 13, 14 y 15 (U11, U12, U13, U14 y U15) debido a que no realizaron esta parte de la prueba.

Se tomaron los resultados de las evaluaciones de la primer consulta (ver Tabla 8), 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 11. Este mismo proceso se hizo para la

segunda y tercer consulta, como resultado se obtuvieron la Tabla 12 y la Tabla 13 respectivamente.

Tabla 11. Prueba 1 "Que es un motor de búsqueda" - Estadísticas

Que es un motor de búsqueda								
ITERACIÓN	RESULTADO	R	N	X	TOTAL	EXACTITUD	PRECISIÓN	PRECISIÓN MEDIA
1	1	15	0	0	15	100,0%	100,0%	49,8%
	2	1	14	0	15	6,7%	53,3%	
	3	0	3	12	15	0,0%	35,6%	
	4	2	11	2	15	13,3%	30,0%	
	5	11	3	1	15	73,3%	38,7%	
	6	8	6	1	15	53,3%	41,1%	
2	1	15	0	0	15	100,0%	100,0%	78,7%
	2	13	2	0	15	86,7%	93,3%	
	3	7	4	4	15	46,7%	77,8%	
	4	9	5	1	15	60,0%	73,3%	
	5	6	9	0	15	40,0%	66,7%	
	6	5	8	2	15	33,3%	61,1%	
3	1	14	0	0	14	100,0%	100,0%	82,9%
	2	9	5	0	14	64,3%	82,1%	
	3	13	0	1	14	92,9%	85,7%	
	4	10	4	0	14	71,4%	82,1%	
	5	7	7	0	14	50,0%	75,7%	
	6	7	6	1	14	50,0%	71,4%	
	TOTALES	152	87	25				

Para la primer consulta (ver Tabla 11) en su primera iteración la precisión oscila entre 41,1% y 100,0%, obteniendo una precisión media de 49,8%, para la segunda iteración la precisión oscila entre 61,1% y 100,0%, obteniendo una precisión media de 78,7% y finalmente para la tercera iteración el valor de la precisión oscila entre 71,4% y 100,0%, obteniendo una precisión media de 82,9%, lo que refleja una mejora de la precisión de una iteración a otra iteración, a medida que el algoritmo EC-IDF procesa las evaluaciones hechas por los usuarios (ver Figura 50).

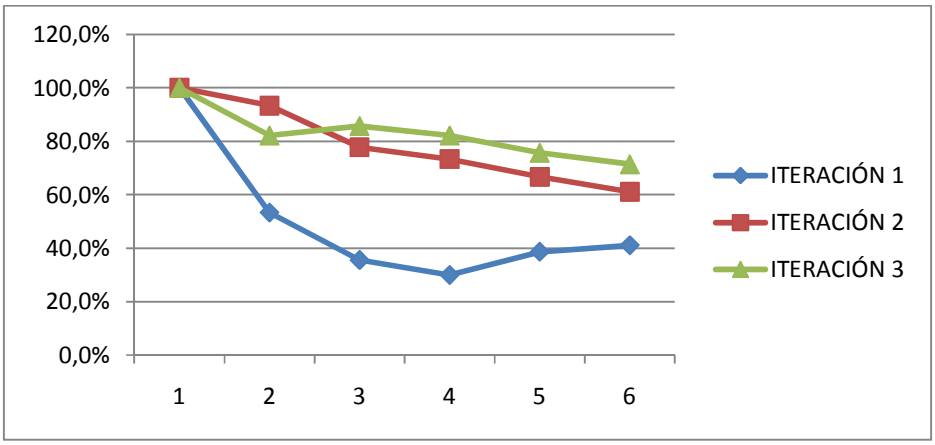


Figura 50. Prueba 1 "Que es un motor de búsqueda"

Tabla 12. Prueba 1 "Que es un meta buscador" - Estadísticas

Que es un meta buscador								
ITERACIÓN	RESULTADO	R	N	X	TOTAL	EXACTITUD	PRECISIÓN	PRECISIÓN MEDIA
1	1	8	6	0	14	57,1%	57,1%	51,7%
	2	5	9	0	14	35,7%	46,4%	
	3	6	7	1	14	42,9%	45,2%	
	4	9	3	2	14	64,3%	50,0%	
	5	10	4	0	14	71,4%	54,3%	
	6	10	4	0	14	71,4%	57,1%	
2	1	12	0	0	12	100,0%	100,0%	88,9%
	2	10	2	0	12	83,3%	91,7%	
	3	9	2	1	12	75,0%	86,1%	
	4	11	1	0	12	91,7%	87,5%	
	5	9	3	0	12	75,0%	85,0%	
	6	9	3	0	12	75,0%	83,3%	
3	1	11	0	0	11	100,0%	100,0%	92,7%
	2	11	0	0	11	100,0%	100,0%	
	3	10	0	1	11	90,9%	97,0%	
	4	8	3	0	11	72,7%	90,9%	
	5	6	5	0	11	54,5%	83,6%	
	6	10	1	0	11	90,9%	84,8%	
	TOTALES	164	53	5				

Para la segunda consulta (ver Tabla 12) en su primera iteración la precisión oscila entre 45,2% y 57,1%, obteniendo una precisión media de 51,7%, para la segunda iteración la precisión oscila entre 83,3% y 100%, obteniendo una precisión media de 88,9% y finalmente para la tercera iteración el valor de la precisión oscila entre 83,6% y 100,0%, obteniendo una precisión media de 92,7%, lo que refleja una mejora de la precisión de una iteración a otra iteración, a medida que el algoritmo EC-IDF procesa las evaluaciones hechas por los usuarios (ver Figura 51).

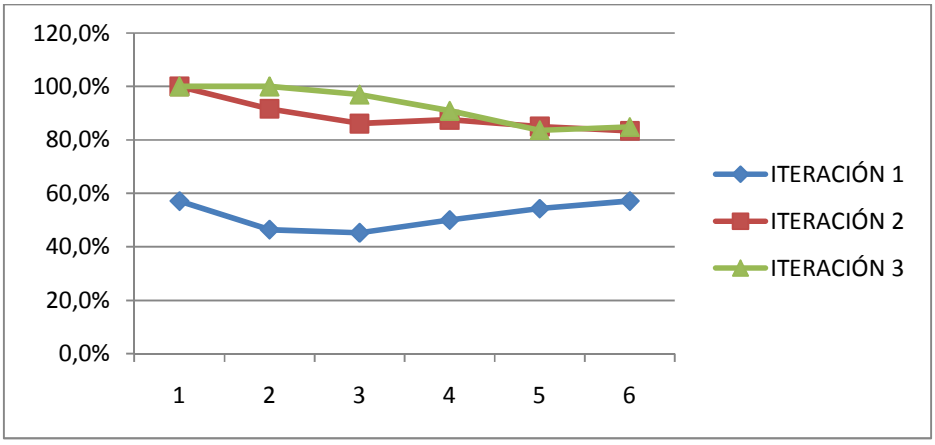


Figura 51. Prueba 1 "Que es un meta buscador"

Tabla 13. Prueba 1 "Que es un índice de búsqueda" - Estadísticas

Que es un índice de búsqueda								
ITERACIÓN	RESULTADO	R	N	X	TOTAL	EXACTITUD	PRECISIÓN	PRECISIÓN MEDIA
1	1	4	5	0	9	44,4%	44,4%	45,7%
	2	6	3	0	9	66,7%	55,6%	
	3	1	8	0	9	11,1%	40,7%	
	4	7	2	0	9	77,8%	50,0%	
	5	2	6	1	9	22,2%	44,4%	
	6	1	8	0	9	11,1%	38,9%	
2	1	7	1	0	8	87,5%	87,5%	75,8%
	2	8	0	0	8	100,0%	93,8%	
	3	3	5	0	8	37,5%	75,0%	

	4	6	2	0	8	75,0%	75,0%	
	5	2	6	0	8	25,0%	65,0%	
	6	2	5	1	8	25,0%	58,3%	
3	1	8	0	0	8	100,0%	100,0%	85,9%
	2	8	0	0	8	100,0%	100,0%	
	3	7	1	0	8	87,5%	95,8%	
	4	4	4	0	8	50,0%	84,4%	
	5	2	5	1	8	25,0%	72,5%	
	6	1	7	0	8	12,5%	62,5%	
	TOTALES	79	68	3				

Para la tercer consulta (ver Tabla 13) en su primera iteración la precisión oscila entre 38,9% y 55,6%, obteniendo una precisión media de 45,7%, para la segunda iteración la precisión oscila entre 58,3% y 93,8%, obteniendo una precisión media de 75,8% y finalmente para la tercera iteración el valor de la precisión oscila entre 62,5% y 100%, obteniendo una precisión media de 85,9%, lo que refleja una mejora de la precisión de una iteración a otra iteración, a medida que el algoritmo EC-IDF procesa las evaluaciones hechas por los usuarios (ver Figura 52).

