Introducción a la recuperación de información

PID_00271446

Blas Torregrosa García

Tiempo mínimo de dedicación recomendado: 2 horas

Universitat Oberta de Catalunya

Blas Torregrosa García

Ingeniero en Informática y máster universitario en Seguridad de las Tecnologías de la Información y de las Comunicaciones (MISTIC) por la Universitat Oberta de Catalunya (UOC). Especializado en ciberseguridad. Profesor colaborador en el máster de Ciencia de Datos de la UOC y profesor asociado en la Universidad de Valladolid (UVA).

El encargo y la creación de este recurso de aprendizaje UOC han sido coordinados por el profesor: Ferran Prados Carrasco (2020)

Primera edición: febrero 2020

Av. Tibidabo, 39-43, 08035 Barcelona
Realización editorial: FUOC

Índice

Introducción

La **recuperación de información** es una disciplina que se ocupa de la representación, el almacenamiento, la organización y el acceso a elementos de información. El objetivo de la recuperación de información es obtener información que pueda ser útil o relevante para el usuario.

Presentaremos la recuperación de información como disciplina científica, proporcionando una caracterización formal basada en la noción de relevancia. Expondremos los criterios de evaluación, la forma en la que se procesan los documentos para obtener un índice y los diferentes modelos clásicos de recuperación.

1. Recuperación de información

La recuperación de información[1] no es un área nueva, sino que se viene desarrollando desde finales de la década de los cincuenta del siglo pasado. Sin embargo en la actualidad, el hecho de disponer de la información necesaria en tiempo y forma puede implicar el éxito o el fracaso de un proyecto.

(1)A menudo abreviada como IR, por sus siglas en inglés, *information retrieval*.

La **recuperación de información** es una «disciplina que se ocupa de la representación, almacenamiento, organización y acceso a elementos de información», según Baeza-Yates. Años antes Salton propuso la definición «un campo relacionado con la estructura, análisis, organización, almacenamiento, búsqueda y recuperación de información».

Las siguientes definiciones inciden explícitamente en el papel del usuario como fuente de consultas y destinatario de las respuestas. Así, Croft considera la recuperación de información como «el conjunto de tareas mediante las cuales el usuario localiza y accede a los recursos de información que son pertinentes para la resolución del problema planteado. En estas tareas desempeñan un papel fundamental los lenguajes documentales, las técnicas de resumen, la descripción del objeto documental, etc.». Por otro lado Korfhage indica que «la localización y presentación a un usuario de información relevante a una necesidad de información se expresa mediante una consulta».

La recuperación de información intenta resolver el problema de «encontrar y organizar documentos relevantes que satisfagan la necesidad de información de un usuario, expresada en un determinado lenguaje de consulta».

1.1. Recuperación de información y recuperación de datos

Aunque puedan parecer conceptos muy parecidos, existen diferencias significativas en cuanto a los objetos con que se trata y su representación, pero también en cuanto a la expresión de las consultas y los resultados.

En **recuperación de datos** los objetos considerados son estructuras de datos conocidas. Su representación se basa en un formato previo bien definido y con un significado implícito para cada elemento. Por ejemplo, una tabla en una base de datos que almacena instancias de clientes de una organización posee un conjunto de columnas que definen los atributos de todos los clientes, y cada fila corresponde a un cliente en particular. Cada elemento (atributo) tiene un dominio conocido y su semántica está claramente establecida.

Por otro lado, en **recuperación de información** el objeto de tratamiento es básicamente un documento de texto, en general sin estructura.

En cuanto a las consultas, la recuperación de datos cuenta con una estructura bien definida dada por un lenguaje de consulta que permite su especificación de manera exacta. Las consultas no son ambiguas y constan de un conjunto de condiciones que deben cumplir los elementos.

Bases de datos relacionales

En bases de datos relacionales, las consultas se especifican utilizando el lenguaje SQL (*Structured Query Language*), cuya semántica es precisa.

