UNIVERSIDAD DEL BÍO-BÍO

FACULTAD DE CIENCIAS EMPRESARIALES

ALGORITMO EN LÍNEA EFICIENTE PARA MEJORAR LA PRECISIÓN DE CONSULTAS SIMILARES EN UN SISTEMA DE RECUPERACIÓN DE LA INFORMACIÓN

AUTOR(ES):

Andrea Macarena Villa Saldías

Gaspar Adolfo Mella Paredes

PROFESOR GUÍA:

Sr. Claudio Orlando Gutiérrez Soto

CARRERA:

Ingeniería Civil Informática

Concepción, 2017

ÍNDICE

Contenido

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

RESUMEN

La recuperación de información (RI) es la búsqueda de material, generalmente documentos, de naturaleza no estructurada, usualmente texto, que satisface una necesidad de información desde una gran colección, generalmente almacenada en computadoras. Un sistema de recuperación de información (SRI) tiene como propósito representar y almacenar grandes cantidades de información. En un SRI común se pueden encontrar dos procesos principales: indexación y matching. El proceso de indexación corresponde a las representaciones y almacenamiento de documentos, que deben ser eficientes no sólo en el almacenamiento sino también en el acceso. El matching tiene la intención de estimar si un documento es relevante para responder una consulta realizada por un usuario. Esta coincidencia se suele representar a través de una puntuación. Cuando se aplica el proceso de similitud, un conjunto de documentos se devuelve al usuario como una lista clasificada por puntuación decreciente. Las mejoras en los SRI aparecieron a finales de los años 50. Las mejoras en la RI más importantes están relacionadas con la evaluación del SRI. La comunidad de la RI se ha beneficiado de la evaluación de las colecciones. Un ejemplo particular es proporcionado por las conferencias TREC (de las siglas en inglés, Text REtrieval Conferences), que organiza anualmente un taller. Estos talleres han ofrecido a los investigadores la oportunidad de medir la eficacia del sistema y comparar enfoques.

Diferentes enfoques en la RI se ocupan de la indexación, las funciones de comparación, los modelos formales y la retroalimentación de relevancia. Sin embargo, pocos enfoques aprovechan las búsquedas realizadas previamente por los usuarios. Las búsquedas pasadas proporcionan una fuente de información que puede ser útil para los nuevos usuarios (nuevas consultas). Por ejemplo, un usuario que busque un tema nuevo podría beneficiarse de las búsquedas pasadas realizadas por usuarios anteriores sobre el mismo tema.

Debido a la falta de colecciones adecuadas para la RI, hasta la fecha, existe un débil interés de la comunidad de la RI en el uso de los resultados de búsqueda anteriores. De hecho, la mayoría de las colecciones de la RI existentes se componen de consultas independientes. Estas colecciones no son apropiadas para evaluar enfoques enraizados en consultas anteriores porque no recogen información de consultas similares debido a la falta de juicios de relevancia. Por lo tanto, no hay una manera fácil de evaluar la conveniencia de estos enfoques. Además, la elaboración de estas colecciones es difícil debido al costo y tiempo necesarios. Por eso, una alternativa factible es simular tales colecciones.

Los documentos relevantes de consultas anteriores similares podrían utilizarse para responder a una nueva consulta. Este principio podría ser útil mediante la agrupación de búsquedas anteriores según sus similitudes. Dos categorías principales de agrupación pueden ser fácilmente identificables: agrupación estática y agrupación posterior a la recuperación. Por un lado, el agrupamiento estático es la aplicación tradicional de este método en una colección de documentos. Por otro lado, la agrupación posterior a la recuperación incluye información de la consulta en el agrupamiento de documentos. Normalmente, para la agrupación estática se utiliza la función de similitud de distancia

de coseno. Sin embargo, estas funciones no consideran el contexto específico bajo el cual se juzga la similitud de dos objetos

Por otro lado, hay aportes en el uso de técnicas y algoritmos probabilísticos con el objetivo de mejorar los resultados del proceso de recuperación. Dos tipos principales de investigación pueden ser fácilmente categorizados, técnicas de aprendizaje y optimización. Los enfoques basados en las técnicas de aprendizaje implican el uso de las Redes Bayesianas y sus variantes, mientras que las técnicas de optimización implican el uso de Algoritmos Genéticos. Las redes bayesianas proporcionan una representación de alto nivel, que es un modelo que representa el dominio del problema, esta representación se obtiene como resultado de un proceso de minería de datos, que podría ser complejo de adquirir en esfuerzo y tiempo. Por otra parte, los resultados aceptables de Algoritmos Genéticos requieren una función adecuada y una población inicial adecuada. Por lo tanto, una elección incorrecta de la función de aptitud así como la población inicial puede implicar altos recursos en tiempo computacional. En resumen, los recursos humanos y computacionales pueden llegar a ser altos en el tiempo. Por el contrario, las soluciones simples para implementar y representar pueden ser propuestas a través de algoritmos aleatorios.

Finalmente, en este Proyecto de título se implementa y evalúa un marco para simular enfoques ad hoc basados en resultados de búsquedas pasadas. Así, se propone y se evalúa un algoritmo, denominado Algoritmo 2.1, para mejorar la precisión y tiempo de respuesta a nuevas consultas, contrastando los resultados con otro algoritmo, denominado Algoritmo 2.0.

CAPÍTULO I

1. INTRODUCCIÓN

La recuperación de información implica tareas como organización, almacenamiento y búsqueda de información. Estas actividades tienen la finalidad de proporcionar información deseable para los usuarios. Los requisitos del usuario se traducen mediante consultas, donde se desea encontrar información que llene su expectativa. Es posible encontrar diferentes fuentes de información, tales como fuentes multimedia (por ejemplo, vídeo y audio), archivos XML entre otros. Por lo general, la información se extiende en un documento o un conjunto de documentos. Por otro lado, la relevancia sobre un documento o un conjunto de documentos, de acuerdo con la consulta del usuario, es proporcionada por los usuarios. Este último es conocido como juicio de usuario, por lo que se les da una consulta y un conjunto de documentos relacionado con la consulta del usuario. Un documento puede ser clasificado por un usuario como relevante o no relevante según la pertinencia de este documento con respecto a su necesidad de información (consulta). El conjunto de documentos, el conjunto de consultas y los juicios de usuarios es denominado por la comunidad de la RI como colecciones. Hoy en día, la información se materializa en miles y millones de documentos, por lo que se requiere la automatización de las tareas antes mencionadas. Por lo tanto, un SRI proporciona soporte al usuario en la búsqueda de información en una colección de documentos. Por lo general, un SRI está compuesto por una colección de documentos, una colección de consultas y juicios de usuarios. Un SRI proporciona un conjunto de documentos clasificados (ranking) que están relacionado con la consulta enviada por un usuario. Comúnmente, la lista de documentos proporcionados por el SRI se clasifica por una puntuación decreciente. Esta puntuación representa la similitud entre los documentos y la consulta calculada por el SRI (la similitud puede aplicarse entre consultas como entre documentos). Sin embargo, todos los documentos recuperados no satisfacen las necesidades del usuario. En otras palabras, no todos los documentos que aparecen en la lista son relevantes para el usuario. Además, la posición de los documentos relevantes en la lista es importante para el usuario. Por lo tanto, una buena situación es cuando los documentos relevantes aparecen no sólo en la parte superior de la lista, sino también en conjunto (es decir, los documentos pertinentes no deben estar muy dispersos). Esta propiedad está relacionada con la precisión en RI (ver Sección 2.4).

Para ilustrarlo, podemos suponer el siguiente escenario: para una consulta q, hay un subconjunto de documentos relevantes cdr para una colección de documentos. Un SRI S1 proporciona una lista de documentos donde cdr aparece en la parte superior de la lista. De manera similar, otro SRI S2 produce una lista de documentos donde cdr aparece en la parte inferior de la lista. Cuando comparamos ambos sistemas, S1 proporciona una mejor precisión que S2 porque el mismo cdr aparece en la parte superior. Un desafío importante y difícil para un SRI es proporcionar la mejor precisión. Hoy en día, los SRI más utilizados corresponden a los motores de búsqueda web. Éstos responden a millones de consultas por día en las colecciones, que implican miles de millones de documentos. Además, con el crecimiento sostenido en el número de documentos, la tarea de encontrar documentos relevantes para una consulta enviada por un usuario puede resultar no rentable. En resumen, el desafío más relevante para un SRI es proporcionar la mejor precisión posible.

Las contribuciones para la RI abordan diferentes perspectivas, tales como funciones de similitud, indexación, modelos formales y retroalimentación de relevancia. Sin embargo, pocas de estas contribuciones obtienen ventaja de las búsquedas realizadas anteriormente. Las búsquedas anteriores pueden ser una fuente útil de información para las búsquedas nuevas. En la literatura de RI, dos tipos de enfoques utilizados en el contexto de consultas anteriores son identificables. El primer tipo de enfoques se basa en las colecciones de las conferencias TREC[1]. La mayoría de estos enfoques utilizan la simulación para construir consultas similares con el objetivo de proporcionar un marco adecuado de evaluación. El segundo tipo de enfoques enraizados en el uso de las consultas históricas en la Web, la mayoría de las cuales se apoyan en consultas repetitivas (Gutiérrez-Soto, 2013). Por otra parte, el bajo nivel de interés en el uso de las consultas pasadas es debido a la falta de colecciones RI adecuadas. De hecho, la mayoría de las colecciones para la RI existentes se componen de consultas independientes. Estas colecciones no son útiles para evaluar enfoques basados en consultas pasadas, ya que no recogen consultas similares para las que se proporcionan juicios de pertinencia real. Por lo tanto, una alternativa es construir ambientes adecuados para analizar las ventajas de los enfoques basados en resultados de búsqueda anteriores usando simulación (Gutiérrez-Soto, 2013). Finalmente, una contribución de este Proyecto de título es el uso de la simulación para dar un entorno ideal basado en consultas pasadas. Es esencial destacar que todos los aportes del Proyecto se basan en el uso de documentos relevantes obtenidos de la consulta anterior (consulta pasada) más similar para una consulta (nueva consulta) con el objetivo de mejorar la precisión de los SRI, una forma es usar el agrupamiento de consultas ejecutadas con anterioridad por un usuario. Para ello utilizaremos el algoritmo presentado en la Sección 3.3, a través del método de construcción de simulación de un conjunto de documentos y consultas presentado en la Sección 3.1.1.

El Agrupamiento y la Cola de Prioridad son estructuras de datos utilizadas en el acceso a documentos, por otra parte el agrupamiento se utiliza para la creación de índices. La cola de prioridad admite una respuesta eficiente a las consultas de palabras clave, mientras que el agrupamiento permite crear grupos similares de documentos. La agrupación en la RI se ha utilizado para mejorar la eficiencia y la eficacia de los SRI. Por un lado, el agrupamiento de consultas es un tipo de agrupación posterior a la recuperación, que se dedica a buscar consultas similares en un grupo de consultas. La similitud entre las consultas está relacionada con la superposición entre los términos de las consultas. Normalmente, las funciones de similitud como el coseno se utilizan para medir la coincidencia entre las consultas (Salton, 1983). Sin embargo, estas funciones no consideran el contexto específico bajo el cual se juzga la similitud de dos objetos. Por otro lado, podemos decir que los documentos relevantes para una consulta tienden a ser muy similares en el contexto definido por la consulta (Gutiérrez-Soto, 2013). De este modo, las consultas con sus documentos se pueden almacenar para proporcionar un contexto. Por lo tanto, otra contribución de este Proyecto de título es la agrupación de consultas ejecutadas con anterioridad junto a sus documentos relevantes, donde los documentos relevantes de una consulta anterior más similar se utilizan para responder a una nueva consulta. Además, es importante considerar los enfoques que promueven respuestas a las consultas en tiempos razonables no sólo en el diseño o búsqueda de mecanismos eficientes para mejorar la precisión de los SRI (es decir, el tiempo involucrado en una tarea de minería de datos para

[1] Se puede ampliar información en *http://trec.nist.gov/*.

asignar el mejor SRI, que mejora la precisión de un tipo específico de consulta), sino también en el tiempo de ejecución del SRI para proporcionar documentos para una consulta determinada.

En resumen, se pueden identificar fácilmente tres problemas:

1) Los enfoques que aprovechan los resultados de búsqueda anteriores se basan en consultas independientes en lugar de consultas similares. Además, las colecciones que permiten evaluar los resultados de búsqueda anteriores no sólo son pocos, sino también casi desconocidos. Esto se debe a la construcción de estas colecciones, que implica un alto costo no sólo en el tiempo sino también en el esfuerzo (Sanderson, 2010).

2) La relación entre los documentos asociados con consultas similares no ha sido ampliamente estudiada (Gutiérrez-Soto, 2013). El agrupamiento de consultas se ha aplicado mediante el uso de funciones de similitud como las distancias de coseno entre las consultas.

3) Por lo general, los enfoques que proporcionan la mejor respuesta a una consulta, implican un análisis exhaustivo de todos los sistemas posibles (es decir, se da cuando el enfoque considera varias consultas y varios sistemas que responden a estas consultas). Típicamente, estos enfoques se basan en el aprendizaje de máquinas o la minería de datos, lo que puede implicar un alto costo en computación y recursos humanos. La principal motivación de este proyecto de título ha sido el estudio de modelos de la RI para mejorar la precisión utilizando los resultados de búsqueda anteriores. Para ello, y teniendo en cuenta las cuestiones antes mencionadas, las principales contribuciones de este Proyecto de título son:

1) Un marco de simulación de conjunto de documentos y consultas, que permite construir un entorno ad hoc para evaluar enfoques bajo el contexto de resultados de consultas anteriores similares.

2) Un Algoritmo en línea, que recoge y almacena resultados de búsquedas anteriores para mejorar la precisión respondiendo a nuevas consultas utilizando documentos relevantes del pasado más similar a la consulta.

3) La comparación en la evaluación de dos SRI (Algoritmo 2.0 y Algoritmo 2.1) bajo el contexto de resultados anteriores similares.

1.1. Objetivos

1.1.1. Objetivo General

Implementar un Algoritmo en Línea, que permita proporcionar respuestas eficientes a nuevas consultas sobre un SRI. Esto considerando los resultados de consultas pasadas similares y repetidas. Además de identificar los objetos más consultados, los cuales pueden ser distribuidos de manera eficiente, reduciendo el tiempo de búsqueda para dar una respuesta a una nueva consulta.