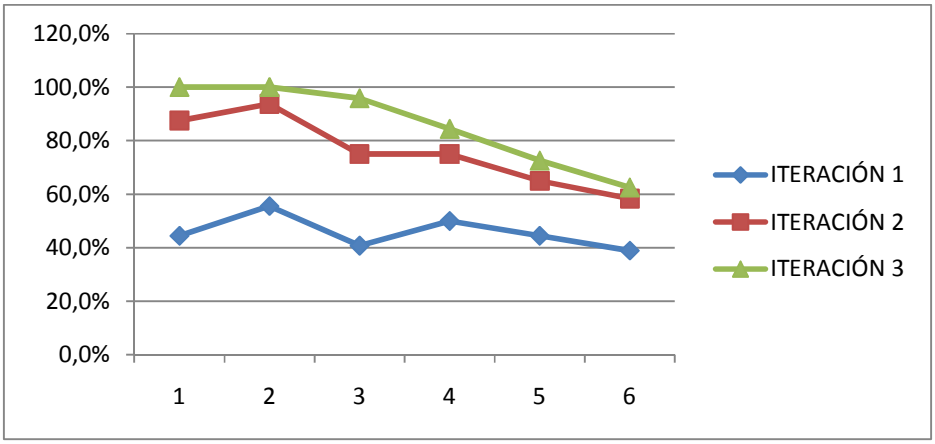


Figura 52. Prueba 1 "Que es un índice de búsqueda"

A continuación se presentan los resultados de 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 solo se puede hacer para la primer iteración de la primer consulta en cada uno de las 3 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 14).

Tabla 14. Prueba 1 – Kappa de Fleiss

Que es un motor de búsqueda							
ITERACIÓN	RESULTADO	R	N	X	TOTAL	Pi	
1	1	15	0	0	15	1,00	
	2	1	14	0	15	0,87	
	3	0	3	12	15	0,66	
	4	2	11	2	15	0,54	
	5	11	3	1	15	0,55	
	6	8	6	1	15	0,41	
	TOTAL	37	37	16	90	4,03	
	Pr	0,41	0,41	0,18		0,67	P-
	Pr^2	0,17	0,17	0,03	0,37	Pe-	

Kappa de Fleiss:	0,479	Moderate Concordance

El valor de Kappa de Fleiss es de 0,479 lo que quiere decir que hubo un acuerdo moderado entre los usuarios al evaluar los 6 documentos en la primera iteración de la primer consulta (ver Tabla 1).

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 15.

Tabla 15. Prueba 1 – Las 3 consultas - Estadísticas

	CONSULTA	1	2	3		
ITERACIÓN	RESULTADOS	PRECISIÓN	PRECISIÓN	PRECISIÓN	PRECISIÓN TOTAL	PRECISIÓN TOTAL MEDIA
1	1	100,0%	57,1%	44,4%	67,2%	49,1%
	2	53,3%	46,4%	55,6%	51,8%	
	3	35,6%	45,2%	40,7%	40,5%	
	4	30,0%	50,0%	50,0%	43,3%	
	5	38,7%	54,3%	44,4%	45,8%	
	6	41,1%	57,1%	38,9%	45,7%	
2	1	100,0%	100,0%	87,5%	95,8%	81,1%
	2	93,3%	91,7%	93,8%	92,9%	
	3	77,8%	86,1%	75,0%	79,6%	
	4	73,3%	87,5%	75,0%	78,6%	
	5	66,7%	85,0%	65,0%	72,2%	
	6	61,1%	83,3%	58,3%	67,6%	
3	1	100,0%	100,0%	100,0%	100,0%	87,2%
	2	82,1%	100,0%	100,0%	94,0%	
	3	85,7%	97,0%	95,8%	92,8%	
	4	82,1%	90,9%	84,4%	85,8%	
	5	75,7%	83,6%	72,5%	77,3%	
	6	71,4%	84,8%	62,5%	72,9%	

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 6 documentos por cada iteración.

Para las tres consultas en la primera iteración (ver Tabla 15) la precisión oscila entre 40,5% y 67,2%, obteniendo una precisión media de 49,1%, para la segunda iteración la precisión oscila entre 67,6% y 95,8%, obteniendo una precisión media de 81,1% y finalmente para la tercera iteración el valor de la precisión oscila entre 72,9% y 100%, obteniendo una precisión media de 87,2%, lo que refleja una mejora de la precisión de una iteración a otra iteración, a medida que el algoritmo EC-IDF procesa las evaluaciones hechas por los usuarios (ver Figura 53).

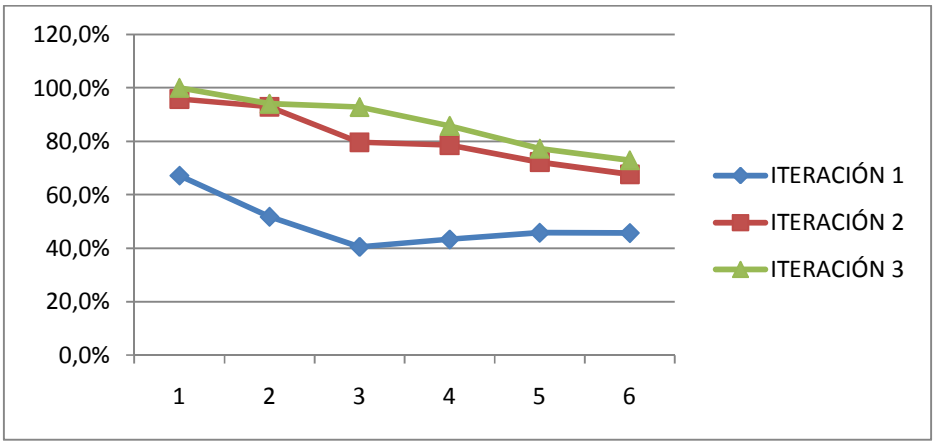


Figura 53. Prueba 1 totales – 3 consultas

En general la Figura 53 muestra como desde la segunda iteración la precisión mejora considerablemente, lo que quiere decir que desde que el algoritmo CE-IDF empieza a recibir las evaluaciones dadas por los usuarios inmediatamente muestra una mejora considerable y al aumentar estas iteraciones sigue mejorando.

4.13.2 PRUEBA 2

Se realizó una segunda prueba con trece (13) estudiantes de Ingeniería de Sistemas de séptimo semestre de la universidad del Cauca pertenecientes al curso “Inteligencia artificial”, orientado por el Ing. Esp. Ember Martínez. Esta vez las consultas fueron las siguientes:

1. “Inferencia Fuzzy”
2. “Aplicación de la lógica difusa”
3. “Sistemas expertos”

La Tabla 16, la Tabla 17 y la Tabla 18 muestran las evaluaciones que dieron los usuarios a cada una de las 3 consultas.

Tabla 16. Prueba 2 "Inferencia Fuzzy" – Resultados de la consulta

		Inferencia Fuzzy												
ITERACIÓN	RESULTADO	U1	U2	U3	U4	U5	U6	U7	U8	U9	U10	U11	U12	U13
1	1	R	N	R	R	R	R	R	R	R	R	R	R	R
	2	R	N	R	R	N	N	N	R	R	R	R	R	R
	3	N	N	N	N	N	N	N	N	N	R	N	N	N
	4	R	R	N	R	R	R	R	R	R	R	N	R	R
	5	N	R	N	N	N	N	N	N	N	R	R	N	N
	6	R	N	R	R	R	R	R	R	N	R	R	R	R
2	1	R	N	R	R	R	R	R	R	R	R	R	R	R
	2	R	R	R	R	R	R	R	R	R	R	R	R	R
	3	N	R	N	N	R	X	R	R	N	R	R	N	R
	4	R	R	R	R	R	R	R	N	R	R	R	R	R
	5	N	N	R	N	N	R	N	N	N	R	N	R	R
	6	R	N	R	R	R	R	N	N	R	R	R	R	R
3	1	R	R	R	R	R	R	R	R	R	R	R	R	R
	2	R	R	R	R	R	R	R	R	N	R	R	R	R
	3	R	R	R	R	R	R	R	N	R	R	R	N	R
	4	R	R	R	R	R	R	R	N	R	R	R	R	R
	5	R	R	R	R	R	R	N	N	R	R	R	N	R
	6	R	N	R	N	N	R	R	N	R	R	N	R	R

Tabla 17. Prueba 2 "Aplicación de la lógica difusa" – Resultados de la consulta

		Aplicación de la lógica difusa											
ITERACIÓN	RESULTADO	U1	U2	U3	U4	U5	U7	U8	U9	U10	U11	U12	U13
1	1	R	N	N	R	R	N	R	X	R	R	X	R
	2	R	R	R	R	N	R	N	R	R	R	N	N
	3	R	N	N	R	R	R	X	R	R	R	N	N
	4	R	R	N	R	N	R	R	N	R	N	R	R
	5	N	R	N	R	R	N	N	R	R	N	R	N
	6	N	N	R	R	R	R	R	N	R	N	R	N
2	1	R	R	R	R	R	N	R	R	R	N	N	
	2	R	N	R	R	R	N	R	R	R	N	N	
	3	N	R	N	R	R	R	R	N	R	R	R	
	4	R	R	R	R	R	R	R	N	R	R	X	
	5	X	R	N	R	N	R	R	R	R	R	R	
	6	R	R	N	R	R	R	N	R	R	R	R	