Finalmente, en un sistema de recuperación de datos los resultados consisten en el conjunto completo de elementos que satisfacen todas las condiciones de la consulta. Como la consulta no admite errores, el resultado tiene que ser el exacto. Y el orden de aparición de los resultados es el especificado por la consulta (o ninguno), pero no tiene ninguna influencia en la importancia del resultado.

Tabla 1. Diferencias entre recuperación de datos y recuperación de información

	Recuperación de datos	**Recuperación de información**
Estructura	Información estructurada con semántica bien definida.	Información no estructurada.
Recuperación	**Determinística**. Todo el conjunto «Solución» es relevante para el usuario.	**Probabilística**. Una parte de los documentos recuperados puede no ser relevante.
Consulta	**Especificación precisa**. Lenguaje formal, preciso y estructurado	**Especificación imprecisa**. Lenguaje natural, ambiguo y no estructurado.
Resultados	Aciertos exactos.	Aciertos parciales.

1.2. El problema de la recuperación de información

De forma general, el problema de la recuperación de información tiene que abordarse desde dos puntos de vista: el computacional y el humano. El **computacional** tiene que ver con la construcción de estructuras de datos y algoritmos eficientes que mejoren la calidad de las respuestas. En el caso **humano** se refiere al estudio del comportamiento y de las necesidades del usuario.

Desde un punto de vista general, el problema de la recuperación de información consta de los elementos siguientes:

- Un **conjunto de documentos** que contienen información de interés (sobre uno o varios temas).

- Unos **usuarios** con unas necesidades de información que plantean al sistema de recuperación de información en forma de una **consulta**.

- Como respuesta, el sistema devuelve referencias a **documentos relevantes**, es decir, aquellos que satisfacen la necesidad de información solicitada, generalmente en forma de una lista ordenada.

Para cumplir con sus objetivos estos sistemas deben realizar algunas tareas básicas, las cuales se encuentran formuladas como procedimientos computacionales:

- Representación lógica de los documentos. Algunos sistemas solo almacenan partes de los documentos, mientras otros lo hacen de manera completa.

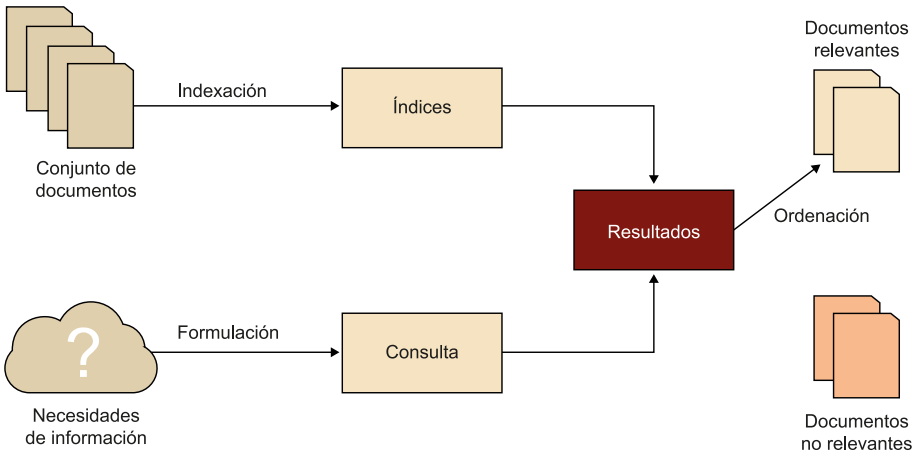
- Representación de la necesidad de información del usuario en forma de consulta.

- Evaluación de los documentos respecto de una consulta para establecer la relevancia de cada uno.

- Ordenación (ranquin) de los documentos considerados relevantes para formar el conjunto «Solución» o «Respuesta».

- Presentación de la respuesta al usuario.