1.1.2. Objetivos Específicos

- Implementación de un Algoritmo en Línea el cual recupere documentos de manera eficiente y mejore la precisión de éstos.
- Analizar y determinar distintos escenarios experimentales acorde al conjunto de consultas similares y repetidas.
- Analizar el rendimiento del algoritmo (Algoritmo 2.1) a través de un análisis empírico, contrastando su rendimiento con una recuperación tradicional (Algoritmo 2.0) bajo el contexto de un SRI que responde con resultados de búsquedas pasadas similares (Gutiérrez-Soto, 2013).
- Proponer mejoras a los resultados obtenidos en los experimentos.

1.2. Límites

Los algoritmos de los métodos de acceso mencionados sólo serán estudiados y no se les realizarán modificaciones a ellos o sus estructuras de datos asociadas para cumplir con las mejoras de eficiencia.

1.3. Organización del documento

El resto el documento está organizado en los siguientes capítulos. El Capítulo 2 presenta los conceptos preliminares necesarios para el desarrollo del proyecto, tales como la definición de la Recuperación de Información y la estructura para la representación de documentos y consultas, abordando su generación a partir de una matriz y la evaluación de un Sistema de Recuperación de Información. El Capítulo 3 presenta la forma de representar, mediante simulación, un SRI, mostrando la representación de las tareas básicas del SRI y los algoritmos desarrollados para realizar las funciones principales del algoritmo en línea. En el Capítulo 4 se describe el diseño experimental utilizado para dar soporte a las mejoras de eficiencia del SRI y se presentan los resultados de la experimentación realizada para medir la mejora en eficiencia en términos de la recuperación de los documentos relevantes, precisión y el tiempo de ejecución del SRI para la respuesta a consultas similares. Finalmente, en el Capítulo 5 se presentan las conclusiones obtenidas como resultado del Proyecto de título.

CAPÍTULO II

2. CONCEPTOS PRELIMINARES

2.1.Recuperación de la Información

La recuperación de información trata con la representación, el almacenamiento, la organización y el acceso a la información (Salton, 1983). Además, el problema de la recuperación de información es que, dada una necesidad de información (consulta) y un conjunto de documentos, la respuesta a esta consulta sea a través de la organización de los documentos, es decir, ordenar éstos del más al menos relevante y presentar un subconjunto de aquellos de mayor relevancia (Baeza-Yates, 1999). En la solución a este problema se identifican dos etapas: a) la elección de un modelo que permita calcular la relevancia de un documento frente a una consulta. b) diseño de algoritmos y estructuras de datos que implementen el modelo de forma eficiente.

Un SRI para cumplir sus objetivos debe realizar tareas básicas como la representación lógica de los documentos (por ejemplo: conjunto de términos) y consultas (necesidad de información del usuario) así como también la evaluación de los documentos respecto de una consulta, esto para establecer la similitud. Por otra parte es necesario determinar la clasificación (ranking) de los documentos considerados relevantes para formar el conjunto solución clasificados por una puntuación decreciente. Con la representación de esta respuesta se puede aumentar la calidad de ésta mediante la retroalimentación. Por lo tanto, un SRI debe contar con mecanismos para la localización de los documentos y presentarlos posteriormente al usuario, en la Ilustración 1 se resume las tareas antes mencionadas.

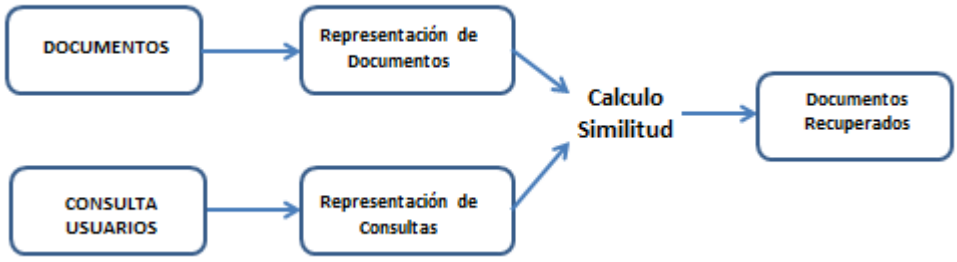


Ilustración 1: Arquitectura de un SRI.

Fuente: <http://www.hipertext.net>, núm. 5, 2007

En un SRI la respuesta ideal está formada solamente por documentos relevantes a la consulta, pero, en la práctica, esta no es aún alcanzable. Esto se debe a que, entre otros motivos, existe el problema de compatibilizar la expresión de la necesidad de información (usuario), el lenguaje y la expresión de los documentos. Además, hay una carga de subjetividad que depende de los usuarios.

Entonces, el SRI recupera la mayor cantidad posible de documentos relevantes, minimizando la cantidad de documentos no relevantes en la respuesta. En términos de eficiencia, se plantea la idea de precisión de la respuesta, es decir, mientras más documentos relevantes tiene el conjunto solución para una consulta dada, más preciso será.

2.2. Representación de documentos y consultas

Generalmente los documentos son transformados por un SRI de la forma original a una representación interna, este proceso se llama indexación. El propósito de este proceso es proporcionar una representación de la información. Para lograr este objetivo, se asigna un conjunto de características de indexación para cada documento. Las características más relevantes de un documento corresponden a una lista de palabras, que permiten discriminar entre ellas, éstas son conocidas como términos. De hecho, un documento no sólo está representado por esta lista de términos, sino que también se accede por términos que pertenecen a su lista. En el contexto de este Proyecto de título, las representaciones de los documentos son proporcionadas por las listas de términos, que se extraen de una biblioteca de términos, representados por el alfabeto inglés.

Un proceso de normalización tiene lugar antes de la indexación. El objetivo de este proceso es proporcionar sólo términos relevantes. Por ejemplo, las palabras con alta frecuencia en el documento (palabras vacías) no serán consideradas en la indexación (Rijsbergen, 1979). Estas palabras son artículos (por ejemplo, el) y preposiciones (por ejemplo, en). La principal ventaja de este proceso es reducir el volumen de texto hasta un 50 por ciento. Otro proceso antes de la indexación corresponde a eliminar los sufijos de las palabras restantes del texto de entrada. Para ello, se aplica un algoritmo de derivación para reducir las palabras a una forma de raíz común. Por ejemplo, el algoritmo de derivación reduce las palabras 'cardiovascular', 'cardiología' a la palabra cardio, que estará en el vocabulario de los términos del índice. Es importante destacar que en los experimentos relacionados con la simulación se ha omitido el proceso de derivación. Se hace esta elección porque en cada experimento se construye en un entorno ideal, donde cada término es único. Por lo tanto, no hay una raíz común entre los términos. De la misma manera, la normalización (es decir, la eliminación de palabras vacías) se descarta porque se supone que los términos usados son términos representativos. En resumen, puede considerarse como la aplicación general del mismo proceso de derivación y normalización en todos los experimentos.

Los SRI utilizan principalmente estructuras de datos para almacenar los términos, índices y estructuras de archivos (Rijsbergen, 1979). La estructura de datos que se utilizó para propinar un escenario ad hoc de almacenamiento la presentamos en la Sección 3.1.

2.3. Modelos para la Recuperación de Información

El diseño de un SRI se realiza bajo un modelo, donde queda definido cómo se obtienen las representaciones de los documentos y de la consulta, la estrategia para evaluar la relevancia de un documento respecto a una consulta (grado de semejanza) y los métodos para establecer la importancia (clasificación) de los documentos de salida (Villena Román, 1997). Por lo tanto, a partir de la tarea inicial que realiza el usuario en el sistema, podemos distinguir los siguientes modelos

clásicos: el modelo booleano, el modelo el modelo vectorial y el modelo probabilístico. Siendo el modelo vectorial el más utilizado en la actualidad, especialmente en la Web. En este modelo, la representación de documentos se expresa mediante vectores que recogen la frecuencia de aparición de los términos en los documentos.

	t_1	t_2
D_1	0.4	0.2
D_2	0.5	0.3
D_3	1	0.2
D_4	0.8	0.4

Tabla 1: Matriz de términos y documentos en el Espacio Vectorial.

En teoría, los documentos que contengan términos similares estarán a muy poca distancia entre sí. De igual forma se trata a la consulta. Luego, a partir de una consulta dada es posible devolver una lista de documentos ordenados por distancia (documentos más relevantes primero). Por ejemplo, se tiene una consulta q = {t_1, t_2}, con valor de frecuencia de término 0,4 y 0.3, respectivamente. Para responder a la pregunta se calcula la semejanza entre la consulta y los documentos que contienen los términos t_1 y t_2 a través de la función de distancia coseno. Consideremos los documentos D_2 y D_3 de la Tabla 1, la similitud entre los documentos y la consulta q resulta:

$$\text{sim}(q,D_2) = \frac{(0.4 \times 0.5) + (0.3 \times 0.3)}{\sqrt{(0.4)^2 + (0.3)^2 \times (0.5)^2 + (0.3)^2}} = \frac{0.46}{\sqrt{0.26}} = 0{,}9$$

$$\text{Sim}(q,D_3) = \frac{(0.4 \times 1) + (0.3 \times 0.2)}{\sqrt{(0.4)^2 + (0.3)^2 \times (1)^2 + (0.2)^2}} = \frac{0.29}{\sqrt{0.085}} = 1$$

Por lo tanto la respuesta para la consulta q es: {D_3, D_2}.

2.4. Evaluación de la Recuperación de Información

Los SRI son susceptibles de estar bajo evaluación, específicamente a la determinación de evaluación que determine las medidas que permitan valorar su efectividad. (Baeza-Yates, 1999), selecciona los siguientes criterios: la eficacia en la ejecución, el efectivo almacenamiento de los datos, la efectividad en la recuperación de la información y la serie de características que ofrece el sistema al usuario. Sin embargo, la evaluación de un SRI no es una tarea sencilla debido a que el conjunto de respuesta no es exacta, se requiere ponderar cómo este se ajusta a la consulta y ésta a la necesidad de información del usuario. Aquí aparecen las cuestiones subjetivas que se plantean al especificar una consulta, al adoptar una representación lógica de los documentos de la colección y al utilizar una función de ranking (clasificación ordenada decreciente) determinada.

En esta sección se presenta la medida de evaluación con énfasis en los aspectos centrales que conforman el núcleo de este Proyecto de título. Sin embargo, el enfoque más relevante de ésta corresponde a la evaluación de la cantidad de documentos relevantes proporcionados por un SRI en respuesta a una consulta de un usuario y en el intervalo de tiempo transcurrido entre que el sistema recibe la consulta del usuario y presenta las respuestas.

Desde los primeros esfuerzos relacionados con la evaluación de los SRI hasta la actualidad, la aproximación clásica para describir el escenario de la recuperación consiste en determinar cuántos documentos relevantes se recuperaron y cómo se rankearon para entregarlos al usuario. Cuando un usuario plantea una consulta a un SRI, obtiene como respuesta una lista de documentos determinado por el sistema. La respuesta está formada por documentos relevantes y no relevantes. Además, la lista, generalmente, no contiene todos los documentos de la colección (en colecciones grandes sería imposible revisar toda la respuesta). En la Ilustración 2 se ejemplifica esta situación. Dada una consulta cualquiera, un SRI recuperará el grupo identificado como C, de los cuales solo una parte es relevante D.

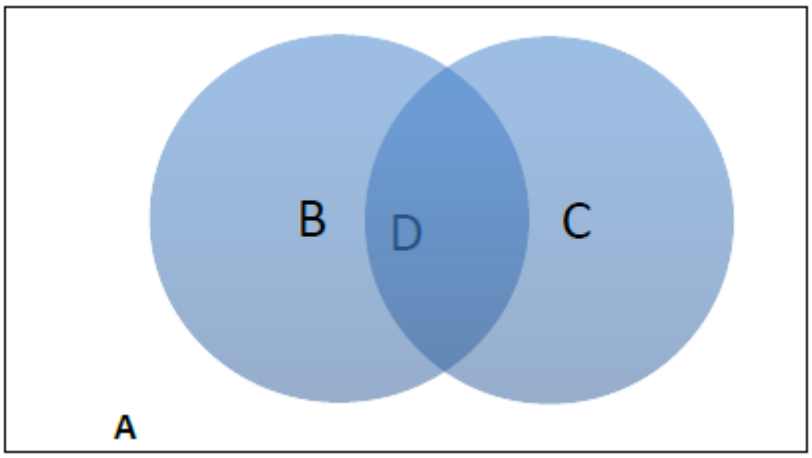


Ilustración 2: documentos en una colección

A: Todos – B: Relevantes – C: Recuperados – D: Relevantes recuperados (B ∩ C)

Uno de los aspectos más importantes en la evaluación de los SRI es la precisión. La precisión corresponde al porcentaje de los documentos relevantes recuperados por el sistema, del total de documentos relevantes, formalmente definida por:

$$P = \frac{|\{doc\_relevantes\} \cap \{doc\_recuperados\}|}{|\{doc\_recuperados\}|}$$

Esta medida permite evaluar la habilidad del sistema para ponderar el conjunto de documentos, donde se espera que los resultados se presenten en una lista ordenada de documentos (por ejemplo, como en los buscadores web de hoy en día).

2.5. Búsquedas pasadas

El uso de los resultados de búsqueda anteriores para responder a una nueva búsqueda no es reciente. Varios enfoques que tratan con el uso de las preguntas históricas para mejorar los resultados de la búsqueda, se pueden encontrar en la literatura de la RI. La repetición y la expansión de consultas han alcanzado un éxito considerable con la explotación de la información disponible dentro de los archivos web. Los archivos de registro se han estudiado ampliamente considerando consultas repetidas y reformulación de consultas (es decir, agregando términos o eliminando términos).

2.6. Resumen

En este capítulo se describió todo lo relacionado con la RI como también la medida de evaluación de un SRI. Se ha proporcionado el marco utilizado para dar soporte a la representación de un SRI, específicamente a la forma de representar y evaluar un SRI. En el siguiente Capítulo se profundizara la construcción de un SRI, utilizando la simulación para representar una colección de documentos y consultas, para evaluar nuestro enfoque basado en la búsqueda de respuesta, para una consulta, utilizando el resultado de consultas pasadas.