3	1	R	R	R	R	R	R	R	R	R	R	N	
	2	R	R	R	R	R	R	R	R	R	R	N	
	3	R	R	R	R	R	R	R	N	R	R	R	
	4	R	R	X	R	R	R	R	R	R	R	R	
	5	X	R	R	R	R	R	R	R	R	R	R	
	6	R	R	N	R	N	R	R	N	R	N	N	

El usuario 6 (U6) fue excluido ya que no evaluó todos los documentos en la primera iteración, estas evaluaciones incompletas afecta los resultados obtenidos en la iteración 2 y 3 para este usuario

Tabla 18. Prueba 2 "Sistemas expertos" – Resultados de la consulta

		Sistemas expertos										
ITERACIÓN	RESULTADO	U1	U2	U3	U4	U6	U7	U8	U9	U10	U11	U12
1	1	R	N	R	R	R	R	R	R	R	N	N
	2	N	N	R	R	R	R	R	R	R	R	R
	3	R	N	R	N	R	R	R	R	R	N	N
	4	R	R	N	R	N	R	R	R	R	N	R
	5	R	R	R	R	R	R	N	R	R	R	X
	6	R	R	R	R	R	X	R	R	R	R	X
2	1	R	R	R	R	R	R	R	R	R	R	N
	2	R	R	R	R	N	R	R	R	R	R	R
	3	R	N	R	R	R	R	R	N	N	N	N
	4	R	N	R	R	N	R	R	N	R	R	R
	5	R	R	R	R	R	R	R	N	R	N	N
	6	R	R	R	R	N	N	R	R	R	R	R
3	1	R	R	R	R	X	R	R	R	R	R	N
	2	R	R	R	R	R	R	R	R	R	R	R
	3	R	R	R	R	R	X	R	N	R	R	N
	4	R	R	R	R	R	N	R	R	R	N	R
	5	R	N	R	R	R	R	R	N	R	R	N
	6	R	R	R	R	R	X	R	R	R	N	N

Para esta consulta se excluyó al usuario 5 (U5) debido a que se evidenciaba una discordancia en su criterio en cada una de las iteraciones, es decir, no fue

consecuente con sus evaluaciones y el usuario 13 (U13) que no completo esta parte de la prueba.

Tomando los resultados de la Tabla 16, 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 lo muestra la Tabla 19. Este mismo proceso se hizo para la segunda y tercer consulta, como resultado se obtuvieron la Tabla 17 y la Tabla 18 respectivamente.

Tabla 19. Prueba 2 "Inferencia Fuzzy" - Estadísticas

Inferencia Fuzzy								
ITERACIÓN	RESULTADO	R	N	X	TOTAL	EXACTITUD	PRECISIÓN	PRECISIÓN MEDIA
1	1	12	1	0	13	92,3%	92,3%	68,1%
	2	9	4	0	13	69,2%	80,8%	
	3	1	12	0	13	7,7%	56,4%	
	4	11	2	0	13	84,6%	63,5%	
	5	3	10	0	13	23,1%	55,4%	
	6	11	2	0	13	84,6%	60,3%	
2	1	12	1	0	13	92,3%	92,3%	84,4%
	2	13	0	0	13	100,0%	96,2%	
	3	7	5	1	13	53,8%	82,1%	
	4	12	1	0	13	92,3%	84,6%	
	5	5	8	0	13	38,5%	75,4%	
	6	10	3	0	13	76,9%	75,6%	
3	1	13	0	0	13	100,0%	100,0%	92,4%
	2	12	1	0	13	92,3%	96,2%	
	3	11	2	0	13	84,6%	92,3%	
	4	12	1	0	13	92,3%	92,3%	
	5	10	3	0	13	76,9%	89,2%	
	6	8	5	0	13	61,5%	84,6%	
	TOTALES	172	61	1				

Para la primer consulta (ver Tabla 19) en su primera iteración la precisión oscila entre 55,4% y 92,3%, obteniendo una precisión media de 68,1%, para la segunda

iteración la precisión oscila entre 75,4% y 96,2%, obteniendo una precisión media de 84,4% y finalmente para la tercera iteración el valor de la precisión oscila entre 84,6% y 100,0%, obteniendo una precisión media de 92,4%, lo que refleja una mejora de la precisión, de una iteración a otra iteración, a medida que el algoritmo EC-IDF procesa las evaluaciones hechas por los usuarios (ver Figura 54).

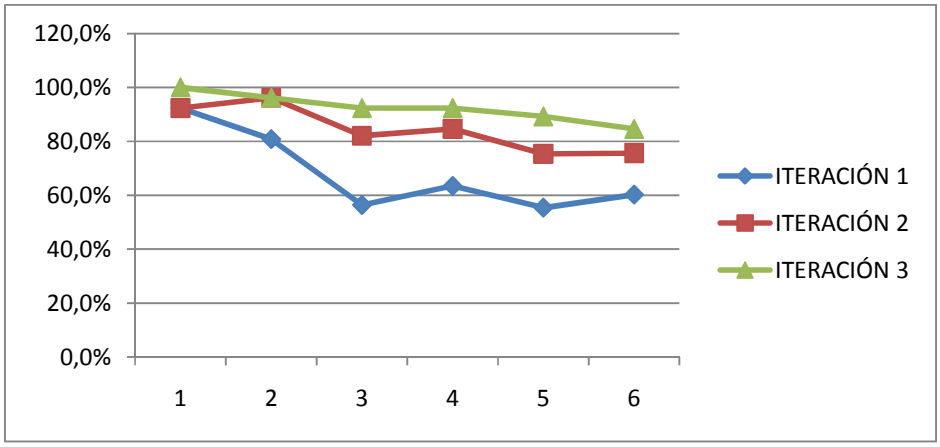


Figura 54. Prueba 2 "Inferencia Fuzzy"

Tabla 20. Prueba 2 "Aplicación de la lógica difusa" – Estadísticas

Aplicación de la lógica difusa								
ITERACIÓN	RESULTADO	R	N	X	TOTAL	EXACTITUD	PRECISIÓN	PRECISIÓN MEDIA
1	1	7	3	2	12	58,3%	58,3%	60,7%
	2	8	4	0	12	66,7%	62,5%	
	3	7	4	1	12	58,3%	61,1%	
	4	8	4	0	12	66,7%	62,5%	
	5	6	6	0	12	50,0%	60,0%	
	6	7	5	0	12	58,3%	59,7%	
2	1	8	3	0	11	72,7%	72,7%	71,7%
	2	7	4	0	11	63,6%	68,2%	
	3	8	3	0	11	72,7%	69,7%	
	4	9	1	1	11	81,8%	72,7%	
	5	8	2	1	11	72,7%	72,7%	

	6	9	2	0	11	81,8%	74,2%	
3	1	10	1	0	11	90,9%	90,9%	89,9%
	2	10	1	0	11	90,9%	90,9%	
	3	10	1	0	11	90,9%	90,9%	
	4	10	0	1	11	90,9%	90,9%	
	5	10	0	1	11	90,9%	90,9%	
	6	6	5	0	11	54,5%	84,8%	
	TOTALES	148	49	7				

Para la segunda consulta (ver Tabla 20) en su primera iteración la precisión oscila entre 58,3% y 62,5%, obteniendo una precisión media de 60,7%, para la segunda iteración la precisión oscila entre 72,7% y 74,2%, obteniendo una precisión media de 71,7% y finalmente para la tercera iteración el valor de la precisión oscila entre 84,8% y 90,9%, obteniendo una precisión media de 89,9%, lo que refleja una mejora de la precisión de una iteración a otra iteración, a medida que el algoritmo EC-IDF procesa las evaluaciones hechas por los usuarios (ver Figura 55).

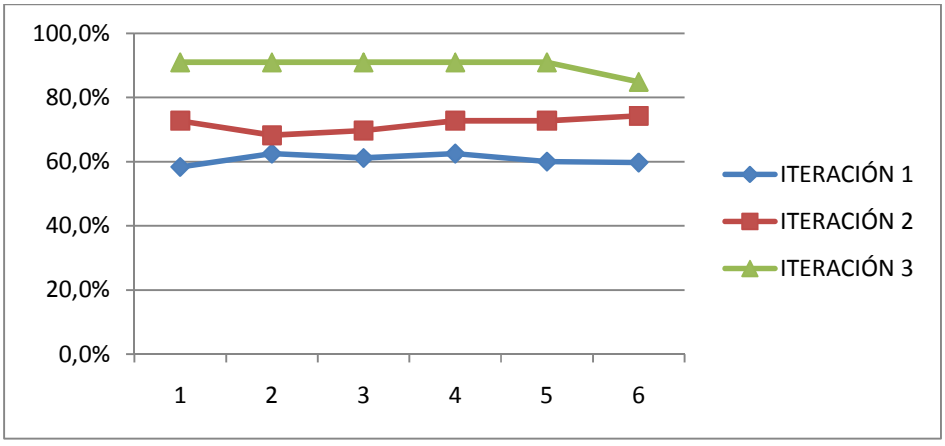


Figura 55. Prueba 2 "Aplicación de la lógica difusa"