Figura 1. El problema de la recuperación de información



Como se puede observar, se parte de un conjunto de documentos, los cuales están compuestos por sucesiones de palabras con su estructura gramatical (párrafos, secciones, etc.). Estos documentos suelen estar escritos en lenguaje natural. El conjunto de todos los documentos tratados por el sistema se denomina **corpus** o **colección**. Para poder realizar operaciones sobre un corpus, es necesario obtener en primer lugar una representación lógica de todos sus documentos, la cual puede consistir en un conjunto de términos, frases u otras unidades (sintácticas o semánticas) que permitan caracterizarlos.

A partir de esa representación lógica de los documentos, un proceso de **indexación** llevará a cabo la construcción de estructuras de datos, denominadas **índices**, que almacenen dicha representación. Estas estructuras permiten realizar búsquedas eficientes.

El **algoritmo de búsqueda** acepta como entrada una expresión de consulta del usuario y comprobará en el índice qué documentos pueden satisfacerla. A continuación un **algoritmo de ordenación** o ranquin determinará la relevancia de cada documento y devolverá una lista con la respuesta. Se sobrentiende que el primer elemento de dicha lista corresponde al documento más relevante respecto a la consulta.

En general, un sistema de recuperación de información no da una respuesta directa a una consulta, sino que permite localizar referencias a documentos que pueden contener información útil.

1.3. Definiendo relevancia

La recuperación de información se define como la disciplina que en una consulta encuentra documentos relevantes en lugar de simples coincidencias de patrones. Esto revela un aspecto fundamental: la **relevancia** de los resultados se evalúa en relación con la necesidad de información, no con la consulta.

Vamos a ver un ejemplo. Supongamos la necesidad de información para determinar si comer chocolate es beneficioso para reducir la presión arterial. Se podría expresar esto en una consulta a un motor de búsqueda: «chocolate efecto presión». Se evaluará un documento resultante como relevante si responde a la necesidad de información, y no solo porque contiene todas las palabras de la consulta.

Cabe destacar que la relevancia es un concepto con propiedades interesantes:

1) La primera es que es **subjetivo**: dos usuarios pueden tener la misma necesidad de información y emitir juicios diferentes sobre el mismo documento recuperado.

2) Otra característica es su **naturaleza dinámica**, tanto en el espacio como en el tiempo: los documentos recuperados y mostrados al usuario en un momento dado pueden influir en la relevancia de los documentos que se mostrarán más adelante. Además, de acuerdo con su estado actual, un usuario puede expresar diferentes juicios sobre el mismo documento (dada la misma consulta).

3) Y finalmente, la relevancia es **multifacética**, ya que está determinada no únicamente por el contenido de un resultado recuperado, sino también por factores como la autoridad, la credibilidad, la especificidad, la exhaustividad, la actualidad y la claridad de la fuente.

Cabe reseñar que la relevancia no es conocida por el sistema antes del juicio del usuario. De hecho, podríamos decir que la tarea de un sistema de recuperación de información es «adivinar» un conjunto D de documentos relevantes con respecto a una consulta dada q calculando una función de relevancia R para cada documento de la colección.

1.4. Tratando con datos no estructurados

Una de las dificultades fundamentales en el proceso de recuperación de información es la naturaleza no estructurada (generalmente texto) de los documentos, y otra es el tamaño ingente de las colecciones de documentos.

Un punto clave en recuperación de información con respecto a recuperación de datos es su naturaleza no estructurada. La recuperación de datos, tal como la realizan las bases de datos relacionales, se refiere a recuperar todos los objetos que satisfacen condiciones claramente definidas y expresadas a través de un lenguaje de consulta formal. En ese contexto, los datos tienen una estructura bien definida y se accede a través de lenguajes de consulta como SQL. Además, los resultados tienen coincidencias exactas, es decir, no se devuelven coincidencias parcialmente correctas como parte de la respuesta. Por lo tanto la relevancia no se aplica a la recuperación de datos.