CAPÍTULO III

3. SIMULACION EN LA RI

El uso de la simulación en RI no es reciente, ésta puede ser vista como un método en el que una gran colección de consultas junto con sus juicios se pueden obtener sin la interacción del usuario. La simulación se ha utilizado en diferentes contextos en RI. Un algoritmo fue desarrollado con el propósito de simular la pertinencia de los juicios de los usuarios. Aquí, la precisión simulada se compara con la precisión real. Los autores propusieron un modelo para construir temas simulados que son comparables a temas reales. Se han creado trabajos dedicados a simular la interacción entre las consultas, el registro de clics y las preferencias de los usuarios. (Yisong Yue, 2009), proponen un marco de aprendizaje en línea basado en comparaciones de pares, que pueden aprender en tiempo real del comportamiento del usuario observado en motores de búsqueda y otros SRI. Además, los autores producen resultados por simulación a través del algoritmo Dueling Bandit Gradient Descent (DBGD). Para un entorno de simulación más realista, se utiliza un conjunto de datos de búsqueda web real de Microsoft Research. La idea es simular usuarios que emitan consultas mediante muestreo de consultas en el conjunto de datos. Para cada consulta, las funciones de recuperación de competencia producirán clasificaciones, donde posteriormente el usuario preferirá al azar una clasificación sobre la otra. Este conjunto de datos se utiliza en el primer paso para simular el comportamiento del usuario sobre la configuración de aprendizaje en línea. A partir de los resultados empíricos, los autores señalan que el algoritmo DBGD puede aplicarse en un contexto general.

Una colección habitual de RI se compone de tres partes: un conjunto de documentos, un conjunto de consultas y un conjunto de juicios de relevancia por consulta (es decir, indicaciones sobre documentos considerados relevantes o no relevantes por un usuario). En primer lugar, hay varios enfoques que aprovechan el uso de los resultados de búsqueda anteriores, con el propósito de ser utilizados en nuevas búsquedas se basan en dos tipos de colecciones: las colecciones TREC y los archivos de registro. Por ejemplo, a partir del escenario ad-hoc propuesto por (Gutiérrez-Soto, 2013), que modificó todos los enfoques basados en las colecciones TREC, con el objetivo de simular el uso de los resultados de búsquedas anteriores. Por otro lado, las colecciones basadas en archivos de registro no contienen juicio de pertinencia de los usuarios. Además, este marco es necesario en la evaluación de algoritmos aleatorios ya que proporciona un ambiente experimental limpio, es decir, los procesos de detención se omiten (ver Sección 2.2), y robustos, es decir, los resultados finales no dependen de características particulares de un sistema. En consecuencia, nuestro enfoque se propone por simulación basado en resultados de búsquedas anteriores.

3.1.Simulación Framework

Nuestro método se divide en tres pasos: a) La creación de documentos y consultas, b) la simulación de juicios de usuarios, c) y la recuperación de documentos relevantes basado en consultas pasadas.

3.1.1. Creación de documentos y consultas

En un comienzo de construye un conjunto de términos. Cada término está compuesto de letras del alfabeto inglés. Este conjunto de términos se puede dividir en subconjuntos denominados tópicos para representar temas diferentes (por ejemplo, informática, biología, etc.). Además, cada documento se define de acuerdo a todos los temas. Para construir un documento, los temas se seleccionan utilizando la distribución exponencial, luego los términos que constituyen el documento se eligen utilizando una distribución uniforme. Así, un documento se construye con términos de un tema, principalmente, pero no exclusivamente. Las consultas pasadas se crean a partir de documentos. Para crear una consulta pasada, se elige un documento con una distribución uniforme. Los términos que constituyen la consulta se eligen del documento bajo distribución uniforme. Es importante enfatizar que la intersección entre las consultas pasadas es vacía, es decir, no tienen términos en común. Las consultas nuevas se construyen a partir de consultas pasadas con el fin de estimar que un usuario nuevo se podría beneficiar de las búsquedas realizadas anteriormente. Por ejemplo, para cada consulta pasada q se crea dos nuevas consultas. Una consulta similar $q_{similar}$, ya sea cambiando o añadiendo un término y una consulta idéntica $q_{repetida}$, con términos idénticos de la consulta, en este caso se eligió la consulta similar. Por lo tanto, la consulta más similar para la nueva consulta es su consulta pasada correspondiente. Consideremos los siguientes datos:

Términos (Vocabulario):
$T = \{t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8\}$

Documentos:
$D_1 = \{t_1, t_2, t_3, t_4, t_5\}$
$D_2 = \{t_1, t_2, t_3\}$
$D_3 = \{t_2, t_4, t_5, t_8\}$
$D_4 = \{t_1, t_3, t_5, t_7\}$
$D_5 = \{t_4, t_5, t_6\}$

Aleatoriamente se elige un documento, en este caso D_1, esto es para crear la consulta a partir del documento seleccionado: $q = \{t_1, t_2, t_3, t_4\}$, entonces tenemos que:

$q_{pasada} = \{t_1, t_2, t_3, t_4\}$
$q_{similar} = \{t_1, t_2, t_3\}$, se quita un término de q_{pasada}
$q_{repetida} = \{t_1, t_2, t_3\}$, se asigna el total de términos de la consulta, en este caso $q_{similar}$

Entonces podemos notar que una consulta pasada está representada por los términos de un documento.

3.1.2. Simulación juicio de usuario

Para simular la decisión dada por un usuario acerca de si un documento es relevante o no es relevante para una consulta determinada. Nos basamos, para la implementación, en la distribución Zeta. Esta distribución da una discreta aproximación de la ley de Bradford. La ley de Bradford dice

que, entre la producción de artículos de revistas, hay un número heterogéneo de artículos donde los artículos más relevantes están en pocas revistas, mientras que un número de artículos relevantes se distribuyen en una gran cantidad de revistas.

En nuestro caso, para una consulta, significa que los documentos más relevantes deben estar en la parte superior de la lista porque son los más similares con respecto a la consulta (esto es, los artículos más relevantes están en pocas publicaciones), mientras que algunos documentos relevantes deben ser distribuidos en la parte inferior del documento de la lista dada una consulta.

A partir del escenario proporcionado por (Gutiérrez-Soto, 2013) que, supone que para dos consultas similares q y q', cuando un documento es relevante para una consulta, también podría ser relevante para la otra consulta. De esta manera, podemos decir que, hay un subconjunto de documentos relevantes comunes para ambas consultas. Esto no implica que todos los documentos relevantes para la consulta q, sean documentos relevantes para la consulta q'. Por lo tanto, cuando se mide la efectividad, por ejemplo con precisión en diez documentos recuperados (P @ 10), la precisión para la consulta q no es necesariamente la misma que para la consulta q'. Con el objetivo de simular este escenario, usamos la distribución zeta para determinar documentos relevantes sobre un subconjunto de documentos comunes entre consultas similares. Después, para tener todos los documentos relevantes para cada consulta, la distribución Zeta se aplica de nuevo en la lista de documentos, conservando los documentos relevantes del subconjunto de documentos comunes.

Por ejemplo, en la Ilustración 4 muestra los documentos relevantes para las consultas q y q', que son similares. Los documentos relevantes tienen valor peso 1 y los documentos no relevantes tienen valor peso igual a 0. Primero, se recupera la lista de documentos de ambas consultas, desde la intersección (D_3, D_5, D_6) se calculan los documentos relevantes comunes a ambas consultas utilizando distribución Zeta. Posteriormente, conservando los documentos relevantes de la intersección (D_3, D_5), se ordenan los documentos relevantes de q' (decreciente) y a los documentos que tienen valor peso 0 se le calcula los nuevos documentos relevantes para la consulta q'. Esto es para obtener el nuevo juicio de usuario. Finalmente, para P @ 10 de la consulta q no es necesariamente la misma que para la consulta q'.

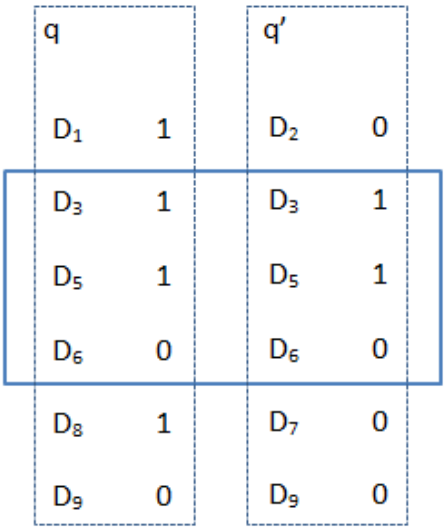


Ilustración 3: Obtención de juicios de usuarios para consultas identificadas como similares, mediante la comparación de los documentos de la nueva consulta q' con la consulta q.

Por otra parte, en la Ilustración 4 se ve el caso cuando las consultas son identificadas como repetidas, esto es cuando los documentos relevantes para ambas consultas son idénticos. Al igual que el ejemplo anterior, los documentos relevantes para cada consulta tienen valor 1 y los documentos no relevantes valor 0. Primero se recupera la lista de documentos para consulta q y q', a partir de la lista completa se calculan los documentos relevantes de ambas consultas utilizando distribución Zeta. Posteriormente, se ordenan los documentos de la consulta q', ubicando los documentos pertinentes en la parte superior. Finalmente, los documentos relevantes de la consulta q podría ser útil para la consulta q', esto es porque un nuevo usuario se puede beneficiar de la respuesta de una búsqueda pasada (por ejemplo, cuando un usuario pregunta sobre un tema nuevo), por lo tanto P @ 10 de la consulta q puede ser la misma que para la consulta q'.

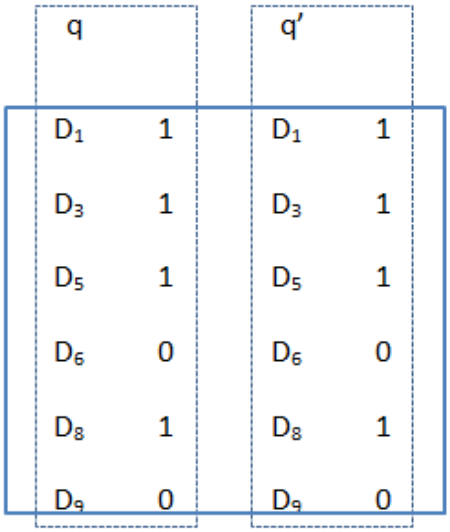


Ilustración 4: Obtención juicios de usuario para consultas identificadas como idénticas, mediante la comparación de los documentos de la nueva consulta q' con la consulta q.

3.2. Recuperación usando consultas pasadas

La idea básica detrás de nuestro enfoque es incorporar al sistema cada consulta con su conjunto de documentos asociados(los documentos son parte de la respuesta a esta consulta). Así, el sistema tiene no sólo los conjuntos de documentos sino también las consultas realizadas por usuarios (consultas ejecutadas pasadas) con su conjunto de documentos. Al principio sólo hay documentos sin las consultas, pero cada vez que una consulta es procesada por el sistema, se agrega con sus documentos al sistema. Cuando se envía una nueva consulta, primero ésta se comprueba y compara con la agrupación de consultas ejecutadas con anterioridad, así como también se comparan con las consultas anteriores que están en el sistema. Cuando se encuentra una consulta en el conjunto de consultas ejecutadas con anterioridad se pueden recuperar los documentos relevantes de esta consulta para responder la nueva consulta. Finalmente si la consulta no fue ejecutada con anterioridad, puede realizarse una recuperación tradicional de los documentos relevantes para responder la nueva consulta, esto es a través de la distancia del coseno (ver Sección 2.3).

3.3. Diseño Algoritmo en línea

Lo que hace el algoritmo es incorporar al sistema todas las consultas junto a sus documentos. Además almacena el conjunto de consultas ejecutadas con anterioridad con sus documentos pertinentes en una cola de prioridad. Primero sólo hay documentos sin consultas, pero cada vez que se procesa una nueva consulta se agrega con sus documentos al sistema. A partir de la creación del conjunto de consultas asociadas a sus documentos se selecciona aleatoriamente, bajo distribución exponencial, una serie de consultas que representan el conjunto de consultas pasadas más consultadas (procesadas) en el sistema, implementado en una cola de prioridad que almacena valores máximos. El primer elemento que almacena la cola de prioridad es la consulta que más veces

fue procesada en el sistema. A partir de lo anterior, primero se comprueba si la nueva consulta se encuentra en la cola de prioridad, se determina si es similar o repetida, esto es para determinar el método de recuperación de documentos relevantes para responder a la nueva consulta. Cuando se determina que es una consulta repetida, se responderá con los documentos relevantes de la consulta almacenada en cola de prioridad. Si se comprueba que la consulta es similar, entonces se recuperan los documentos relevantes de la consulta almacenada en la cola de prioridad para luego volver a realizar el nuevo juicio de usuario, usando distribución Zeta, con el fin de determinar los nuevos documentos relevantes para la nueva consulta, esto se aplica a los documentos recuperados que no son relevantes (representados con valor peso igual a 0), la idea de determinar un nuevo juicio de usuario, es para representar la subjetividad entre usuarios, es decir, los documentos determinados pertinentes para un usuario no es necesariamente los mismos que para otro usuario. Finalmente, si la nueva consulta no se encuentra en la cola de prioridad se procede a responder de la forma tradicional.

Por ejemplo, cuando se envía una nueva consulta q' al sistema, primero se comprueba y compara con el conjunto de consultas ejecutadas con anterioridad. Si existe una consulta bastante similar o repetida en el sistema, se recupera el conjunto de documentos N de la consulta q para responder a la nueva consulta q'. Al mismo tiempo, la nueva consulta debe ser verificada y comparada con el conjunto de consultas anteriores del sistema. De ambos, es posible comparar nuestro enfoque. Para ejemplificar nuestro enfoque, en la Ilustración 5 podemos observar que cuando un usuario ingresa una nueva consulta, se realiza primero una búsqueda en el conjunto de consultas ejecutadas con anterioridad. Cuando la nueva consulta es comparada y luego verificada (comparación de documentos) como repetida o similar, el sistema responde con los documentos recuperados relevantes de la consulta pasada. De otro modo, si la consulta q' no se encuentra en la agrupación de consultas ejecutadas pasadas se procede a responder con la búsqueda de la consulta más similar dentro de la agrupación de consultas del sistema.

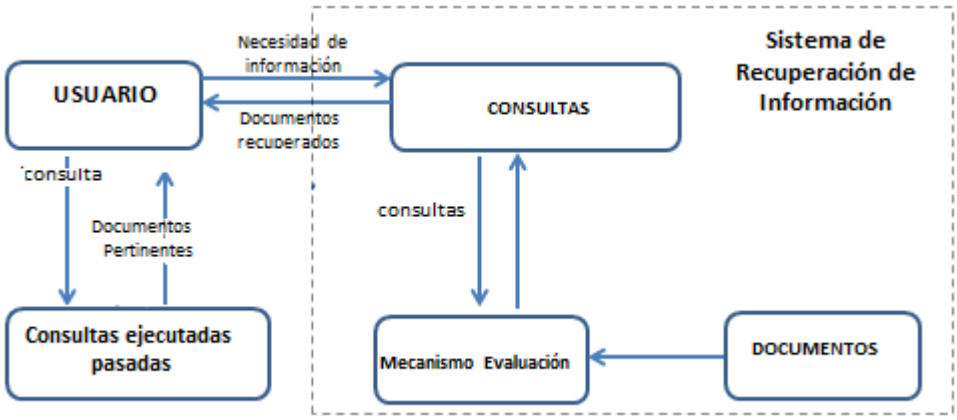


Ilustración 5: Composición SRI.