Tabla 21. Prueba 2 "Sistemas expertos" - Estadísticas

Sistemas expertos								
ITERACIÓN	RESULTADO	R	N	X	TOTAL	EXACTITUD	PRECISIÓN	PRECISIÓN MEDIA
1	1	8	3	0	11	72,7%	72,7%	74,3%
	2	9	2	0	11	81,8%	77,3%	
	3	7	4	0	11	63,6%	72,7%	
	4	8	3	0	11	72,7%	72,7%	
	5	9	1	1	11	81,8%	74,5%	
	6	9	0	2	11	81,8%	75,8%	
2	1	10	1	0	11	90,9%	90,9%	81,9%
	2	10	1	0	11	90,9%	90,9%	
	3	6	5	0	11	54,5%	78,8%	
	4	8	3	0	11	72,7%	77,3%	
	5	8	3	0	11	72,7%	76,4%	
	6	9	2	0	11	81,8%	77,3%	
3	1	9	1	1	11	81,8%	81,8%	84,0%
	2	11	0	0	11	100,0%	90,9%	
	3	8	2	1	11	72,7%	84,8%	
	4	9	2	0	11	81,8%	84,1%	
	5	8	3	0	11	72,7%	81,8%	
	6	8	2	1	11	72,7%	80,3%	
	TOTALES	154	38	6				

Para la tercer consulta (ver Tabla 21) en su primera iteración la precisión oscila entre 72,7% y 75,8%, obteniendo una precisión media de 74,3%, para la segunda iteración la precisión oscila entre 76,4% y 90,9%, obteniendo una precisión media de 81,9% y finalmente para la tercera iteración el valor de la precisión oscila entre 80,3% y 90,9%, obteniendo una precisión media de 84,0%, lo que refleja una mejora de la precisión de una iteración a otra iteración, a medida que el algoritmo EC-IDF procesa las evaluaciones hechas por los usuarios (ver Figura 56).

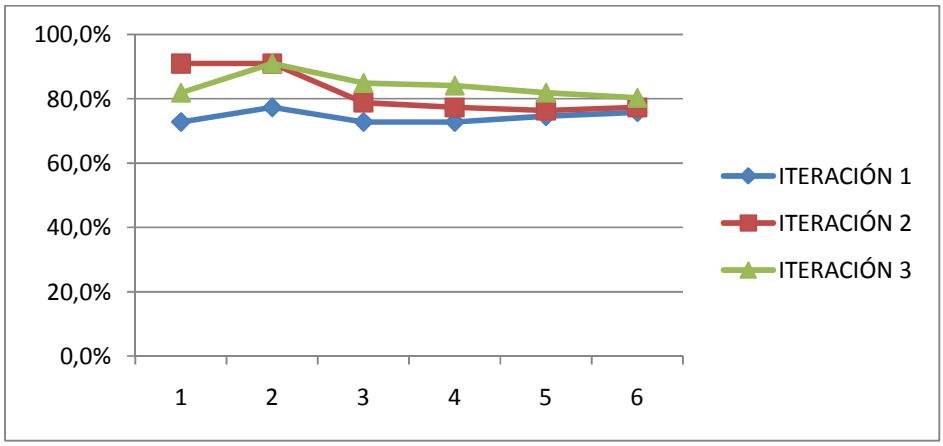


Figura 56. Prueba 2 "Sistemas expertos"

A continuación se presentan los resultados de los cálculos realizados para medir el nivel de concordancia entre los usuarios con respecto a las evaluaciones hechas sobre cada uno de los documentos recuperados en la primera iteración para la primera consulta, ya que para las otras iteraciones los documentos recuperados no son los mismos (ver Tabla 22).

Tabla 22. Prueba 2 – Kappa de Fleiss

Inferencia Fuzzy						
ITERACIÓN	RESULTADO	R	N	X	TOTAL	Pi
1	1	12	1	0	13	0,85
	2	9	4	0	13	0,54
	3	1	12	0	13	0,85
	4	11	2	0	13	0,72
	5	3	10	0	13	0,62
	6	11	2	0	13	0,72
	TOTAL	47	31	0	78	4,28
	Pr	0,60	0,40	0,00		0,71 P-
	Pr^2	0,36	0,16	0,00	0,52	Pe-

Kappa de Fleiss: 0,402 Fair Concordance

El valor de Kappa de Fleiss es de 0,402 lo que quiere decir que hubo un acuerdo razonable entre los usuarios al evaluar los documentos (ver Tabla 1).

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.

Tabla 23. Prueba 2 – Las 3 consultas – Estadísticas

	CONSULTA	1	2	3		
ITERACIÓN	RESULTADOS	PRECISIÓN	PRECISIÓN	PRECISIÓN	PRECISIÓN TOTAL	PRECISIÓN TOTAL MEDIA
1	1	92,3%	58,3%	72,7%	74,5%	67,7%
	2	80,8%	62,5%	77,3%	73,5%	
	3	56,4%	61,1%	72,7%	63,4%	
	4	63,5%	62,5%	72,7%	66,2%	
	5	55,4%	60,0%	74,5%	63,3%	
	6	60,3%	59,7%	75,8%	65,2%	
2	1	92,3%	72,7%	90,9%	85,3%	79,3%
	2	96,2%	68,2%	90,9%	85,1%	
	3	82,1%	69,7%	78,8%	76,8%	
	4	84,6%	72,7%	77,3%	78,2%	
	5	75,4%	72,7%	76,4%	74,8%	
	6	75,6%	74,2%	77,3%	75,7%	
3	1	100,0%	90,9%	81,8%	90,9%	88,8%
	2	96,2%	90,9%	90,9%	92,7%	
	3	92,3%	90,9%	84,8%	89,4%	
	4	92,3%	90,9%	84,1%	89,1%	
	5	89,2%	90,9%	81,8%	87,3%	
	6	84,6%	84,8%	80,3%	83,3%	

Para las tres consultas en la primera iteración (ver Tabla 23) la precisión oscila entre 63,3% y 74,5%, obteniendo una precisión media de 67,7%, para la segunda iteración la precisión oscila entre 74,8% y 85,3%, obteniendo una precisión media de 79,3% y finalmente para la tercera iteración el valor de la precisión oscila entre 83,3% y 92,7%, obteniendo una precisión media de 88,8%, lo que refleja una mejora de la precisión de una iteración a otra iteración a medida que el algoritmo EC-IDF procesa las evaluaciones hechas por los usuarios (ver Figura 57).

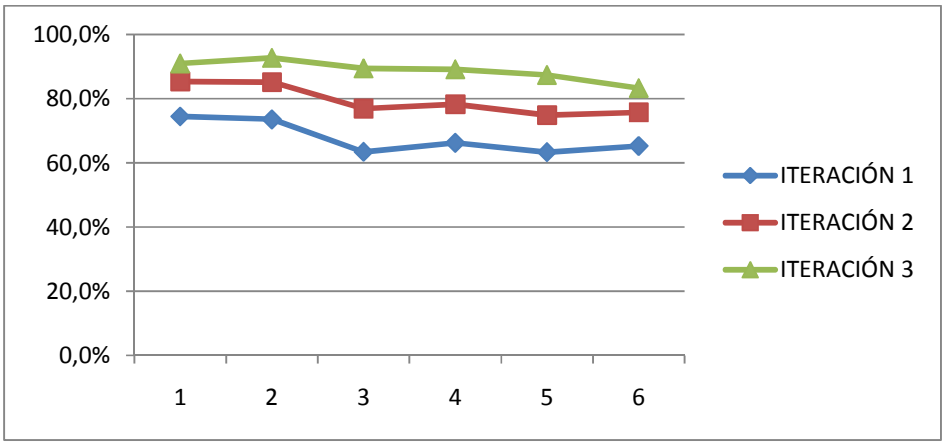


Figura 57. Prueba 2 totales – 3 consultas

En general la Figura 57 muestra como desde la segunda iteración la precisión mejora considerablemente, lo que quiere decir que desde que el algoritmo CE-IDF empieza a recibir las evaluaciones dadas por los usuarios inmediatamente muestra una mejora considerable y al aumentar estas iteraciones sigue mejorando.

4.13.3 PRUEBA 3

Se realizó una tercera prueba con veintiún (21) estudiantes de Ingeniería de Sistemas de diferentes niveles de la universidad del Cauca pertenecientes a un curso ofrecido por la célula .net, orientado por el estudiante de Ingeniería de Sistemas Henry Muñoz. Esta vez las consultas fueron las siguientes:

1. “Método get y método post”
2. “Variables de aplicación”
3. “Tipos de autenticación en ASP.NET”

La Tabla 24, la Tabla 25 y la Tabla 26 muestran las evaluaciones que dieron los usuarios a cada una de las 3 consultas.