Por otra parte, la «sociedad digital» está produciendo una gran cantidad de contenidos. De hecho, mientras que en 2006 se crearon o replicaron en todo el mundo alrededor de 10^{18} bytes (10K petabytes) de información, en 2010 este número aumentó en un factor de 6 (988 exabytes, es decir, casi un zettabyte). Comenzó entonces la **era del zettabyte**, que es un período de la historia de la humanidad y de la informática en el que el tráfico en internet global superó por primera vez un zettabyte (hecho que ocurrió en 2016) y la cantidad de datos digitales en el mundo superó por primera vez el zettabyte (lo que sucedió en 2012).

Petabytes y zettabyte

Un **petabyte** es una unidad de almacenamiento de información que equivale a 10^{15} bytes. Un **zettabyte** es una unidad de almacenamiento de información que equivale a 10^{21} bytes (= 1.099.511.627.776 gigas).

1.5. Definición formal

Un sistema de recuperación de información (SRI) se puede caracterizar por una cuaterna:

$$\text{SRI} = \{D, Q, F, R(q_k, d_j)\}$$

donde:

- D es el conjunto de representaciones de los documentos en la colección (cada uno referenciado como d_j).

- Q es el conjunto de las representaciones de las necesidades de información del usuario, denominadas **consultas** (referenciadas individualmente como q_k).

- F es un marco de trabajo o estrategia para modelar la representación de los documentos, las consultas y sus relaciones.

- $R(q_k, d_j)$ es una función de ordenación o ranquin que asocia un número real a cada documento d_j según su relevancia respecto a la consulta q_k.

La función $R(q_k, d_j)$ determina el orden de relevancia de los documentos y es la pieza clave en todo el proceso de recuperación de información.

1.6. Aplicaciones de la recuperación de información

La aplicación de los sistemas de recuperación de información más conocida y extendida son los motores de búsqueda, pero las técnicas de recuperación de información también son fundamentales para otras tareas.

Un **motor de búsqueda** es un sistema de recuperación de información diseñado para ayudar a encontrar información almacenada en sistemas informáticos. Los resultados de búsqueda generalmente se presentan como una lista y se denominan **resultados**. Los motores de búsqueda ayudan a minimizar el tiempo necesario para encontrar información.

Motor de búsqueda

Un ejemplo son los buscadores de Internet (generalmente en la web).

Los **sistemas de filtrado** eliminan información redundante o no deseada que aparece en un flujo de información, mediante métodos automáticos, antes de presentarlos a usuarios. Una aplicación clásica del filtrado de información son los filtros de *spam*, que aprenden a distinguir entre correos electrónicos útiles y correos dañinos en función del contenido.

Los **sistemas de recomendación** (pueden considerarse como una forma de filtrado de información) presentan al usuario elementos de información interesantes, como canciones, películas, libros, etc. en función de su perfil o de las elecciones de elementos parecidos por proximidad geográfica, conocimiento o intereses comunes.

El **resumen de documentos** es otra aplicación consistente en crear una versión abreviada de un texto para reducir la sobrecarga de información. El resumen suele ser generalmente extractivo, es decir, se seleccionan las frases más relevantes de un documento y se recopilan para formar una versión más compacta del documento.

La agrupación y categorización de documentos también son aplicaciones importantes en recuperación de información. La **agrupación** consiste en reunir documentos en función de su afinidad, mientras que la **categorización** utiliza una serie predefinida de clases y asigna a cada documento a la clase más relevante. Las aplicaciones típicas de la categorización son la identificación de categorías en artículos y en noticias.

Los **sistemas de respuesta a preguntas**[2] se ocupan de la selección de documentos relevantes para responder a las consultas del usuario, formuladas en lenguaje natural. La característica principal de estos sistemas es que proporcionan respuestas en forma de oraciones, frases o incluso palabras relevantes, dependiendo del tipo de pregunta formulada.