3.3.1. Método en detalle

Primero hay un conjunto de documentos D. El conjunto de consultas Q está vacío. Al principio la primera consulta q es dada por el usuario, se verifica con el conjunto de documentos D. A continuación, se recuperan los documentos N para q. Ahora, este conjunto de documentos está asociado con la consulta q en un nuevo conjunto y este conjunto se agrega al sistema. A partir de este conjunto Q no está más vacío. Nuestro método consta de dos partes, en primer lugar se crean consultas similares e idénticas, esto a partir del conjunto de documentos D explicado en la Sección 3.1.1 .Debemos precisar que una consulta pasada se crea a partir de un documento y una nueva consulta se crea a partir de una consulta pasada.

El algoritmo presentado en Ilustración 6, se puede observar la creación de las consultas. Para representar las consultas pasadas es necesario recurrir al conjunto de documentos D, que permite determinar los términos que constituye a la consulta, esto partir del documento seleccionado aleatoriamente de la colección D. Consideremos la consulta q_1, primero se selecciona un documento aleatoriamente asignado al identificador idDoc, a partir del identificador se puede determinar, mediante la función ObtenerTopicoD(), el tema al cual pertenece un documento (ver Sección 2.2). Posteriormente, con la función ObtenerTDocumento() se obtienen los términos del documento (termdoc) elegido para crear la consulta q_1. Para generar la consulta q_1 se determina el número de términos (Nterm) que tendrá esta consulta. En la función EstaEn () se compara cada término del documento seleccionado de la colección de documentos D, si existe coincidencia se retorna un 1 sino 0. Esto es para asignar los términos a la consulta q_1. Posteriormente, se crean dos consultas q_2 y q_3, donde q_2 representa a la consulta similar y q_3 consulta repetida. La consulta q_2 se le asigna Nterm-1 términos de la consulta q_1 y la consulta q_3 se le asignan todos los términos de la consulta q_2. Finalmente, se asignan los términos que componen a cada consulta al conjunto de consultas Q y se retorna el agrupamiento de consultas headQ. Este proceso se realiza de forma iterativa hasta crear el total de consultas (NQueries).

Algorithm 1 Creación Consultas

Input: Conjunto de documentos D $headD$, $NQueries$ número de consultas a generar, $NDoc$ número de documentos.
Output: conjunto de consultas Q $headQ$.

```
termdoc[30] ← 0;
termquery[8] ← 0;
termquery2[8] ← 0;
termquery3[8] ← 0;
headQ ← NULL;
headD ← ConjuntoDocumentosD;
while idquery ≤ NQueries do
  idDoc ← (random(1, ..., NDoc);
  termdoc ← ObtenerTDocumento(hedD, idDoc);
  IdTopic ← ObtenerTopicoD(idDoc, headD);
  for i = 0, l = 0; i < 30; i + + do
    if termdoc[i] ≠ 0 then
      l + +;
    end if
  end for
  Nterm = (random(3, .., l);
  while i < NTerm and j < l do
    if  EstaEn (termdoc[j], headD) == 1 AND j < 1 then
      j + +;
    else
      termquery[i + +] = termdoc[j + +];
    end if
  end while
  while j < (l − 1) do
    for i = 0; i < NTerm; i + + do
      //para la consulta similar
      termquery2[i − 1] ← termquery[i];
      //para la consulta idéntica
      termquery3[i − 1] ← termquery2[i − 1];

      headQ ← insertQ(headQ, TermQuery, idQuery, IdTopic);
      idQuery = idQuery+1;
      headQ ← insertQ(headQ, TermQuery2, idQuery, IdTopic);
      idQuery = idQuery+1;
      headQ ← insertQ(headQ, TermQuery3, idQuery, IdTopic);
      idQuery = idQuery+1;
    end for
  end while
end while
return  headQ;
```

Ilustración 6: Pseudocódigo Algoritmo - Creación de Consultas

A partir de lo anterior, supongamos que se ingresa una primera consulta, en este caso q_1. El sistema evalúa y compara los términos de la consulta q_1 y los términos de cada documento del conjunto D, con el fin de determinar un subconjunto de documentos similares como respuesta para la consulta q_1. Primero se obtienen los términos de la consulta q_1, para compararlos con los términos que componen a los documentos del conjunto D. Supongamos que al comparar los términos de la consulta q_1 se encuentran coincidencias de términos con 93 documentos del conjunto D, se seleccionan los primeros 30 documentos relevantes para posteriormente asociar los 30 documentos a la consulta q_1, determinando así la respuesta a la consulta q_1. Una vez que se expande la consulta, esto es al agrupar la consulta junto a sus documentos relevantes, se podrá hacer uso de esta respuesta para responder en un futuro a una nueva consulta similar ingresada en el SRI.

En segundo lugar se crea un conjunto de consultas que representan a las consultas ejecutadas en el pasado por otros usuarios, esto es a partir del conjunto de consultas asociadas a sus documentos recuperados pertinentes (conjunto QD), que se seleccionan bajo distribución exponencial con la finalidad de representar aquellas consultas que fueron más veces consultadas en el sistema. Entonces se tiene un conjunto QD (consultas asociadas a documentos) las que serán almacenadas en una cola de prioridad, tal como dice su nombre, esta estructura de datos posee un carácter de prioridad que en nuestro caso representa el número de veces que se procesó una consulta en el SRI. Por lo tanto a partir de este carácter se ordenará la lista de consultas.

En la Ilustración 9 se presenta el pseudocódigo para la creación del conjunto de consultas ejecutadas con anterioridad. Para representar aquellas consultas más consultadas en el SRI se requiere del conjunto QD, el número de consultas NQueries y el rango range que representa la división en intervalos (NQueries/range) de la agrupación de consultas QD. Además cada rango tendrá un loop de 100 iteraciones para generar aquellas consultas más consultadas. Consideremos un conjunto de 30 consultas asociadas a sus documentos y un range de valor 5. A partir de lo anterior el conjunto de consultas QD se distribuye en intervalos. Cada intervalo contiene un número determinado de consultas (Nelements), en este caso seis consultas, las cuales serán seleccionadas aleatoriamente. En la Ilustración 7 se muestra lo mencionado anteriormente.

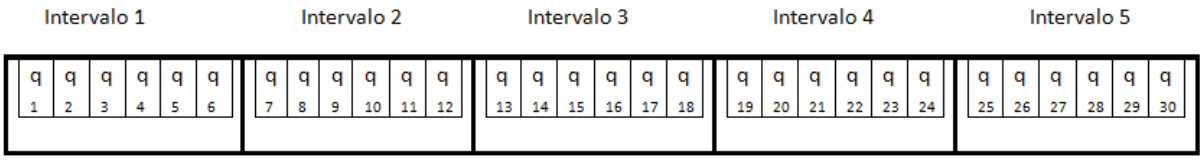


Ilustración 7: Rangos del conjunto de consultas QD

Por ejemplo, a partir del intervalo 1 se seleccionarán aleatoriamente consultas. Cuando se selecciona una consulta se inserta en la cola de prioridad, mediante la función insertar() que verifica si la consulta está presente en la cola de prioridad, si no está presente se inserta la consulta, junto a sus documentos relevantes y su carácter de prioridad (prio) con valor 1, al final de la cola de prioridad. En caso que la consulta esté presente en la cola de prioridad se procederá al aumento de prio, posteriormente se ordena la cola de prioridad, mediante la función ordenar(), de acuerdo al valor máximo de prioridad. La Ilustración 8 muestra el intervalo 1, en el caso 8.a) se realiza la inserción de una nueva consulta seleccionada q_2, agregando al final de la lista el elemento junto a su

carácter de prioridad con valor 1. Por otra parte en el caso 8.b) se selecciona la consulta q_1, la que se verifica que se encuentra almacenada en la cola de prioridad, por lo tanto se realiza la actualización del carácter de prioridad de la consulta seleccionada, aquí no hay necesidad de ordenar la lista ya que el elemento que sigue tiene valor prio menor.

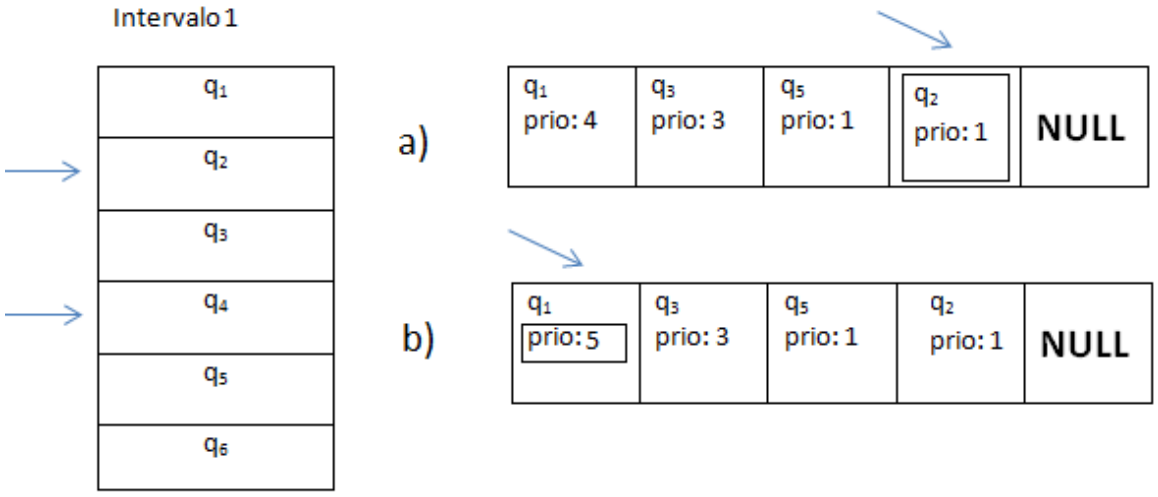


Ilustración 8: Intervalo para la creación del conjunto de consultas ejecutadas con anterioridad

Supongamos que el siguiente elemento seleccionado sea nuevamente la consulta q_2, se realiza la actualización de prio con valor 2. Posteriormente la función ordenar() verifica el valor de prio de la consulta en la posición anterior a q_2, en este caso la consulta q_5, que tiene valor de prioridad igual a 1, siendo menor al valor prio de la consulta q_2. Para ordenar la lista se realiza un cambio de posición, obtiendo una lista de elementos ordenados en base a valores máximos. Por lo tanto, la nueva lista quedaría organizada de la siguiente manera: q_1, q_3, q_2 y q_5, con carácter de prioridad 5, 3, 2 y 1 respectivamente. Una vez seleccionadas las consultas de cada intervalo, a través del loop de 100 iteraciones, se retornará el conjunto de consultas más consultadas con anterioridad CP.

Algorithm 1 Creación Cola de Prioridad

Input: QD Conjunto de consultas junto a sus documentos relevantes, $NQueries$ número total de consultas, $range$ rango que divide en secciones el conjunto QD.

Output: CP es un conjunto de consultas ejecutadas con anterioridad junto a sus documentos relevantes, $prio$ representa el número de veces que fue consultada una consulta.

$loop \leftarrow 100;$
$final \leftarrow range + 1;$
$Nelements \leftarrow NQueries/range;$
$prio \leftarrow 0;$
$CP \leftarrow NULL;$
for $j = 0; j < range; j++$ **do**
  for $i = 0; i < loop; i++$ **do**
    repeat
    $exp\_rv \leftarrow expon((1.0/lambda));$
    $final \leftarrow (final * NQueries)/range;$
    $final \leftarrow ceil(exp\_rv);$
    until $(final < range);$
    $final \leftarrow (final + (Nelements * j));$
    $insertar(CP, headQD, final, prio);$
    $ordenarPrioridad(CP);$
  end for
end for
return $(CP);$

Ilustración 9: Pseudocódigo Algoritmo - Creación Cola de Prioridad.

3.4. Resumen

En este Capítulo se describió el método para la simulación de un SRI, utilizando un enfoque se basa en la reutilización de documentos relevantes recuperados de la consulta anterior más similar o repetida, cuando se envía una nueva consulta en un SRI. Además, se proporcionan los algoritmos utilizados para la representación del SRI. En el siguiente Capítulo se profundizara en la construcción del escenario experimental, detallando el tamaño del conjunto de documentos y conjunto de consultas, como también los parámetros a utilizar para la creación de éstos.

CAPÍTULO IV

4. DISEÑO EXPERIMENTAL

4.1. Escenario Experimental

El entorno experimental se establece como sigue: la longitud de un término | T |, está entre 3 y 7. El número de término es de 3500 en cada experimento. El número de términos que compone cada documento puede ser entre 15 y 30 términos. El número de tópicos (temas) utilizados en cada experimento es 7. Cada tema está formado por 500 términos. Cuando se construye un documento, los términos de otros temas se eligen utilizando la distribución exponencial. Por lo tanto, la mayoría de las palabras, que componen un documento se eligen de un tema específico.

El número del conjunto de documentos D utilizado en cada experimento es de 700, 1400, 2100, 2800 y 3500. Por otro lado, los la cantidad de términos que conforman una consulta varía entre 3 y 8. Los términos de una consulta son seleccionados de un documento en particular. Ambos, términos y documentos, fueron seleccionados utilizando distribución uniforme para crear las consultas pasadas. Para construir las consultas anteriores se seleccionó la mitad del número de consultas en cada experimento. Para crear las consultas ejecutadas con anterioridad, se eligen bajo distribución exponencial. El número del conjunto de consultas Q en cada experimento es de 30, 60, 90, 120, 150, 180 y 210.

Con el fin de simular el juicio de usuario sobre los documentos recuperados de un consulta, se implementó la distribución de Zeta a los primeros 10 documentos relevantes, con parámetros S=2, S=3 y S=4, para cada experimento. Por otra parte, con el fin almacenar consultas similares e idénticas ejecutadas con anterioridad por un usuario, se seleccionó las consultas del conjunto QD bajo distribución Exponencial, con el propósito de representar las consultas más consultadas en el sistema con anterioridad. Cada experimento reúne dos escenarios diferentes de distribución Exponencial con parámetro λ=1,0 y λ=1,5. Eventualmente, La distribución Exponencial se aplica de la siguiente manera: primero se define los intervalos a partir de un rango. El rango se define dependiendo del número de consultas del experimento. Para 30, 60, 90 y 120 consultas el rango es de 5; y para 150, 180 y 210 consultas el rango es de 10.