Tabla 24. Prueba 3 "Método get y método post"– Resultados de la consulta

		Método get y método post																			
ITERACION	RESULTADO	U1	U2	U3	U4	U5	U6	U8	U9	U10	U11	U12	U13	U14	U15	U16	U27	U18	U19	U20	U21
1	1	N	X	R	X	X	R	R	R	X	X	X	N	R	X	X	N	N	X	R	X
	2	N	R	R	R	N	N	N	R	R	R	R	R	N	R	X	R	N	R	X	R
	3	X	X	R	N	X	R	X	R	X	X	X	N	N	R	N	N	X	R	N	X
	4	R	N	N	N	R	N	R	R	R	N	N	X	R	N	N	N	N	R	R	R
	5	X	R	R	R	R	R	R	N	R	X	R	R	N	R	R	N	R	R	R	R
	6	R	N	N	N	N	R	R	N	N	N	N	R	R	R	X	R	N	N	N	N
2	1	R	R	R	R	R	R	X	R	X		R	N	R	R	R	R	R	N	N	X
	2	X	R	R	R	R	R	R	R	R		R	R	R	R	R	N	X	R	R	R
	3	R	R	R	R	N	R	R	R	X		X	N	N	R	N	R	N	R	R	X
	4	R	R	R	R	R	R	R	R	R		X	R	R	R	N	N	N	R	N	R
	5	R	N	R	R	R	R	R	N	R		R	R	R	R	N	N	N	R	R	R
	6	N	R	R	R	N	N	R	N	N		N	R	R	N	N	N	N	R	R	N
3	1	R	R	R	R	R	R	X	R	X		R		R	R	R					
	2	R	R	R	R	R	R	R	R	R		R		R	R	R					
	3	N	R	R	R	R	R	N	R	X		R		R	R	N					
	4	R	R	R	R	R	R	R	R	R		R		R	R	N					
	5	R	R	R	R	R	R	R	N	R		R		R	R	R					
	6	N	R	R	N	R	R	R	N	N		R		N	N	R					

Para esta consulta se excluyó al usuario 7 (U7) debido a que desde la primera iteración dejo documentos sin evaluar lo que afecta enormemente los resultados de la prueba (ver Tabla 24).

Tabla 25. Prueba 3 "Variables de aplicación" – Resultados de la consulta

		Variables de aplicación																				
ITERACIÓN	RESULTADO	U1	U2	U3	U4	U5	U6	U7	U8	U9	U10	U11	U12	U13	U14	U15	U16	U17	U18	U19	U20	U21
1	1	N	R	N	N	N	R	R	R	R	R	N	N	R	N	R	N	R	N	N	R	R
	2	R	R	N	R	R	N	R	R	R	R	N	R	R	R	R	R	N	N	N	N	R
	3	R	N	R	N	R	N	R	R	N	R	R	N	R	R	N	N	R	R	R	R	R
	4	N	R	N	N	R	R	R	R	R	X	N	R	R	R	R	R	R	N	R	N	X
	5	N	N	N	N	N	N	R	R	R	N	R	N	R	N	X	R	R	R	X	R	R
	6	R	R	N	N	R	N	R	R	R	N	X	R	R	N	N	N	N	N	R	N	N
2	1	R	R	R	R	R	R	R	R	R	R	R	R	N	R	R	R		N	N	R	
	2	R	R	R	N	R	R	R	R	R	R	N	R	R	N	R	R		R	N	R	
	3	R	R	R	N	R	R	N	R	N	R	N	R	R	R	N	R		R	R	R	
	4	R	R	N	N	N	R	R	R	R	X	N	N	R	R	R	R		N	R	R	
	5	N	R	R	N	N	R	R	R	R	N	N	N	R	R	X	X		N	X	R	
	6	X	R	N	N	R	N	N	R	R	N	R	N	N	R	N	X		N	N	X	
3	1	R	R	R	R	R	R	R	R		R	R	R	R	R		R				R	
	2	R	R	R	N	R	R	R	R		R	X	N	R	R		R				R	
	3	R	R	N	N	R	R	R	R		R	N	R	R	N		R				R	
	4	R	R	R	R	R	R	R	R		X	R	N	R	R		R				R	
	5	R	R	N	R	R	R	R	R		R	R	R	R	R		R				R	
	6	N	R	R	N	N	N	R	R		N	X	N	N	R		N				R	

Tabla 26. Prueba 3 "Tipos de autenticación en ASP.NET" – Resultados de la consulta

		Tipos de autenticación en ASP.NET																				
ITERACIÓN	RESULTADO	U1	U2	U3	U4	U5	U6	U7	U8	U9	U10	U11	U12	U13	U14	U15	U16	U17	U18	U19	U20	U21
1	1	N	R	R	R	N	N	R	N	N	N	N	N	R	R	N	R	R	R	R	R	N
	2	R	R	N	N	N	N	N	R	N	R	N	R	R	R	R	N	N	N	R	R	N
	3	R	N	R	R	R	R	R	R	N	N	X	N	R	R	N	N	N	N	N	R	R
	4	N	N	R	N	R	N	R	R	R	N	R	N	R	N	N	N	R	R	R	R	N
	5	R	R	R	R	N	N	R	N	R	N	R	N	N	R	R	N	N	N	R	R	N
	6	X	N	R	N	X	R	R	R	N	N	R	N	R	N	R	N	X	R	X	N	R
2	1	R	R	R	R	N	R	R	N	N	N	N	R	R	N	X	N	R		R	R	
	2	R	R	R	R	R	R	R	R	N	R	R	N	R	R	N	R	N		R	R	

	3	R	R	R	R	N	N	N	R	R	N	R	R	R	R	X	N	R		N	R	
	4	N	N	R	R	N	R	X	R	N	N	N	N	R	N	R	N	N		R	R	
	5	R	R	R	N	R	N	N	R	R	N	R	N	N	R	N	N	N		R	R	
	6	R	N	R	N	N	N	R	R	N	N	R	R	R	R	R	N	X		X	R	
3	1	R	R	R	R	R	R	X		N		N	R	R	R	N	R	R			R	
	2	R	R	R	R	R	R	R		N		R	R	R	R	R	N	R			R	
	3	R	R	R	R	R	R	N		N		R	R	R	R	N	R	R			R	
	4	R	N	R	R	N	R	R		R		R	N	R	R	R	N	R			R	
	5	R	R	R	R	N	N	N		N		R	R	R	N	R	N	N			R	
	6	X	R	R	N	X	R	R		R		R	N	R	N	R	N	N			N	

Tomando los resultados de la Tabla 16 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 lo muestra la Tabla 27.

Este mismo proceso se hizo para la segunda y tercera consulta, como resultado se obtuvieron la Tabla 25 y la Tabla 26 respectivamente.

Tabla 27. Prueba 3 "Método get y método post" - Estadísticas

Método get y método post								
ITERACIÓN	RESULTADO	R	N	X	TOTAL	EXACTITUD	PRECISIÓN	PRECISIÓN MEDIA
1	1	6	4	10	20	30,0%	30,0%	40,9%
	2	12	6	2	20	60,0%	45,0%	
	3	5	6	9	20	25,0%	38,3%	
	4	9	10	1	20	45,0%	40,0%	
	5	15	3	2	20	75,0%	47,0%	
	6	7	12	1	20	35,0%	45,0%	
2	1	13	3	3	19	68,4%	68,4%	70,7%
	2	16	1	2	19	84,2%	76,3%	
	3	11	5	3	19	57,9%	70,2%	
	4	14	4	1	19	73,7%	71,1%	
	5	14	5	0	19	73,7%	71,6%	
	6	8	11	0	19	42,1%	66,7%	

3	1	11	0	2	13	84,6%	84,6%	86,3%
	2	13	0	0	13	100,0%	92,3%	
	3	9	3	1	13	69,2%	84,6%	
	4	12	1	0	13	92,3%	86,5%	
	5	12	1	0	13	92,3%	87,7%	
	6	7	6	0	13	53,8%	82,1%	
	TOTALES	194	81	37				

Para la primera consulta (ver Tabla 19) en su primera iteración la precisión oscila entre 30,0% y 47,0%, obteniendo una precisión media de 40,9%, para la segunda iteración la precisión oscila entre 66,7% y 76,3%, obteniendo una precisión media de 70,7% y finalmente para la tercera iteración el valor de la precisión oscila entre 82,1% y 92,3%, obteniendo una precisión media de 86,3%, lo que refleja una mejora de la precisión de una iteración a otra iteración, a medida que el algoritmo EC-IDF procesa las evaluaciones hechas por los usuarios (ver Figura 58).