[(2)]QA, por sus siglas en inglés, *Question Answering*.

Finalmente, un asunto interesante se refiere a la recuperación en varios idiomas, es decir, la recuperación de documentos en un idioma diferente del idioma en el que se formuló la consulta del usuario. Una aplicación notable de esta tecnología se refiere a la recuperación de documentos legales.

2. Evaluación de un sistema de recuperación de información

Hasta este momento se ha considerado la relevancia como el criterio clave para determinar la calidad de un sistema de recuperación de información, destacando el hecho de que se refiere a una necesidad implícita del usuario.

Para formalizar esta cuestión, se dice que un sistema de recuperación de información será **medible** en términos de relevancia cuando se dispone de la información siguiente:

1) Una colección D de documentos de referencia,

2) un conjunto Q de consultas de referencia, y

3) una terna $t_{jk} = \langle d_j, q_k, r* \rangle$ para cada consulta $q_k \in Q$ y cada documento $d_j \in D$ que contiene un juicio binario de relevancia $r*$ del documento d_j con respecto a la consulta q_k, juicio emitido por una autoridad de referencia.

En términos generales, la **precisión** (P) es la fracción de documentos recuperados que son relevantes para una consulta y proporciona una medida de la «solidez» del sistema.

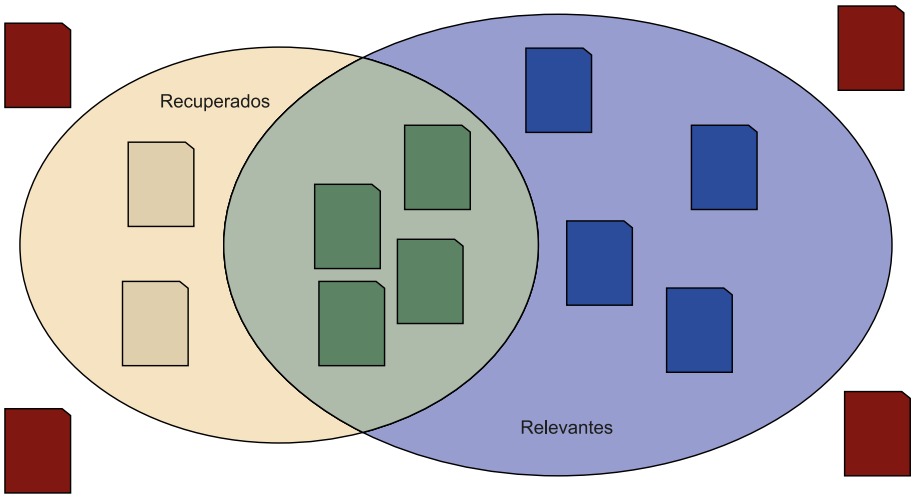
La precisión no tiene que ver con el número total de documentos que el sistema considera relevantes. Este detalle se explica mediante la **exhaustividad**[3] (R), que se define como la fracción de documentos «verdaderamente» relevantes que se recuperan efectivamente y, por lo tanto, proporciona una medida de la «integridad» del sistema.

(3)Se ha traducido el término en inglés *recall* como exhaustividad por ser el más aproximado conceptualmente.

Tabla 2. Matriz de confusión para evaluación

	Relevante	**No relevante**
Recuperado	**Verdaderos positivos (TP)**	**Falsos positivos (FP)** Error de tipo I
No recuperado	**Falsos negativos (FN)** Error de tipo II	**Verdaderos negativos (TN)**

Figura 2. Medidas de rendimiento



Más formalmente, dado el conjunto completo de documentos D y una consulta q, se define el subconjunto $TP \subseteq D$ como el conjunto de resultados **verdaderos positivos**, es decir, documentos recuperados que son realmente relevantes para la consulta q. Y se define $FP \subseteq D$ como el conjunto de **falsos positivos**, es decir, el conjunto de documentos recuperados que no son relevantes para la consulta q. También el subconjunto $FN \subseteq D$ como el conjunto de documentos que corresponden a las necesidades del usuario pero que el sistema no recupera. Con esta notación se definen la precisión y la exhaustividad:[4]