4.2. Resultados Experimentales

Los resultados preliminares informan el promedio de P@10 (los primeros 10 documentos relevantes) de las consuntas, esto es para todos los conjuntos de consultas Q.

Experimento 1. Se utilizó una distribución exponencial con parámetro $\lambda = 1{,}0$ para construir los conjuntos de datos D y para crear el conjunto QD. Además se utilizó distribución Zeta con parámetro S=2, S=3 y S=4 para proporcionar el juicio de usuario.

Experimento 2. Se utilizó una distribución exponencial con parámetro $\lambda = 1{,}5$ para construir el conjunto de datos D y para crear el conjunto QD. Además se utilizó distribución Zeta con parámetro S=2, S=3 y S=4 para proporcionar el juicio de usuario.

Por otro lado, el objetivo de utilizar los parámetros de distribución Zeta es para proporcionar el juicio de usuario en el conjunto respuesta y así analizar cómo influye en el promedio de documentos recuperados (P@10). Cuando se incrementa el parámetro S, es posible notar que tanto el P@10 en la recuperación de consultas similares como nuestro enfoque (consultas ejecutadas con anterioridad) disminuyen.

Es posible notar que no hay una tendencia radical en la diferencia del promedio para P @ 10 en la recuperación de los documentos relevantes para consultas ejecutadas con anterioridad (Algoritmo 2.1) y para la recuperación de búsquedas pasadas (Algoritmo 2.0), cuando el número de documentos se incrementa. Del mismo modo, el aumento en el número de consultas en estos experimentos, no debe afectar a los resultados finales. Las razones son las siguientes:

- Cada consulta anterior se construye a partir de un documento, mientras que la nueva consulta se obtiene de una consulta anterior. Significa, para cada consulta anterior hay una nueva consulta (similar o repetida). Además, la intersección entre las consultas anteriores es vacía. Por lo tanto, aumentar el número de consultas no tiene impacto en los resultados finales, ya que no influye en la creación de las nuevas consultas.

- Al igual que el anterior, el aumento del número de documentos no afecta a los resultados finales, ya que la pertinencia de los documentos (juicio de usuario) se obtiene aplicando la distribución Zeta en la lista de documentos recuperados (P @ 10), que se obtienen de la coincidencia entre la consulta y los documentos.

A partir de lo anterior en las siguientes Tablas se compara ambos enfoques con los siguientes parámetros de distribución Zeta y Exponencial:

- **Distribución Exponencial λ = 1,0 y distribución Zeta S =2**

En la Tabla 2 y Tabla 3 se ve el contraste de resultados con las consultas identificadas como repetidas de nuestro enfoque (Algoritmo 2.1) y de un enfoque tradicional (Algoritmo 2.0) para los distintos números de consultas que se utilizaron en los experimentos, para esto se tomó en cuenta el valor máximo de precisión obtenida en cada grupo de documentos y consultas.

	Colección de Documentos						
Número de Consultas	700 D	1400 D	2100 D	2800 D	3500 D	Máximo	Promedio
30Q	0,100000	0,096889	0,100000	0,099000	0,099000	**0,100000**	**0,098978**
60Q	0,096889	0,096889	0,093528	0,096889	0,099000	**0,099000**	**0,096639**
90Q	0,096889	0,099000	0,096889	0,096889	0,099000	**0,099000**	**0,097733**
120Q	0,096889	0,096889	0,096889	0,100000	0,096889	**0,100000**	**0,097511**
150Q	0,096889	0,099000	0,099000	0,096889	0,099000	**0,099000**	**0,098156**
180Q	0,099000	0,099000	0,093528	0,096889	0,093528	**0,099000**	**0,096389**
210Q	0,096889	0,093528	0,093528	0,096889	0,088738	**0,096889**	**0,093914**
Máximo	**0,100000**	**0,099000**	**0,100000**	**0,100000**	**0,099000**	0,100000	
Promedio	**0,097635**	**0,097314**	**0,096195**	**0,097635**	**0,096451**		**0,097046**

Tabla 2: Precisión máxima y promedio en cada serie de consultas que fueron identificadas como repetidas utilizando el Algoritmo 2.1 con parámetros S=2, λ=1,0

	Colección de Documentos						
Número de Consultas	700 D	1400 D	2100 D	2800 D	3500 D	Máximo	Promedio
30Q	0,096889	0,099000	0,099000	0,096889	0,099000	0,099000	0,098156
60Q	0,099000	0,093528	0,093528	0,096889	0,096889	**0,099000**	**0,095967**
90Q	0,096889	0,099000	0,099000	0,099000	0,096889	0,099000	0,098156
120Q	0,100000	0,096889	0,099000	0,096889	0,100000	**0,100000**	**0,098556**
150Q	0,093528	0,096889	0,099000	0,096889	0,099000	0,099000	0,097061
180Q	0,096889	0,093528	0,093528	0,099000	0,093528	**0,099000**	**0,095295**
210Q	0,099000	0,093528	0,093528	0,093528	0,096889	0,099000	0,095295
Máximo	0,100000	0,099000	0,099000	0,099000	0,100000	0,100000	
Promedio	**0,097456**	**0,096052**	**0,096655**	**0,097012**	**0,097456**		**0,096926**

Tabla 3: Precisión máxima y promedio en cada serie de consultas repetidas utilizando el Algoritmo 2.0 con parámetros S=2, λ=1,0

Al igual que en las tablas anteriores, en la Tabla 4 y Tabla 5 se ve el contraste de resultados, para las consultas identificadas como similares, de nuestro enfoque (Algoritmo 2.1) y de un enfoque tradicional (Algoritmo 2.0).

	Colección de Documentos						
Número de Consultas	700 D	1400 D	2100 D	2800 D	3500 D	Máximo	Promedio
30Q	0,100000	0,100000	0,100000	0,100000	0,100000	**0,100000**	**0,100000**
60Q	0,100000	0,100000	0,100000	0,100000	0,100000	**0,100000**	**0,100000**
90Q	0,100000	0,100000	0,100000	0,100000	0,100000	**0,100000**	**0,100000**
120Q	0,100000	0,100000	0,100000	0,100000	0,100000	**0,100000**	**0,100000**
150Q	0,100000	0,100000	0,100000	0,100000	0,100000	**0,100000**	**0,100000**
180Q	0,100000	0,100000	0,100000	0,100000	0,100000	**0,100000**	**0,100000**
210Q	0,100000	0,100000	0,100000	0,100000	0,100000	**0,100000**	**0,100000**
Máximo	**0,100000**	**0,100000**	**0,100000**	**0,100000**	**0,100000**	**0,100000**	
Promedio	**0,100000**	**0,100000**	**0,100000**	**0,100000**	**0,100000**		**0,100000**

Tabla 4: Precisión máxima y promedio en cada serie de consultas que fueron identificadas como similares utilizando el Algoritmo 2.1 con parámetros S=2, λ =1,0

	Colección de Documentos						
Número de Consultas	700 D	1400 D	2100 D	2800 D	3500 D	Máximo	Promedio
30Q	0,096889	0,096889	0,099000	0,099000	0,099000	0,099000	0,098156
60Q	0,096889	0,096889	0,093528	0,096889	0,096889	**0,096889**	**0,096217**
90Q	0,096889	0,096889	0,093528	0,096889	0,099000	0,099000	0,096639
120Q	0,096889	0,099000	0,096889	0,100000	0,096889	**0,100000**	**0,097933**
150Q	0,096889	0,099000	0,099000	0,099000	0,099000	0,099000	0,098578
180Q	0,099000	0,099000	0,093528	0,096889	0,093528	**0,099000**	**0,096389**
210Q	0,096889	0,093528	0,093528	0,093528	0,088738	0,096889	0,093242
Máximo	0,099000	0,099000	0,099000	0,100000	0,099000	0,100000	
Promedio	**0,097191**	**0,097314**	**0,095572**	**0,097456**	**0,096149**		**0,096736**

Tabla 5: Precisión máxima y promedio en cada serie de consultas similares utilizando el Algoritmo 2.0 con parámetros S=2, λ =1,0

- **Distribución Exponencial λ = 1,5 y distribución Zeta S =2**

En la Tabla 6 y Tabla 7 se ve el contraste de resultados (con las consultas identificadas como repetidas) de nuestro enfoque (Algoritmo 2.1) y de un enfoque tradicional (Algoritmo 2.0) para los distintos números de consultas que se utilizaron en los experimentos, para esto se tomó en cuenta el valor máximo de precisión obtenida en cada grupo de documentos y consultas.

	Colección de Documentos						
Número de Consultas	700 D	1400 D	2100 D	2800 D	3500 D	Máximo	Promedio
30Q	0,099000	0,099000	0,099000	0,099000	0,100000	**0,100000**	**0,099200**
60Q	0,096889	0,099000	0,093528	0,099000	0,099000	**0,099000**	**0,097483**
90Q	0,096889	0,096889	0,099000	0,100000	0,099000	**0,100000**	**0,098356**
120Q	0,093528	0,096889	0,100000	0,096889	0,100000	**0,100000**	**0,097461**
150Q	0,096889	0,093528	0,096889	0,096889	0,099000	**0,099000**	**0,096639**
180Q	0,099000	0,096889	0,099000	0,088738	0,096889	**0,099000**	**0,096103**
210Q	0,099000	0,093528	0,093528	0,099000	0,099000	**0,099000**	**0,096811**
Máximo	**0,099000**	**0,099000**	**0,100000**	**0,100000**	**0,100000**	0,100000	
Promedio	**0,097314**	**0,096532**	**0,097278**	**0,097074**	**0,098984**		**0,097436**

Tabla 6: Precisión máxima y promedio en cada serie de consultas que fueron identificadas como repetidas utilizando el Algoritmo 2.1 con parámetros S=2, λ =1,5.

	Colección de Documentos						
Número de Consultas	700 D	1400 D	2100 D	2800 D	3500 D	Máximo	Promedio
30Q	0,099000	0,099000	0,099000	0,099000	0,100000	0,100000	0,099200
60Q	0,100000	0,100000	0,093528	0,096889	0,093528	**0,100000**	**0,096789**
90Q	0,096889	0,099000	0,099000	0,096889	0,100000	0,100000	0,098356
120Q	0,096889	0,096889	0,096889	0,099000	0,093528	**0,099000**	**0,096639**
150Q	0,093528	0,096889	0,096889	0,099000	0,096889	0,099000	0,096639
180Q	0,096889	0,093528	0,093528	0,088738	0,093528	**0,096889**	**0,093242**
210Q	0,096889	0,093528	0,093528	0,088738	0,093528	0,096889	0,093242
Máximo	0,100000	0,100000	0,099000	0,099000	0,100000	0,100000	
Promedio	**0,097155**	**0,096976**	**0,096052**	**0,095465**	**0,095857**		**0,096301**

Tabla 7: Precisión máxima y promedio en cada serie de consultas repetidas utilizando el Algoritmo 2.0 con parámetros S=2, λ =1,5.

Al igual que el anterior caso, en la Tabla 8 y Tabla 9 se ve el contraste de resultados, para las consultas identificadas como similares, de nuestro enfoque (Algoritmo 2.1) y de un enfoque tradicional (Algoritmo 2.0).

	Colección de Documentos						
Número de Consultas	700 D	1400 D	2100 D	2800 D	3500 D	Máximo	Promedio
30Q	0,100000	0,100000	0,100000	0,100000	0,100000	**0,100000**	**0,100000**
60Q	0,100000	0,100000	0,100000	0,100000	0,100000	**0,100000**	**0,100000**
90Q	0,100000	0,100000	0,100000	0,100000	0,100000	**0,100000**	**0,100000**
120Q	0,100000	0,100000	0,100000	0,100000	0,100000	**0,100000**	**0,100000**
150Q	0,100000	0,100000	0,100000	0,100000	0,100000	**0,100000**	**0,100000**
180Q	0,100000	0,100000	0,100000	0,100000	0,100000	**0,100000**	**0,100000**
210Q	0,100000	0,100000	0,100000	0,100000	0,100000	**0,100000**	**0,100000**
Máximo	**0,100000**	**0,100000**	**0,100000**	**0,100000**	**0,100000**	**0,100000**	
Promedio	**0,100000**	**0,100000**	**0,100000**	**0,100000**	**0,100000**		**0,100000**

Tabla 8: Precisión máxima y promedio en cada serie de consultas que fueron identificadas como similares utilizando el Algoritmo 2.1 con parámetros S=2, λ =1,5.

	Colección de Documentos						
Número de Consultas	700 D	1400 D	2100 D	2800 D	3500 D	Máximo	Promedio
30Q	0,099000	0,099000	0,099000	0,099000	0,100000	0,100000	0,099200
60Q	0,096889	0,099000	0,093528	0,099000	0,099000	**0,099000**	**0,097483**
90Q	0,096889	0,096889	0,099000	0,100000	0,096889	0,100000	0,097933
120Q	0,096889	0,096889	0,100000	0,096889	0,100000	**0,100000**	**0,098133**
150Q	0,096889	0,093528	0,099000	0,096889	0,099000	0,099000	0,097061
180Q	0,099000	0,096889	0,099000	0,088738	0,096889	**0,099000**	**0,096103**
210Q	0,099000	0,096889	0,096889	0,099000	0,099000	0,099000	0,098156
Máximo	0,099000	0,099000	0,100000	0,100000	0,100000	0,100000	
Promedio	**0,097794**	**0,097012**	**0,098060**	**0,097074**	**0,098683**		**0,097724**

Tabla 9: Precisión máxima y promedio en cada serie de consultas similares utilizando el Algoritmo 2.0 con parámetros S=2, λ 1,5.

- **Distribución Exponencial λ = 1,0 y distribución Zeta S =3**

En la Tabla 10 y Tabla 11 se ve el contraste de resultados (con las consultas identificadas como repetidas) de nuestro enfoque (Algoritmo 2.1) y de un enfoque tradicional (Algoritmo 2.0) para los distintos números de consultas que se utilizaron en los experimentos, para esto se tomó en cuenta el valor máximo de precisión obtenida en cada grupo de documentos y consultas.