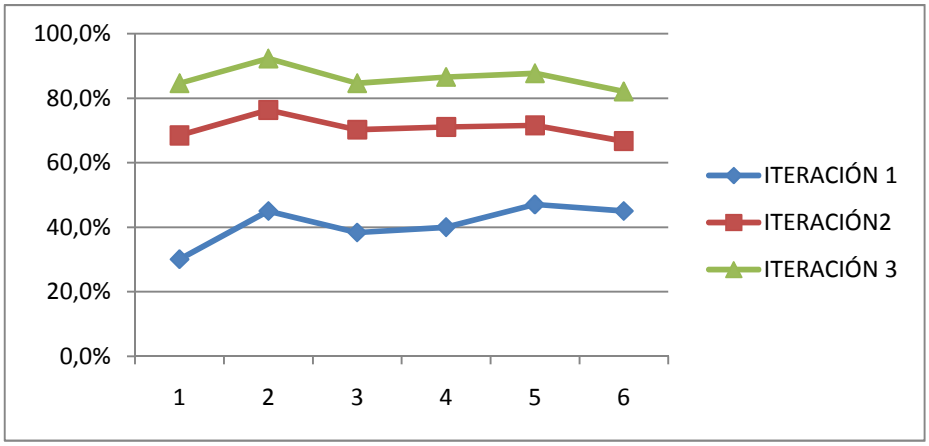


Figura 58. Prueba 3 "Método get y método post"

Tabla 28. Prueba 3 "Variables de aplicación" - Estadísticas

Variables de aplicación								
ITERACIÓN	RESULTADO	R	N	X	TOTAL	EXACTITUD	PRECISIÓN	PRECISIÓN MEDIA
1	1	11	10	0	21	52,4%	52,4%	58,5%
	2	14	7	0	21	66,7%	59,5%	
	3	14	7	0	21	66,7%	61,9%	
	4	13	6	2	21	61,9%	61,9%	
	5	10	9	2	21	47,6%	59,0%	
	6	9	11	1	21	42,9%	56,3%	
2	1	16	3	0	19	84,2%	84,2%	75,4%
	2	15	4	0	19	78,9%	81,6%	
	3	14	5	0	19	73,7%	78,9%	
	4	12	6	1	19	63,2%	75,0%	
	5	9	7	3	19	47,4%	69,5%	
	6	6	10	3	19	31,6%	63,2%	
3	1	15	0	0	15	100,0%	100,0%	87,5%
	2	12	2	1	15	80,0%	90,0%	
	3	11	4	0	15	73,3%	84,4%	
	4	13	1	1	15	86,7%	85,0%	
	5	14	1	0	15	93,3%	86,7%	
	6	6	8	1	15	40,0%	78,9%	
	TOTALES	214	101	15				

Para la segunda consulta (ver Tabla 28) en su primera iteración la precisión oscila entre 52,4% y 61,9%, obteniendo una precisión media de 58,5%, para la segunda iteración la precisión oscila entre 63,2% y 84,2%, obteniendo una precisión media de 75,4% y finalmente para la tercera iteración el valor de la precisión oscila entre 78,9% y 100,0%, obteniendo una precisión media de 87,5%, lo que refleja una mejora de la precisión de una iteración a otra iteración a medida que el algoritmo EC-IDF procesa las evaluaciones hechas por los usuarios (ver Figura 59).

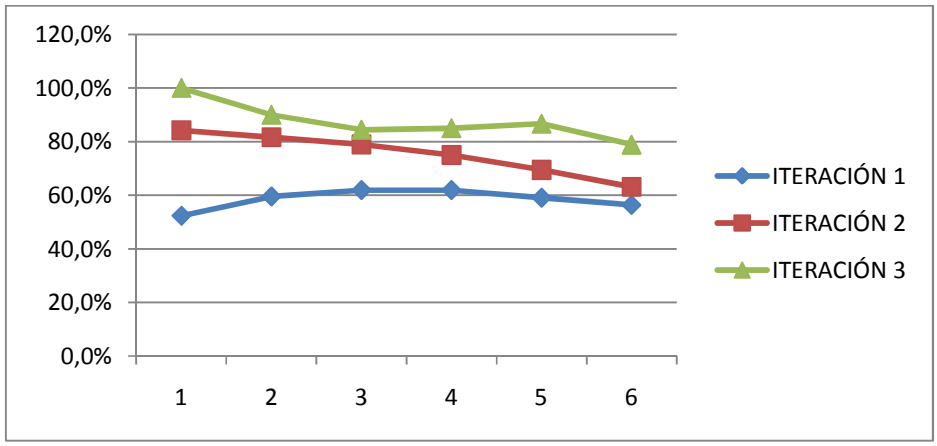


Figura 59. Prueba 3 "Variables de aplicación"

Tabla 29. Prueba 3 "Tipos de autenticación en ASP.NET" - Estadísticas

Tipos de autenticación ASP .NET								
ITERACIÓN	RESULTADO	R	N	X	TOTAL	EXACTITUD	PRECISIÓN	PRECISIÓN MEDIA
1	1	11	10	0	21	52,4%	52,4%	51,0%
	2	10	11	0	21	47,6%	50,0%	
	3	11	9	1	21	52,4%	50,8%	
	4	11	10	0	21	52,4%	51,2%	
	5	11	10	0	21	52,4%	51,4%	
	6	9	8	4	21	42,9%	50,0%	
2	1	11	7	1	19	57,9%	57,9%	61,7%
	2	15	4	0	19	78,9%	68,4%	
	3	12	6	1	19	63,2%	66,7%	
	4	8	10	1	19	42,1%	60,5%	
	5	10	9	0	19	52,6%	58,9%	
	6	10	7	2	19	52,6%	57,9%	
3	1	12	3	1	16	75,0%	75,0%	77,2%
	2	14	2	0	16	87,5%	81,3%	
	3	13	3	0	16	81,3%	81,3%	
	4	12	4	0	16	75,0%	79,7%	
	5	9	7	0	16	56,3%	75,0%	
	6	8	6	2	16	50,0%	70,8%	
	TOTALES	197	126	13				

Para la tercera consulta (ver Tabla 29) en su primera iteración la precisión oscila entre 50,0% y 52,4%, obteniendo una precisión media de 51,0%, para la segunda iteración la precisión oscila entre 57,9% y 68,4%, obteniendo una precisión media de 61,7% y finalmente para la tercera iteración el valor de la precisión oscila entre 70,8% y 81,3%, obteniendo una precisión media de 77,2%, lo que refleja una mejora de la precisión de una iteración a otra iteración, a medida que el algoritmo EC-IDF procesa las evaluaciones hechas por los usuarios (ver Figura 60).

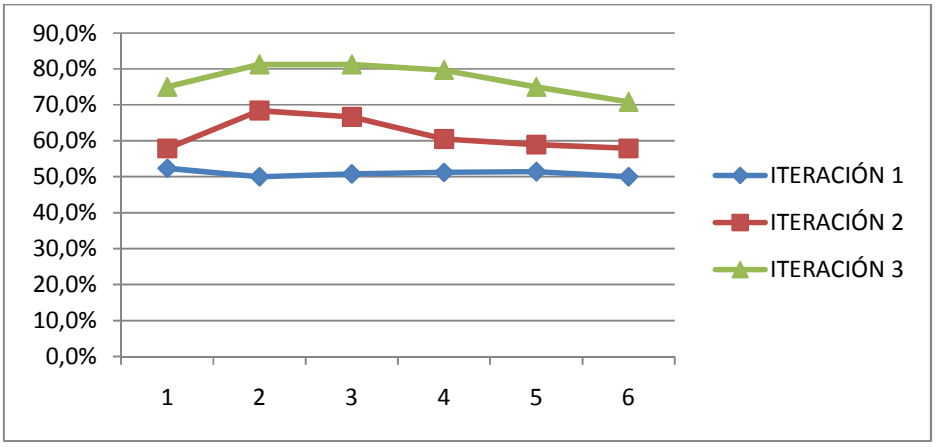


Figura 60. Prueba 3 "Tipos de autenticación en ASP.NET"

A continuación se presentan los resultados de cálculos realizados para medir el nivel de concordancia de usuarios con respecto a las evaluaciones hechas sobre cada uno de los documentos recuperados en la primera iteración para la primera consulta, ya que para las otras iteraciones los documentos recuperados no son los mismos (ver Tabla 30).

Tabla 30. Prueba 3 – Kappa de Fleiss

Método get y método post						
ITERACIÓN	RESULTADO	R	N	X	TOTAL	Pi
1	1	6	4	10	20	0,35
	2	12	6	2	20	0,43
	3	5	6	9	20	0,32
	4	9	10	1	20	0,43
	5	15	3	2	20	0,57
	6	7	12	1	20	0,46
	TOTAL	54	41	25	120	2,56
	Pr	0,45	0,34	0,21		0,43 P-
	Pr^2	0,20	0,12	0,04	0,36	Pe-

Kappa de Fleiss:	0,0999071	Slight Concordance

El valor de Kappa de Fleiss es de 0,0999071 lo que quiere decir que hubo un acuerdo ligero entre los usuarios al evaluar los documentos (ver Tabla 1).

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 31.

Tabla 31. Prueba 3 Las 3 consultas - Estadísticas

	CONSULTA	1	2	3		
ITERACIÓN	RESULTADOS	PRECISIÓN	PRECISIÓN	PRECISIÓN	PRECISIÓN TOTAL	PRECISIÓN TOTAL MEDIA
1	1	30,0%	52,4%	52,4%	44,9%	50,1%
	2	45,0%	59,5%	50,0%	51,5%	
	3	38,3%	61,9%	50,8%	50,3%	
	4	40,0%	61,9%	51,2%	51,0%	
	5	47,0%	59,0%	51,4%	52,5%	
	6	45,0%	56,3%	50,0%	50,4%	
2	1	68,4%	84,2%	57,9%	70,2%	69,3%
	2	76,3%	81,6%	68,4%	75,4%	
	3	70,2%	78,9%	66,7%	71,9%	
	4	71,1%	75,0%	60,5%	68,9%	
	5	71,6%	69,5%	58,9%	66,7%	
	6	66,7%	63,2%	57,9%	62,6%	
3	1	84,6%	100,0%	75,0%	86,5%	83,7%

2	92,3%	90,0%	81,3%	87,9%	
3	84,6%	84,4%	81,3%	83,4%	
4	86,5%	85,0%	79,7%	83,7%	
5	87,7%	86,7%	75,0%	83,1%	
6	82,1%	78,9%	70,8%	77,3%	

Para las 3 consultas en la primera iteración (ver Tabla 31) la precisión oscila entre 44,9% y 52,5%, obteniendo una precisión media de 50,1%, para la segunda iteración la precisión oscila entre 62,6% y 75,4%, obteniendo una precisión media de 69,3% y finalmente para la tercera iteración el valor de la precisión oscila entre 77,3% y 87,9%, obteniendo una precisión media de 83,7%, lo que refleja una mejora de la precisión de una iteración a otra iteración, a medida que el algoritmo EC-IDF procesa las evaluaciones hechas por los usuarios (ver Figura 61).