(4)Denominada *R*, del inglés, *recall*.

$$\text{Precisión}(P) = \frac{TP}{TP+FP}$$

y

$$\text{Exhaustividad}(R) = \frac{TP}{TP+FN}$$

Estas dos medidas se encuentran altamente correlacionadas. Empíricamente se ha comprobado que una alta exhaustividad viene acompañada de una precisión muy baja y viceversa; es decir, se cumple una relación inversa entre las dos medidas.

Existe un compromiso entre exhaustividad y precisión, de modo que aumentar la exhaustividad (recuperando mayor cantidad de documentos) hace que disminuya la precisión (aumentando el número de documentos no relevantes). Por el contrario, si recuperamos unos pocos documentos y todos son relevantes, se tendrá una precisión máxima, pero seguramente habrá documentos relevantes que no son recuperados. El sistema ideal es aquel que siempre recupera todos los documentos relevantes y solo esos.

Como la precisión y la exhaustividad tienen diferentes ventajas y desventajas, se ha definido una única medida de evaluación que equilibra las dos mediciones. Se denomina **Medida-F1** (*Score-F1*) o **media harmónica** de P y R:

$$F_1 = 2\frac{PR}{P+R} = \frac{2 \times TP}{2 \times TP + FN + FP}$$

Esta medida combina la precisión y la exhaustividad en un único valor comprendido entre 0 y 1. Lo interesante de esta métrica es que un valor máximo de F_1 corresponde al mejor compromiso entre P y R, y su valor solamente será alto cuando ambas medidas tengan valores altos. Si $F_1 = 0$, no se han recuperado documentos relevantes, mientras que si $F_1 = 1$ se han recuperado todos los documentos relevantes y solo estos.

La **exactitud**[5] (A) es la medida de rendimiento más intuitiva y es simplemente una relación entre lo recuperado correctamente (TP y TN) y el total de documentos:

(5)Abreviada como A, del inglés *accuracy*.

$$A = \frac{TP + TN}{TP + TN + FP + FN}$$

El **ruido**[6] (N) determina la proporción de documentos no relevantes hallados en los documentos recuperados:

(6)Abreviado como N, del inglés *noise*.

$$N = \frac{FP}{TP + FP}$$

3. Preprocesamiento

Es evidente que recorrer todos los documentos de una colección cada vez que se realiza una consulta es una solución poco práctica, y en general imposible. Para evitar esto hay que indexar los documentos con antelación.

Un **índice** es una vista lógica que representa los documentos de una colección mediante un conjunto de términos o palabras clave, esto es, cualquier palabra que aparezca en el texto del documento. Un **término** es una instancia de una secuencia de caracteres agrupados para su procesamiento y que tienen una unidad semántica.

La idea que subyace a la indexación es que tanto la semántica de los documentos como las necesidades de información del usuario pueden expresarse adecuadamente mediante conjuntos de términos y las relaciones entre ellos. Esto se debe a que no todos los términos que componen un documento son igualmente representativos de su contenido. Cuestiones como su posición, la cantidad de apariciones o su función lingüística definen el grado de importancia de cada uno de los términos.

El resultado es una representación de la colección computacionalmente adecuada para los procesos siguientes y se denomina **indexación de la colección**.

3.1. Proceso de indexación

La **indexación** es una operación que tiene como propósito la identificación de los términos que representan el contenido de un documento y la traducción de estos a una forma computacionalmente manejable.

El concepto de indexación incluye la construcción de las estructuras de datos que permitan almacenar los términos representativos para posibilitar posteriormente la recuperación eficiente de los documentos.