	Colección de Documentos						
Número de Consultas	700 D	1400 D	2100 D	2800 D	3500 D	Máximo	Promedio
30Q	0,088738	0,088738	0,093528	0,088738	0,088738	**0,093528**	**0,089696**
60Q	0,082282	0,088738	0,088738	0,088738	0,088738	**0,088738**	**0,087447**
90Q	0,093528	0,088738	0,093528	0,093528	0,096889	**0,096889**	**0,093242**
120Q	0,093528	0,093528	0,096889	0,096889	0,093528	**0,096889**	**0,094872**
150Q	0,093528	0,088738	0,093528	0,088738	0,093528	**0,093528**	**0,091612**
180Q	0,093528	0,093528	0,088738	0,088738	0,088738	**0,093528**	**0,090654**
210Q	0,088738	0,088738	0,088738	0,088738	0,082282	**0,088738**	**0,087447**
Máximo	**0,093528**	**0,093528**	**0,096889**	**0,096889**	**0,096889**	0,096889	
Promedio	**0,090553**	**0,090107**	**0,091955**	**0,090587**	**0,090349**		**0,090710**

Tabla 10: Precisión máxima y promedio en cada serie de consultas que fueron identificadas como repetidas utilizando el Algoritmo 2.1 con parámetros S=3, λ =1,0

	Colección de Documentos						
Número de Consultas	700 D	1400 D	2100 D	2800 D	3500 D	Máximo	Promedio
30Q	0,093528	0,093528	0,093528	0,093528	0,093528	0,093528	0,093528
60Q	0,088738	0,088738	0,093528	0,088738	0,088738	**0,093528**	**0,089696**
90Q	0,093528	0,096889	0,096889	0,096889	0,093528	0,096889	0,095545
120Q	0,093528	0,093528	0,096889	0,093528	0,096889	**0,096889**	**0,094872**
150Q	0,088738	0,088738	0,093528	0,093528	0,093528	0,093528	0,091612
180Q	0,093528	0,088738	0,088738	0,093528	0,088738	**0,093528**	**0,090654**
210Q	0,093528	0,088738	0,093528	0,088738	0,088738	0,093528	0,090654
Máximo	0,093528	0,096889	0,096889	0,096889	0,096889	0,096889	
Promedio	**0,092159**	**0,091271**	**0,093804**	**0,09264**	**0,091955**		**0,092366**

Tabla 11: Precisión máxima y promedio en cada serie de consultas repetidas utilizando el Algoritmo 2.0 con parámetros S=3, λ =1,0

Al igual que el anterior en la Tabla 12 y Tabla 13 se ve el contraste de resultados, para las consultas identificadas como similares, de nuestro enfoque (Algoritmo 2.1) y de un enfoque tradicional (Algoritmo 2.0).

	Colección de Documentos						
Número de Consultas	700 D	1400 D	2100 D	2800 D	3500 D	Máximo	Promedio
30Q	0,100000	0,100000	0,100000	0,100000	0,100000	**0,100000**	**0,100000**
60Q	0,096889	0,096889	0,099000	0,100000	0,099000	**0,100000**	**0,098356**
90Q	0,100000	0,100000	0,100000	0,100000	0,100000	**0,100000**	**0,100000**
120Q	0,100000	0,100000	0,100000	0,100000	0,100000	**0,100000**	**0,100000**
150Q	0,100000	0,100000	0,100000	0,100000	0,100000	**0,100000**	**0,100000**
180Q	0,100000	0,099000	0,100000	0,100000	0,099000	**0,100000**	**0,099600**
210Q	0,099000	0,100000	0,100000	0,100000	0,100000	**0,100000**	**0,099800**
Máximo	**0,100000**	**0,100000**	**0,100000**	**0,100000**	**0,100000**	0,100000	
Promedio	**0,099413**	**0,099413**	**0,099857**	**0,100000**	**0,099714**		**0,099679**

Tabla 12: Precisión máxima y promedio en cada serie de consultas que fueron identificadas como similares utilizando el Algoritmo 2.1 con parámetros S=3, λ =1,0

	Colección de Documentos						
Número de Consultas	700 D	1400 D	2100 D	2800 D	3500 D	Máximo	Promedio
30Q	0,088738	0,088738	0,088738	0,088738	0,088738	0,088738	0,088738
60Q	0,082282	0,088738	0,088738	0,088738	0,082282	**0,088738**	**0,086156**
90Q	0,088738	0,088738	0,088738	0,093528	0,093528	0,093528	0,090654
120Q	0,093528	0,093528	0,096889	0,096889	0,093528	**0,096889**	**0,094872**
150Q	0,093528	0,093528	0,093528	0,088738	0,096889	0,096889	0,093242
180Q	0,093528	0,093528	0,088738	0,088738	0,088738	**0,093528**	**0,090654**
210Q	0,088738	0,088738	0,088738	0,088738	0,088738	0,088738	0,088738
Máximo	0,093528	0,093528	0,096889	0,096889	0,096889	0,096889	
Promedio	**0,089869**	**0,090791**	**0,090587**	**0,090587**	**0,090349**		**0,090436**

Tabla 13: Precisión máxima y promedio en cada serie de consultas similares utilizando el Algoritmo 2.0 con parámetros S=3, λ =1,0

- **Distribución Exponencial λ = 1,5 y distribución Zeta S =3**

En la Tabla 14 y Tabla 15 se ve el contraste de resultados (con las consultas identificadas como repetidas) de nuestro enfoque (Algoritmo 2.1) y de un enfoque tradicional (Algoritmo 2.0) para los distintos números de consultas que se utilizaron en los experimentos, para esto se tomó en cuenta el valor máximo de precisión obtenida en cada grupo de documentos y consultas.

	Colección de Documentos						
Número de Consultas	700 D	1400 D	2100 D	2800 D	3500 D	Máximo	Promedio
30Q	0,088738	0,088738	0,093528	0,093528	0,093528	**0,093528**	**0,091612**
60Q	0,093528	0,093528	0,088738	0,088738	0,093528	**0,093528**	**0,091612**
90Q	0,093528	0,093528	0,096889	0,093528	0,096889	**0,096889**	**0,094872**
120Q	0,093528	0,096889	0,093528	0,088738	0,096889	**0,096889**	**0,093914**
150Q	0,088738	0,093528	0,088738	0,093528	0,093528	**0,093528**	**0,091612**
180Q	0,093528	0,088738	0,093528	0,088738	0,093528	**0,093528**	**0,091612**
210Q	0,096889	0,088738	0,088738	0,096889	0,088738	**0,096889**	**0,091998**
Máximo	**0,096889**	**0,096889**	**0,096889**	**0,096889**	**0,096889**	0,096889	
Promedio	**0,09264**	**0,091955**	**0,091955**	**0,091955**	**0,093804**		**0,092462**

Tabla 14: Precisión máxima y promedio en cada serie de consultas que fueron identificadas como repetidas utilizando el Algoritmo 2.1 con parámetros S=3, λ =1,5.

	Colección de Documentos						
Número de Consultas	700 D	1400 D	2100 D	2800 D	3500 D	Máximo	Promedio
30Q	0,088738	0,093528	0,093528	0,093528	0,093528	0,093528	0,09257
60Q	0,093528	0,093528	0,088738	0,088738	0,082282	**0,093528**	**0,089363**
90Q	0,093528	0,093528	0,093528	0,088738	0,096889	0,096889	0,093242
120Q	0,093528	0,096889	0,093528	0,093528	0,093528	**0,096889**	**0,0942**
150Q	0,096889	0,096889	0,088738	0,093528	0,088738	0,096889	0,092956
180Q	0,093528	0,088738	0,093528	0,088738	0,093528	**0,093528**	**0,091612**
210Q	0,088738	0,088738	0,088738	0,093528	0,088738	0,093528	0,089696
Máximo	0,096889	0,096889	0,093528	0,093528	0,096889	0,096889	
Promedio	**0,09264**	**0,09312**	**0,091475**	**0,091475**	**0,091033**		**0,091949**

Tabla 15: Precisión máxima y promedio en cada serie de consultas repetidas utilizando el Algoritmo 2.0 con parámetros S=3, λ =1,5.

Al igual que el anterior caso, en la Tabla 16 y Tabla 17 se ve el contraste de resultados, para las consultas identificadas como similares, de nuestro enfoque (Algoritmo 2.1) y de un enfoque tradicional (Algoritmo 2.0)

	Colección de Documentos						
Número de Consultas	700 D	1400 D	2100 D	2800 D	3500 D	Máximo	Promedio
30Q	0,099000	0,099000	0,100000	0,100000	0,100000	**0,100000**	**0,099600**
60Q	0,099000	0,100000	0,096889	0,099000	0,099000	**0,100000**	**0,098778**
90Q	0,100000	0,100000	0,100000	0,100000	0,100000	**0,100000**	**0,100000**
120Q	0,100000	0,100000	0,100000	0,100000	0,100000	**0,100000**	**0,100000**
150Q	0,100000	0,100000	0,100000	0,100000	0,100000	**0,100000**	**0,100000**
180Q	0,100000	0,100000	0,099000	0,100000	0,099000	**0,100000**	**0,099600**
210Q	0,100000	0,100000	0,100000	0,100000	0,100000	**0,100000**	**0,100000**
Máximo	**0,100000**	**0,100000**	**0,100000**	**0,100000**	**0,100000**	**0,100000**	
Promedio	**0,099714**	**0,099857**	**0,099413**	**0,099857**	**0,099714**		**0,099711**

Tabla 16: Precisión máxima y promedio en cada serie de consultas que fueron identificadas como similares utilizando el Algoritmo 2.1 con parámetros S=3, λ =1,5.

	Colección de Documentos						
Número de Consultas	700 D	1400 D	2100 D	2800 D	3500 D	Máximo	Promedio
30Q	0,088738	0,088738	0,093528	0,093528	0,093528	0,093528	0,091612
60Q	0,093528	0,093528	0,088738	0,088738	0,093528	**0,093528**	**0,091612**
90Q	0,088738	0,093528	0,096889	0,093528	0,093528	0,096889	0,093242
120Q	0,093528	0,096889	0,093528	0,093528	0,096889	**0,096889**	**0,094872**
150Q	0,082282	0,093528	0,088738	0,088738	0,093528	0,093528	0,089363
180Q	0,093528	0,088738	0,093528	0,088738	0,093528	**0,093528**	**0,091612**
210Q	0,096889	0,088738	0,088738	0,096889	0,096889	0,096889	0,093629
Máximo	0,096889	0,096889	0,096889	0,096889	0,096889	0,096889	
Promedio	**0,091033**	**0,091955**	**0,091955**	**0,091955**	**0,094488**		**0,092277**

Tabla 17: Precisión máxima y promedio en cada serie de consultas similares utilizando el Algoritmo 2.0 con parámetros S=3, λ =1,5.

- **Distribución Exponencial λ = 1,0 y distribución Zeta S =4**

En la Tabla 18 y Tabla 19 se ve el contraste de resultados (con las consultas identificadas como repetidas) de nuestro enfoque (Algoritmo 2.1) y de un enfoque tradicional (Algoritmo 2.0) para los distintos números de consultas que se utilizaron en los experimentos, para esto se tomó en cuenta el valor máximo de precisión obtenida en cada grupo de documentos y consultas.

	Colección de Documentos						
Número de Consultas	700 D	1400 D	2100 D	2800 D	3500 D	Máximo	Promedio
30Q	0,082282	0,082282	0,082282	0,082282	0,082282	**0,082282**	**0,082282**
60Q	0,073825	0,082282	0,082282	0,082282	0,082282	**0,082282**	**0,080591**
90Q	0,073825	0,073825	0,062869	0,073825	0,073825	**0,073825**	**0,071634**
120Q	0,088738	0,088738	0,088738	0,088738	0,088738	**0,088738**	**0,088738**
150Q	0,088738	0,088738	0,088738	0,088738	0,088738	**0,088738**	**0,088738**
180Q	0,088738	0,088738	0,088738	0,088738	0,088738	**0,088738**	**0,088738**
210Q	0,082282	0,088738	0,082282	0,082282	0,073825	**0,088738**	**0,081882**
Máximo	**0,088738**	**0,088738**	**0,088738**	**0,088738**	**0,088738**	0,088738	
Promedio	**0,082633**	**0,084763**	**0,082276**	**0,083841**	**0,082633**		**0,083229**

Tabla 18: Precisión máxima y promedio en cada serie de consultas que fueron identificadas como repetidas utilizando el Algoritmo 2.1 con parámetros S=4, λ =,0.

	Colección de Documentos						
Número de Consultas	700 D	1400 D	2100 D	2800 D	3500 D	Máximo	Promedio
30Q	0,082282	0,082282	0,082282	0,082282	0,088738	0,088738	0,083573
60Q	0,082282	0,082282	0,082282	0,073825	0,073825	**0,082282**	**0,078899**
90Q	0,073825	0,073825	0,073825	0,082282	0,062869	0,082282	0,073325
120Q	0,093528	0,088738	0,088738	0,088738	0,093528	**0,093528**	**0,090654**
150Q	0,082282	0,088738	0,093528	0,088738	0,088738	0,093528	0,088405
180Q	0,088738	0,082282	0,082282	0,082282	0,088738	**0,088738**	**0,084864**
210Q	0,088738	0,082282	0,082282	0,082282	0,082282	0,088738	0,083573
Máximo	0,093528	0,088738	0,093528	0,088738	0,093528	0,093528	
Promedio	**0,084525**	**0,082918**	**0,083603**	**0,082918**	**0,082674**		**0,083328**

Tabla 19: Precisión máxima y promedio en cada serie de consultas repetidas utilizando el Algoritmo 2.0 con parámetros S=4, λ =1,0.

Al igual que el anterior caso, en la Tabla 20 y Tabla 21 se ve el contraste de resultados, para las consultas identificadas como similares, de nuestro enfoque (Algoritmo 2.1) y de un enfoque tradicional (Algoritmo 2.0).

	Colección de Documentos						
Número de Consultas	700 D	1400 D	2100 D	2800 D	3500 D	Máximo	Promedio
30Q	0,093528	0,093528	0,096889	0,093528	0,093528	**0,096889**	**0,094200**
60Q	0,088738	0,088738	0,088738	0,093528	0,096889	**0,096889**	**0,091326**
90Q	0,088738	0,088738	0,093528	0,088738	0,088738	**0,093528**	**0,089696**
120Q	0,096889	0,099000	0,100000	0,099000	0,099000	**0,100000**	**0,098778**
150Q	0,096889	0,096889	0,096889	0,100000	0,099000	**0,100000**	**0,097933**
180Q	0,093528	0,096889	0,099000	0,096889	0,096889	**0,099000**	**0,096639**
210Q	0,093528	0,096889	0,099000	0,096889	0,093528	**0,099000**	**0,095967**
Máximo	**0,096889**	**0,099000**	**0,100000**	**0,100000**	**0,099000**	0,100000	
Promedio	**0,093120**	**0,094382**	**0,096292**	**0,095510**	**0,095367**		**0,094934**

Tabla 20: Precisión máxima y promedio en cada serie de consultas que fueron identificadas como similares utilizando el Algoritmo 2.1 con parámetros S=4, λ=1,0.