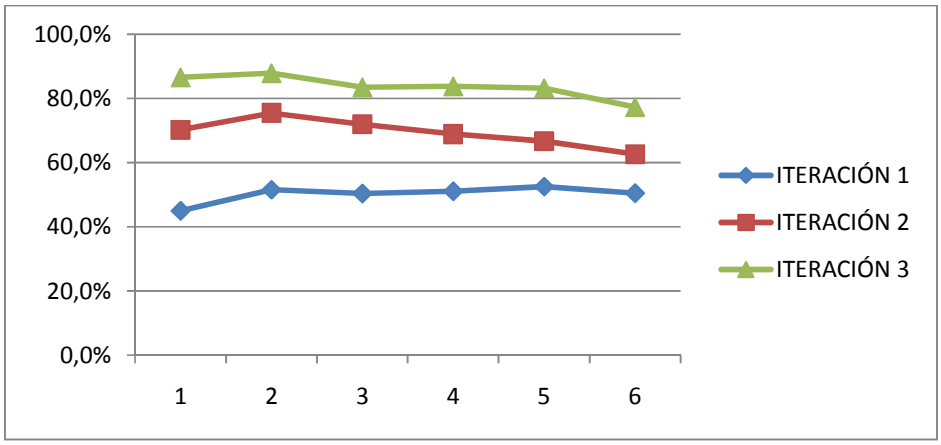


Figura 61. Prueba 3 totales – 3 consultas

En general la Figura 61 muestra como desde la segunda iteración la precisión mejora considerablemente, lo que quiere decir que desde que el algoritmo CE-IDF empieza a recibir las evaluaciones dadas por los usuarios inmediatamente muestra una mejora considerable y al aumentar estas iteraciones sigue mejorando.

Reuniendo los datos obtenidos de las 3 pruebas se hicieron cálculos globales cuyos resultados se presentan en la Tabla 32.

Tabla 32. Resultados – Las 3 pruebas – Estadísticas

	PRUEBA	1	2	3		
ITERACIÓN	RESULTADOS	PRECISIÓN	PRECISIÓN	PRECISIÓN	PRECISIÓN TOTAL	PRECISIÓN TOTAL MEDIA
1	1	67,2%	74,5%	44,9%	62,2%	55,6%
	2	51,8%	73,5%	51,5%	58,9%	
	3	40,5%	63,4%	50,3%	51,4%	
	4	43,3%	66,2%	51,0%	53,5%	
	5	45,8%	63,3%	52,5%	53,9%	
	6	45,7%	65,2%	50,4%	53,8%	
2	1	95,8%	85,3%	70,2%	83,8%	76,6%
	2	92,9%	85,1%	75,4%	84,5%	
	3	79,6%	76,8%	71,9%	76,1%	
	4	78,6%	78,2%	68,9%	75,2%	
	5	72,2%	74,8%	66,7%	71,2%	
	6	67,6%	75,7%	62,6%	68,6%	
3	1	100,0%	90,9%	86,5%	92,5%	86,5%
	2	94,0%	92,7%	87,9%	91,5%	
	3	92,8%	89,4%	83,4%	88,5%	
	4	85,8%	89,1%	83,7%	86,2%	
	5	77,3%	87,3%	83,1%	82,6%	
	6	72,9%	83,3%	77,3%	77,8%	

Para las 3 pruebas en la primera iteración (ver Tabla 32) la precisión oscila entre 51,4% y 62,2%, obteniendo una precisión media de 55,6%, para la segunda iteración la precisión oscila entre 68,6% y 84,5%, obteniendo una precisión media de 76,6% y finalmente para la tercera iteración el valor de la precisión oscila entre 77,8% y 92,5%, obteniendo una precisión media de 86,5%, lo que refleja una mejora de la precisión de una iteración a otra iteración, a medida que el algoritmo EC-IDF procesa las evaluaciones hechas por los usuarios (ver Figura 61).

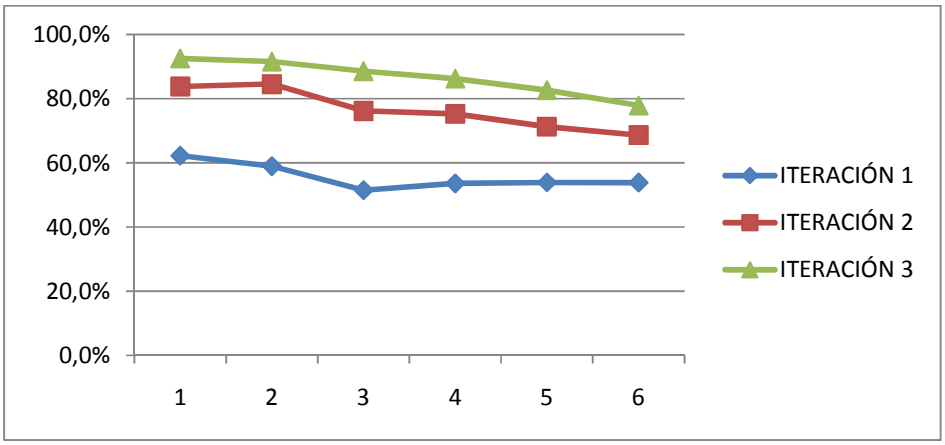


Figura 62. Resultados – Las 3 pruebas

La Figura 62 muestra como desde la segunda iteración la precisión en todas las pruebas mejora considerablemente, lo que quiere decir que desde que el algoritmo CE-IDF empieza a recibir las evaluaciones dadas por los usuarios inmediatamente muestra una mejora considerable y al aumentar estas iteraciones sigue mejorando.

En general los resultados de las pruebas hechas con los usuarios son muy buenos, se ve claramente la mejoría de la precisión en los documentos recuperados iteración tras iteración al realizar la expansión de consulta con el algoritmo CE-IDF, aunque los valores obtenidos en el índice kappa de Fleiss sugieren que las pruebas realizadas no son del todo contundentes y que los resultados son preliminares, ya que los valores de concordancia no son los mejores, pero aun así muestran cual es el comportamiento del algoritmo y los buenos resultados que logra obtener.

5. CONCLUSIONES Y TRABAJO FUTURO

Este capítulo presenta las conclusiones obtenidas en la realización del proyecto de investigación presentado en esta monografía, además se presentan algunas sugerencias para futuros trabajos de investigación en el área.

5.1 CONCLUSIONES

1. Se presentó una nueva función de la importancia relativa de un término en una colección de documentos (IDF). Esta función es continua, está en el rango de 0 a 1 incluidos y refleja la prevalencia de los términos que aparecen mayoritariamente en documentos relevantes.
2. Con base en la función presentada para el cálculo del valor IDF de un término, se presentaron dos algoritmos para la expansión de consulta, el primero denominado VT-IDF que al igual que Rocchio representa una consulta como un vector con términos y sus respectivos pesos para ubicar en el espacio

multidimensional de términos por documentos. El segundo algoritmo presentado se denomina CE-IDF que a diferencia de VT-IDF y Rocchio agrega a la consulta original los términos más relevantes del perfil del usuario, entregando como resultado una lista de términos en una cadena de texto similar a la digitada por el usuario.

3. La evaluación de VT-IDF y CE-IDF se realizó frente a una consulta "básica", es decir, una búsqueda por similitud de cosenos sin proceso de expansión, y el algoritmo de Rocchio de expansión de consulta. Los experimentos se realizaron con dos colecciones de datos reconocidas por la comunidad de I+D en recuperación de información, como lo son CACM y LISA. Para cada uno de los experimentos se tuvieron en cuenta tres escenarios: sin memoria, con memoria de sesión y con memoria a largo plazo. Los resultados en los dos primeros escenarios favorecieron a VT-IDF y dejaron en segundo lugar a CE-IDF con una variación estadísticamente insignificante. Pero en el tercer escenario, el más importante, el algoritmo CE-IDF obtuvo los mejores resultados, demostrando que es un algoritmo que se adapta más rápidamente a los cambios en los requerimientos de los usuarios de búsqueda, una situación muy común en los usuarios de los buscadores de Internet y de los sistemas de recuperación de información en general.
4. El algoritmo CE-IDF muestra ser un algoritmo muy completo y sencillo de implementar, ya que no solo obtuvo muy buenos resultados para los experimentos a corto plazo, con memoria de sesión y a largo plazo (donde mostro una gran superioridad frente al algoritmo de Rocchio), sino que además no tiene la gran desventaja de requerir el afinamiento de parámetros, es decir, no presenta el inconveniente de tener que experimentar con parámetros de ajuste para obtener los mejores resultados de precisión como si ocurre con Rocchio, en cuyo caso, para lograr obtener los mejores resultados en cada uno de los experimentos se debían cambiar estos valores.