	Colección de Documentos						
Número de Consultas	700 D	1400 D	2100 D	2800 D	3500 D	Máximo	Promedio
30Q	0,082282	0,082282	0,082282	0,082282	0,082282	0,082282	0,082282
60Q	0,073825	0,082282	0,082282	0,082282	0,073825	**0,082282**	**0,078899**
90Q	0,073825	0,073825	0,062869	0,073825	0,073825	0,073825	0,071634
120Q	0,088738	0,088738	0,088738	0,088738	0,088738	**0,088738**	**0,088738**
150Q	0,088738	0,088738	0,088738	0,082282	0,088738	0,088738	0,087447
180Q	0,088738	0,088738	0,082282	0,088738	0,073825	**0,088738**	**0,084464**
210Q	0,082282	0,088738	0,088738	0,082282	0,082282	0,088738	0,084864
Máximo	0,088738	0,088738	0,088738	0,088738	0,088738	0,088738	
Promedio	**0,082633**	**0,084763**	**0,082276**	**0,082918**	**0,080502**		**0,082618**

Tabla 21: Precisión máxima y promedio en cada serie de consultas similares utilizando el Algoritmo 2.0 con parámetros S=4, λ =1,0.

- **Distribución Exponencial λ = 1,5 y distribución Zeta S =4**

En la Tabla 22 y Tabla 23 se ve el contraste de resultados (con las consultas identificadas como repetidas) de nuestro enfoque (Algoritmo 2.1) y de un enfoque tradicional (Algoritmo 2.0) para los distintos números de consultas que se utilizaron en los experimentos, para esto se tomó en cuenta el valor máximo de precisión obtenida en cada grupo de documentos y consultas.

	Colección de Documentos						
Número de Consultas	700 D	1400 D	2100 D	2800 D	3500 D	Máximo	Promedio
30Q	0,082282	0,082282	0,082282	0,082282	0,082282	**0,082282**	**0,082282**
60Q	0,082282	0,082282	0,073825	0,073825	0,082282	**0,082282**	**0,078899**
90Q	0,073825	0,073825	0,073825	0,073825	0,073825	**0,073825**	**0,073825**
120Q	0,088738	0,088738	0,093528	0,082282	0,093528	**0,093528**	**0,089363**
150Q	0,082282	0,088738	0,082282	0,088738	0,088738	**0,088738**	**0,086156**
180Q	0,088738	0,082282	0,082282	0,082282	0,082282	**0,088738**	**0,083573**
210Q	0,088738	0,082282	0,082282	0,088738	0,082282	**0,088738**	**0,084864**
Máximo	**0,088738**	**0,088738**	**0,093528**	**0,088738**	**0,093528**	0,093528	
Promedio	**0,083841**	**0,082918**	**0,081472**	**0,081710**	**0,083603**		**0,082709**

Tabla 22: Precisión máxima y promedio en cada serie de consultas que fueron identificadas como repetidas utilizando el Algoritmo 2.1 con parámetros S=4, λ=1,5.

	Colección de Documentos						
Número de Consultas	700 D	1400 D	2100 D	2800 D	3500 D	Máximo	Promedio
30Q	0,082282	0,082282	0,082282	0,082282	0,082282	0,082282	0,082282
60Q	0,082282	0,082282	0,073825	0,082282	0,073825	**0,082282**	**0,078899**
90Q	0,062869	0,062869	0,073825	0,073825	0,073825	0,073825	0,069443
120Q	0,088738	0,088738	0,088738	0,088738	0,088738	**0,088738**	**0,088738**
150Q	0,088738	0,088738	0,082282	0,088738	0,082282	0,088738	0,086156
180Q	0,088738	0,073825	0,082282	0,082282	0,082282	**0,088738**	**0,081882**
210Q	0,088738	0,082282	0,073825	0,082282	0,082282	0,088738	0,081882
Máximo	0,088738	0,088738	0,088738	0,088738	0,088738	0,088738	
Promedio	**0,083198**	**0,080145**	**0,07958**	**0,082918**	**0,080788**		**0,081326**

Tabla 23: Precisión máxima y promedio en cada serie de consultas repetidas utilizando el Algoritmo 2.0 con parámetros S=4, λ=1,5.

Al igual que el anterior caso, en la Tabla 24 y Tabla 25 se ve el contraste de resultados, para las consultas identificadas como similares, de nuestro enfoque (Algoritmo 2.1) y de un enfoque tradicional (Algoritmo 2.0).

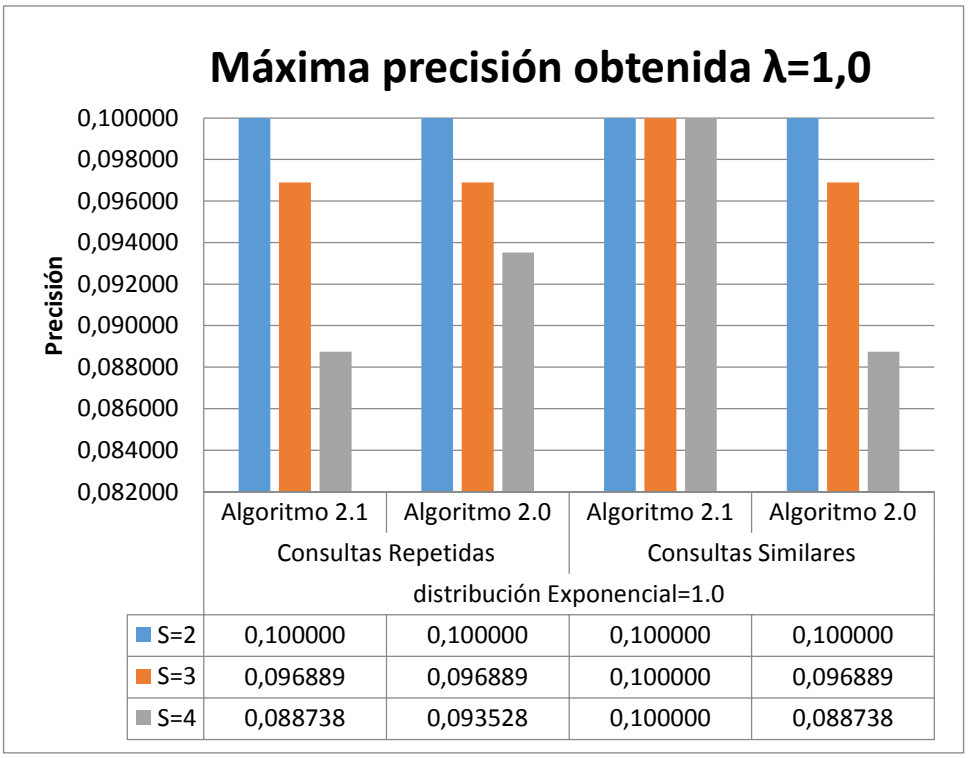
	Colección de Documentos						
Número de Consultas	700 D	1400 D	2100 D	2800 D	3500 D	Máximo	Promedio
30Q	0,093528	0,093528	0,096889	0,093528	0,096889	**0,096889**	**0,094872**
60Q	0,088738	0,093528	0,093528	0,096889	0,088738	**0,096889**	**0,092284**
90Q	0,088738	0,088738	0,082282	0,082282	0,088738	**0,088738**	**0,086156**
120Q	0,099000	0,099000	0,100000	0,100000	0,099000	**0,100000**	**0,099400**
150Q	0,099000	0,100000	0,100000	0,099000	0,096889	**0,100000**	**0,098978**
180Q	0,096889	0,099000	0,099000	0,096889	0,096889	**0,099000**	**0,097733**
210Q	0,096889	0,096889	0,096889	0,099000	0,099000	**0,099000**	**0,097733**
Máximo	**0,099000**	**0,100000**	**0,100000**	**0,100000**	**0,099000**	0,100000	
Promedio	**0,094683**	**0,095812**	**0,095513**	**0,095370**	**0,095163**		**0,095308**

Tabla 24: Precisión máxima y promedio en cada serie de consultas que fueron identificadas como similares utilizando el Algoritmo 2.1 con parámetros S=4, λ =1,5

	Colección de Documentos						
Número de Consultas	700 D	1400 D	2100 D	2800 D	3500 D	Máximo	Promedio
30Q	0,082282	0,082282	0,082282	0,082282	0,082282	0,082282	0,082282
60Q	0,082282	0,082282	0,073825	0,073825	0,082282	**0,082282**	**0,078899**
90Q	0,062869	0,073825	0,073825	0,073825	0,073825	0,073825	0,071634
120Q	0,088738	0,088738	0,093528	0,082282	0,093528	**0,093528**	**0,089363**
150Q	0,082282	0,082282	0,088738	0,082282	0,088738	0,088738	0,084864
180Q	0,088738	0,082282	0,082282	0,082282	0,082282	**0,088738**	**0,083573**
210Q	0,088738	0,082282	0,082282	0,088738	0,088738	0,088738	0,086156
Máximo	0,088738	0,088738	0,093528	0,088738	0,093528	0,093528	
Promedio	**0,082276**	**0,081996**	**0,082395**	**0,080788**	**0,084525**		**0,082396**

Tabla 25: Precisión máxima y promedio en cada serie de consultas similares utilizando el Algoritmo 2.0 con parámetros S=4, λ =1,5.

Es posible notar que para los dos algoritmos utilizados (Algoritmo 2.0 y Algoritmo 2.1), los márgenes de diferencia entre las máximas precisiones obtenidas en la ejecución de todas las series de consultas, esto es para consultas repetidas y consultas similares en un ambiente que contempla todos los conjuntos de documentos, parámetro de distribución Exponencial λ=1,0, λ=1,5 y distribución Zeta S=2, S=3, S=4 (ver Ilustración 10) la precisión promedio para consultas similares siempre es mejor que la precisión promedio de consultas repetidas. Esto se observa si usamos la distribución exponencial λ=1,5 y λ=1,0.



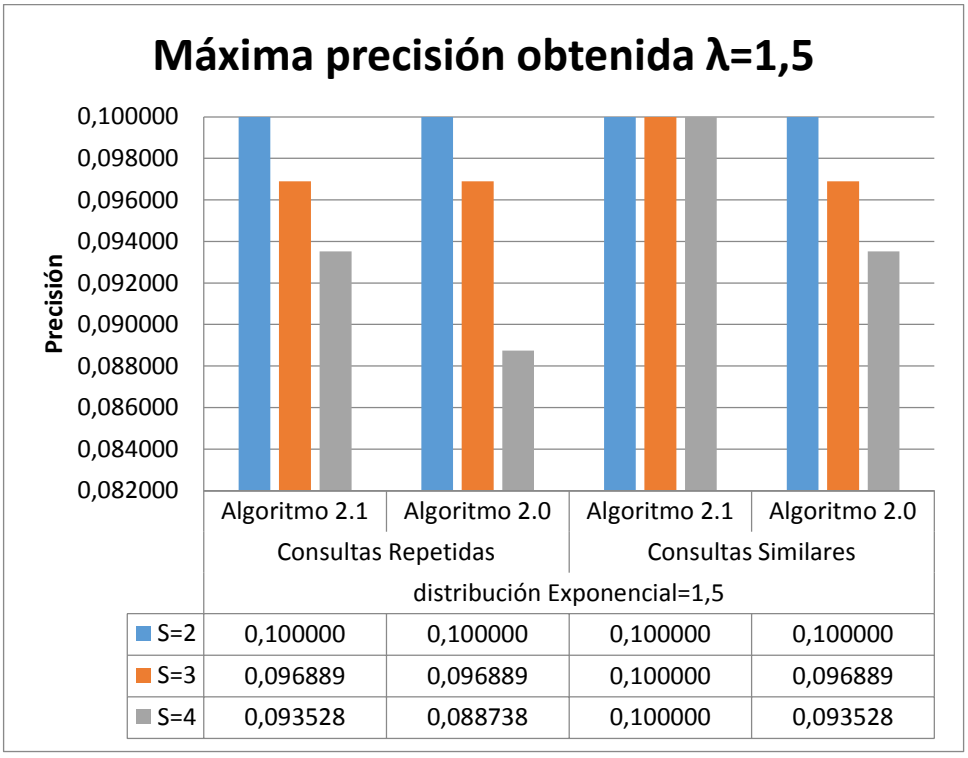
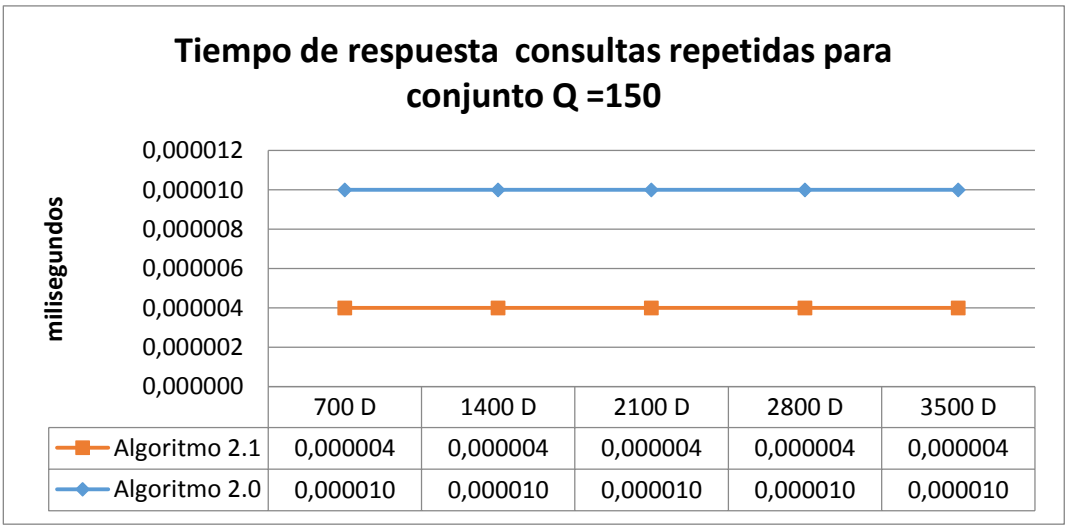


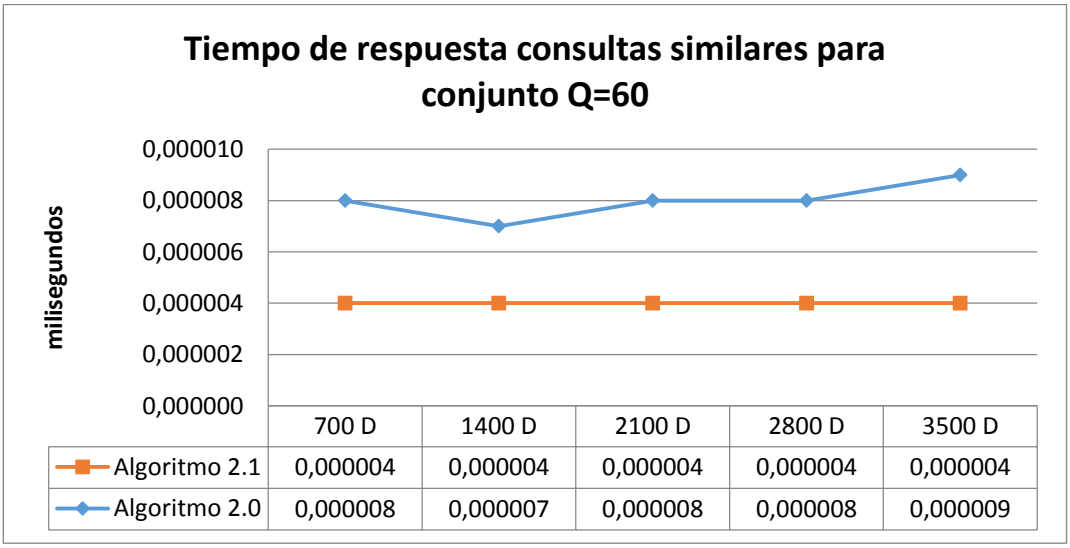
Ilustración 10: Máxima de precisión utilizando valor de λ = 1,0 y λ= 1,5.

La mejora en el tiempo de respuesta en la recuperación de documentos relevantes utilizando el Algoritmo 2.1, esto es para todos parámetros utilizados en los experimentos es siempre mejor que el Algoritmo 2.0. Por ejemplo, en la Ilustración 11, Ilustración 12, Ilustración 13 e Ilustración 14 se puede observar que se disminuyó considerablemente el tiempo de acceso y búsqueda de documentos pertinentes para una nueva consulta. Por lo tanto podemos decir que el uso de resultados de consultas anteriores es útil para responder a una nueva consulta similar o repetida.



	700 D	1400 D	2100 D	2800 D	3500 D
Algoritmo 2.1	0,000004	0,000004	0,000004	0,000004	0,000004
Algoritmo 2.0	0,000010	0,000010	0,000010	0,000010	0,000010

Ilustración 11: Tiempo de respuesta para consultas identificadas como repetidas con parámetros S=4, λ =1,0



	700 D	1400 D	2100 D	2800 D	3500 D
Algoritmo 2.1	0,000004	0,000004	0,000004	0,000004	0,000004
Algoritmo 2.0	0,000008	0,000007	0,000008	0,000008	0,000009

Ilustración 12: Tiempo de respuesta para consultas identificadas como similares con parámetros S=2, λ =1,0

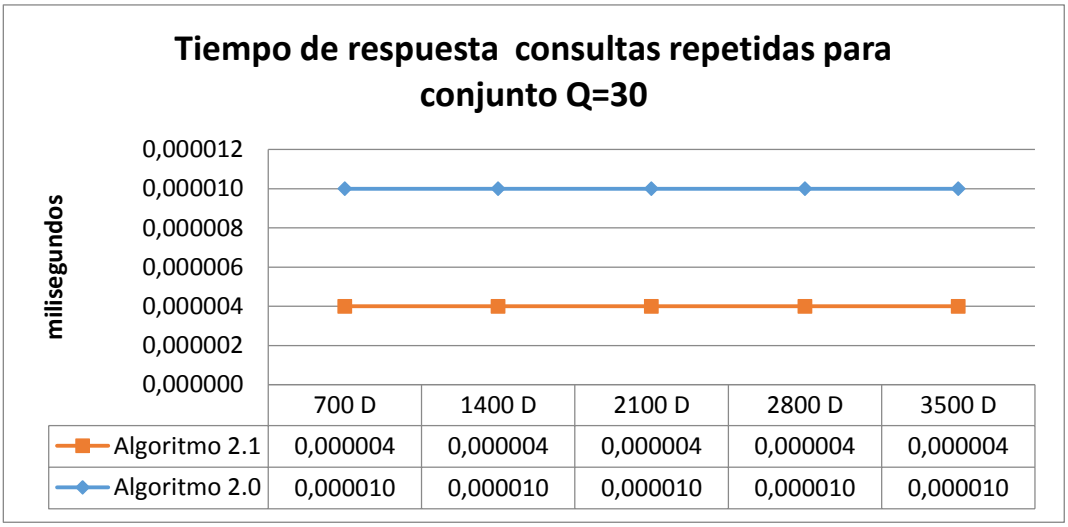


Ilustración 13: Tiempo de respuesta para consultas identificadas como repetidas con parámetros S=3, λ =1,5.

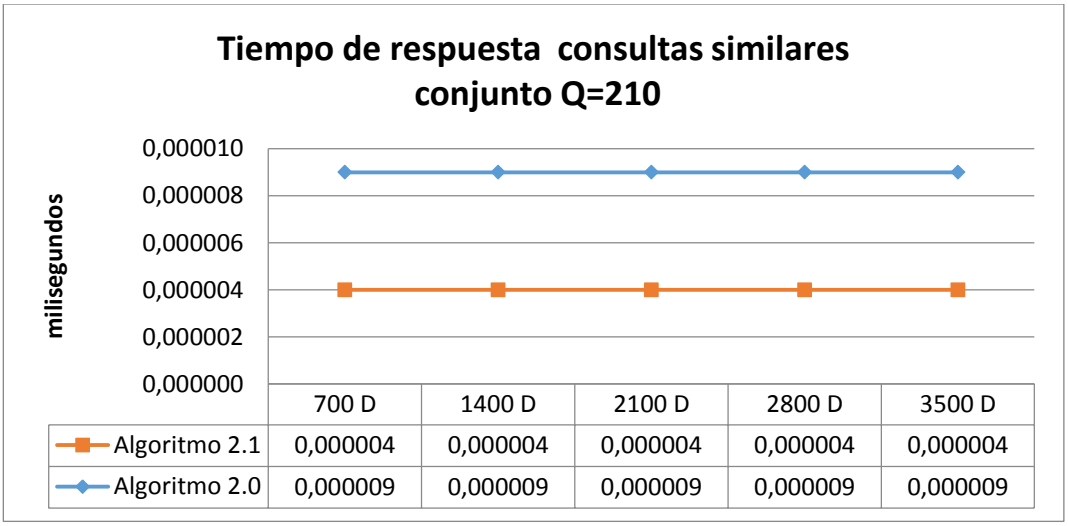


Ilustración 14: Tiempo de respuesta para consultas identificadas como similares con parámetros S=4, λ =1,5.

4.3. Resumen

A partir del escenario proporcionado por (Gutiérrez-Soto, 2013), que da un ambiente ideal para nuestro enfoque, en la Sección 4.1 se ha proporcionado el marco utilizado para dar soporte a la recuperación de información, utilizando el resultado de consultas ejecutadas con anterioridad. Este enfoque se basa en la reutilización de documentos relevantes recuperados de una consulta pasada, para dar respuesta a una nueva consulta procesada por un SRI. En la Sección 4.2 se han simulado dos escenarios con diferentes distribuciones de probabilidad para construir las colecciones de documentos y determinar los documentos pertinentes dados una consulta. Por lo tanto, en base a los resultados, es posible concluir: En primer lugar, los resultados no deben cambiar si se aumenta el

número de documentos. En segundo lugar, el promedio (P@10) para los resultados de búsquedas pasadas (Algoritmo 2.0) y las búsquedas ejecutadas con anterioridad (Algoritmo 2.1), se ven afectados cuando los parámetros de distribución de Zeta son aumentados. Sin embargo, la diferencia entre ellos no debe afectar los resultados finales. En tercer lugar, al realizar una nueva búsqueda y al consultar sobre resultados de búsquedas ejecutadas con anterioridad, el tiempo de esta tarea, siempre es menor al enfoque de búsquedas pasadas, es decir, nuestro enfoque responde de mejor manera, en cuanto a los documentos recuperados y al tiempo de búsqueda para esta evaluación, apoyando así la eficacia de nuestro enfoque.

En términos generales podemos decir que:

- Nuestro Algoritmo presenta mejores resultados que la recuperación tradicional de búsquedas pasadas al evaluar los P@10.
- El tiempo en el cual se responde con los documentos pertinentes, utilizando el Algoritmo 2.1, esto es en el acceso a las consultas ejecutadas pasadas almacenadas en una cola de prioridad, es siempre menor para aquellas consultas que fueron verificadas, mediante la distancia del coseno, como consultas similares o repetidas en el conjunto de consultas del SRI. Ver Ilustración 11, Ilustración 12, Ilustración 13 e Ilustración 14.
- No existe una relación entre el número de consultas y el número de documentos en los resultados finales.

CAPÍTULO V

5. CONCLUSIONES

5.1. Objetivo General

Analizar, diseñar e implementar un algoritmo en línea, el cual recupere documentos de manera eficiente y mejore la precisión utilizando los resultados de consultas pasadas que ya han sido procesados por un sistema de recuperación de la información, considerando su comportamiento con intervalos de tiempo (esto es cómo evolucionan las consultas en el tiempo).

Se logró implementar a través de la unificación de las investigaciones anteriores un algoritmo que permitiera organizar y distribuir consultas ejecutadas con anterioridad de manera eficiente, generando así un algoritmo en línea que recupera los documentos más relevantes para la nueva consulta ejecutada por el usuario. Además de determinar el tiempo de respuesta para la consulta y así determinar qué conjunto de documentos responde de manera más eficaz.

Por otro lado a través de nuestro Algoritmo 2.1 se logró determinar varios escenarios de experimentación (ver Sección 4.1) en los cuales se identificó la nueva consulta, es decir, si es similar o repetida de acuerdo a la consulta más similar del SRI, de esta manera se puede elegir la mejor opción para responder a la nueva consulta con el conjunto de documentos pertinentes de una consulta similar anterior.

5.2. Objetivos Específicos

- **Estudiar los conceptos envueltos en recuperación de la información, así como los métodos tradicionales de almacenamiento.**

Esto se logró mediante el estudio de distintos autores y documentos referidos a la recuperación de información. Ver Capítulo II.

- **Implementación de un Algoritmo en Línea el cual recupere documentos de manera eficiente y mejore la precisión de éstos.**

Como se demostró en la Sección 3.3, se logró mediante la implementación de estructuras de datos que permitió representar las consultas ejecutadas con anterioridad, mejorando el tiempo de respuesta de la consulta, así como también mejorar la precisión de los documentos recuperados mediante la búsqueda en cola de prioridad (ver página 5, Ilustración 10).

- **Analizar y determinar distintos escenarios experimentales acorde al conjunto de consultas similares.**

Esto se logró mediante la utilización de estructuras de datos y distribuciones de probabilidad, esto es para dar al entorno ad hoc para la recuperación de información utilizando consultas pasadas similares (o repetidas), esto es para responder de mejor manera a la nueva consulta de un usuario (ver Sección 4.1).

- **Analizar el rendimiento del algoritmo a través de un análisis empírico, contrastando su rendimiento en un escenario tradicional.**

Como se ve en la Sección 4.2, que se relaciona con el escenario experimental, nuestro enfoque siempre es mejor. Esto es debido a que el almacenamiento de las respuestas de consultas anteriores, resultan ser útiles para responder a nuevas consultas (identificadas como similares o idénticas). Al medir la eficiencia del sistema, la mejora en la recuperación de documentos relevantes, esto es en el tiempo de búsqueda y acceso a el conjunto respuesta (documentos relevantes) suele ser menor en todos los casos al compararlo con el escenario tradicional. Por otra parte la precisión del conjunto de respuesta en la mayoría de los casos comparados fue mayor, en el peor de los casos resulto ser igual la presión, pero aun así el tiempo que tomó el Algoritmo 2.1 en responder fue en todos los casos positivos, esto es comparando el tiempo de respuesta con el Algoritmo 2.0

Bibliografía

Baeza-Yates, R. y Ribeiro-Neto B. 1999. *Modern Information Retrieval.* 1999.

Cleverdon, C.W. 1958. *The evaluation of systems used in information retrieval .In: Proceedings of the International Conference on Scientific Information - Two Volumes.* 1958.

Clough, P., & Sanderson, M. 2013. *Evaluating the performance of information retrieval systems using test collections.* 2013.

Codina, L. 1995. *Teoría de recuperación de información: modelos fundamentales y aplicación a la gestión documental.* 1995.

Garfield. 1980. *Bradford's law.* 1980.

Gutiérrez-Soto, C. and Hubert, G. 2013. *Probabilistic reuse of past search results.* 2013.

Lewis, D. D. and Jones, K. S. 1996. *Natural language processing for information retrieval.* 1996.

Manning, C.D., Raghavan, P., Schütze. 2008. *Introduction to Information Retrieval.* 2008.

Martinez Mendez, F y Rodriguez Muñoz, J. 2004. *Reflexiones sobre la Evaluación de los Sistemas de Recuperación de Información: Necesidad, Utilidad y Viabilidad.* 2004.

Rijsbergen, C. J. V. 1979. *Information Retrieval.* 1979.

Rocchio, J.J. 1973. *Relevance feedback in information retrieval.* 1973.

Salton, G. & McgilL, M.J. 1983. *Introduction to Modern Information Retrieval.* 1983.

Sanderson, M. 2010. *Test Collection Based Evaluation of Information Retrieval Systems.* 2010.

Sanderson, M., Croft, W. 2012. *The history of information retrieval research.* 2012.

Van Rijsbergen, C.J. 1979. *Information retrieval.* 1979.

Villena Román, J. 1997. *Sistemas de Recuperación de Información.* 1997.

Voorhees, E.M. 2005. *TREC: Experiment and Evaluation in Information Retrieval.* 2005.

Yisong Yue, Thorsten Joachims. 2009. *Interactively optimizing information retrieval systems as a dueling bandits problem.* 2009.