5. Se presentó ECWEB, un meta buscador web, sencillo e intuitivo, con una interfaz gráfica de usuario amigable y flexible a las necesidades del usuario, que permite mediante una fácil interacción del usuario con el sistema, aprovechar de forma personal las ventajas que brinda el algoritmo CE-IDF como método de expansión de consulta que mejora la relevancia de los documentos recuperados.
6. Mediante ECWEB se realizaron pruebas con usuarios que apoyaron los resultados obtenidos en los experimentos de laboratorio para la sesión a largo plazo, como parte de cada una de las pruebas se midió el grado de concordancia de los usuarios en cuanto a las evaluaciones dadas, por medio del índice Kappa de Fleiss, y aunque los valores de concordancia entre los jueces no son los mejores (razón por la cual se deben realizar más experimentos), aun así se obtuvieron excelentes resultados que muestran lo rápido que el algoritmo CE- IDF se adaptó a las necesidades de cada uno de los usuarios.

5.2 TRABAJO FUTURO

1. Como trabajo futuro, el grupo de investigación espera evaluar los algoritmos propuestos con otras colecciones comúnmente usadas en recuperación de información, como por ejemplo: TREC, ISI, NPL, TIME, MED [3].

2. Realizar una propuesta de los algoritmos, incluyendo la capacidad de analizar semánticamente los términos, con el objetivo de gestionar conceptos en lugar de términos, a través de ontologías (por ejemplo: WordNet), diccionarios o tesauros de dominio general.

6. GLOSARIO Y BIBLIOGRAFIA

6.1 GLOSARIO

SIR: Proceso que accede a información previamente almacenada, mediante herramientas informáticas que permiten establecer ecuaciones de búsqueda específicas.

Lucene: Es un API de código abierto para recuperación de información, originalmente implementada en Java por Doug Cutting. Está apoyado por el Apache Software Foundation y se distribuye bajo la Apache Software License. Esta API tiene versiones para otros lenguajes incluyendo C#, es útil para cualquier aplicación que requiera indexado y búsqueda de texto completo. Lucene ha sido ampliamente usado por su utilidad en la implementación de motores de búsqueda.

URF: User Relevance Feedback

ARF: Automatic Relevance Feedback

Precisión: corresponde a la fracción de los documentos recuperados por el sistema que realmente son relevantes para el usuario.

Recuerdo: Corresponde a la fracción de los documentos relevantes que han sido recuperados por el sistema del total de documentos relevantes.

KAPPA: Calcula el porcentaje de acuerdo entre evaluadores.

CACM: Communications of the ACM

LISA: Library & Information Science Abstracts

Meta buscador: Sistema que hace uso de los motores de búsqueda más utilizados como Yahoo!, Google, Bing, entre otros, para encontrar información. Estos buscadores carecen de base de datos propia y, en su lugar, aprovecha las de otros buscadores y muestra una combinación de las mejores páginas que ha devuelto cada buscador.

6.2 BIBLIOGRAFÍA

[1] P. M. Alberto Ruiz, Ana García-Serrano. Análisis y expansión de consultas en lenguaje natural para mejora de la búsqueda en Web. Available: http://www.dia.fi.upm.es/~agarcia/publications/archivos/DMO6.pdf

[2] C. d. Wikipedia, "Expansión de consultas," *Wikipedia, La enciclopedia libre,* 2010.

[3] R. Baeza-Yates, A. and B. Ribeiro-Neto, *Modern Information Retrieval*: Addison-Wesley Longman Publishing Co., Inc., 1999.

[4] C. J. V. Rijsbergen, *Information Retrieval*: Butterworth-Heinemann, 1979.

[5] C. Manning*, et al.* (2008). *Introduction to Information Retrieval*. Available: http://www-csli.stanford.edu/~hinrich/information-retrieval-book.html

[6] L. Yongli*, et al.*, "A Query Expansion Algorithm Based on Phrases Semantic Similarity," presented at the Proceedings of the 2008 International Symposiums on Information Processing, 2008.

[7] W. B. Frakes and R. A. Baeza-Yates, *Information Retrieval Data Structures & Algorithms* Prentice-Hall, 1992.

[8] J. Nielsen. (2004). *When Search Engines Become Answer Engines*. Available: http://www.useit.com/alertbox/20040816.html

[9] A. Spink and J. L. Xu, "Selected results from a large study of Web searching: the Excite study," *Information Research,* vol. 6, 2000.

[10] S. Chakrabarti, "Web Search and Information Retrieval," in *Mining the Web*, ed San Francisco: Morgan Kaufmann, 2003, pp. 45-76.

[11] R. Baeza-Yates*, et al.*, "Web Searching," in *Encyclopedia of Language & Linguistics*, ed Oxford: Elsevier, 2006, pp. 527-538.

[12] K. Hammouda, "Title," unpublished|.

[13] E. Garcia. (2009). *RSJ-PM Tutorial: A Tutorial on the Robertson-Sparck Jones Probabilistic Model for Information Retrieval*. Available: http://www.miislita.com/information-retrieval-tutorial/information-retrieval-probabilistic-model-tutorial.pdf

[14] S. E. Robertson and K. Sparck-Jones, "Relevance weighting of search terms," in *Document retrieval systems*, ed: Taylor Graham Publishing, 1988, pp. 143-160.

[15] E. N. Efthimiadis. (1996). *Query Expansion*. Available: http://faculty.washington.edu/efthimis/pubs/Pubs/qe-arist/QE-arist.html

[16] C. Biancalana and A. Micarelli, "Social Tagging in Query Expansion: A New Way for Personalized Web Search," in *SocialCom-09 the 2009 IEEE International Conference on Social Computing*, Vancouver, Canada, 2009, pp. 1060-1065.

[17] B. Marin*, et al.*, "Toward personalized query expansion," presented at the Proceedings of the Second ACM EuroSys Workshop on Social Network Systems, Nuremberg, Germany, 2009.

[18] Z. Dongsheng and W. Liqing, "Study on Key Techniques of Query Expansion Based on Ontology and Its Application," in *Computational Intelligence and Software Engineering, 2009. CiSE 2009. International Conference on*, 2009, pp. 1-4.

[19] T. C. Nguyen and T. T. Phan, "An Ontology-Based Approach of Query Expansion," in *iiWAS'2007 - The Ninth International Conference on Information Integration and Web-based Applications Services*, Jakarta, Indonesia, 2007.

[20] L. Han and G. Chen, "HQE: A hybrid method for query expansion," *Expert Systems with Applications,* vol. 36, pp. 7985-7991, 2009.

[21] M. Blanco. (2003). *Estudio de buscadores*. Available: http://trevinca.ei.uvigo.es/~pcuesta/sm/practicas/Estudio.pdf

[22] BrightPlanet, "The Deep Web: Surfacing Hidden Value," 2000.

[23] Dogpile.com. (2007). *Different Engines, Different Results: Web Searchers Not Always Finding What They're Looking for Online*. Available: http://www.infospaceinc.com/onlineprod/Overlap-DifferentEnginesDifferentResults.pdf

[24] C. d. Wikipedia. (2010). *Microsoft .NET*. Available: http://es.wikipedia.org/w/index.php?title=Microsoft_.NET&oldid=42882644

[25] C. d. Wikipedia. (2010). *C Sharp*. Available: http://es.wikipedia.org/w/index.php?title=Especial:Citar&page=C_Sharp&id=47680841

[26] C. d. Wikipedia. (2011). *ASP.NET*. Available: http://es.wikipedia.org/w/index.php?title=ASP.NET&oldid=49025192.

[27] C. d. Wikipedia. (2010). *Microsoft SQL Server*. Available: http://es.wikipedia.org/w/index.php?title=Microsoft_SQL_Server&oldid=43415049

[28] M. F. Porter, "An algorithm for suffix stripping," *Program,* vol. 14, pp. 130-137, 1980.

[29] S. Dominich, *The Modern Algebra of Information Retrieval*: Springer-Verlag Berlin Heidelberg, 2008.

[30] M. U. Arevalo, "Introcción al Patrón de Arquitectura por Capas," in *Maria Ugenia Arevalo's Blog*, ed, 2010.

[31] C. d. Wikipedia. (2011). *Programación por capas*. Available: http://es.wikipedia.org/w/index.php?title=Especial:Citar&page=Programaci%C3%B3n_por_capas&id=47966346

7. ANEXOS

Anexo 1: Artículo titulado “Algoritmos de Expansión de Consulta basados en una Nueva Función Discreta de Relevancia”.

El articulo refleja el algoritmo propuesto en la investigación y resultados experimentales del mismo, en el marco de una Revista Nacional Indexada por Colciencias y será publicado a finales del año 2011.