3.1.1. Operaciones con el texto

Cuando consideramos un texto en lenguaje natural es fácil notar que no todas las palabras sean igualmente representativas de la semántica del documento. Por lo general, los sustantivos (o grupos de palabras que contienen sustantivos) son los componentes más representativos de un documento en términos de contenido.

Según esto, el sistema de recuperación de información también procesa previamente el texto de los documentos para determinar los términos más «importantes» que se utilizarán para construir el índice. Por lo tanto se selecciona un subconjunto de todas las palabras para representar el contenido de un documento.

Para seleccionar las palabras clave la indexación debe cumplir dos objetivos diferentes y potencialmente opuestos:

1) Ser **exhaustivo**, es decir, usar un número suficientemente grande de términos del documento.

2) La **especificidad**, es decir, excluir términos genéricos que tienen poca semántica y que agrandan el tamaño del índice.

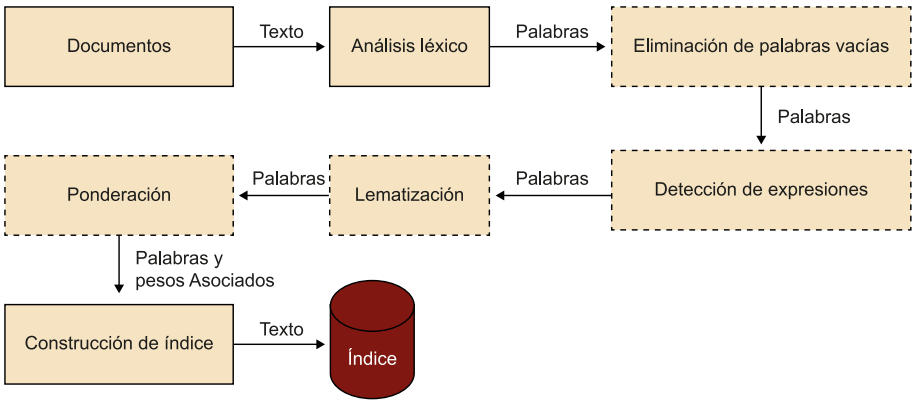
Ciertos términos muy genéricos, como artículos, conjunciones o preposiciones, tienen un bajo poder discriminante pues su frecuencia en cualquier documento tiende a ser alta. En otras palabras, los términos genéricos tienen una **frecuencia de término** alta. Por el contrario, los términos específicos tienen un poder discriminante más alto, debido a su escasa aparición: tienen una **frecuencia de documento** baja.

Frecuencia de término y de documento

Frecuencia de término (*term frequency*) definida como el número de veces que aparece el término en el documento.

Frecuencia de documento (*document frequency*) definida como el número de documentos en los que aparece el término.

Figura 3. Proceso de indexación



A continuación se describen las fases de preprocesamiento realizadas por un sistema de recuperación de información, tomando como entrada un documento y produciendo como salida los términos para el índice.

1) Análisis del documento. Los documentos vienen en todo tipo de idiomas, juegos de caracteres y formatos. Incluso el mismo documento puede contener múltiples idiomas o formatos. El análisis se ocupa del reconocimiento y la descomposición de la estructura del documento en sus componentes.

2) Análisis léxico. El análisis léxico convierte el documento en un conjunto de palabras o *tokens*. Hay una serie de dificultades relacionadas con el análisis léxico que incluyen la identificación correcta de los separadores de palabras (si el idioma los tiene), abreviaturas o fechas. Esta complejidad depende mucho

del idioma del documento. El reconocimiento de abreviaturas y, en particular, de expresiones de tiempo presenta bastante complejidad y hay bastantes estudios sobre este particular.

3) Eliminación de palabras vacías (*stop-words*). Un paso posterior es la eliminación de palabras vacías, es decir, la eliminación de palabras de alta frecuencia con poca carga semántica, aunque con sentido gramatical. Existen listas de palabras vacías para cada idioma, que incluyen, normalmente, artículos, preposiciones, conjunciones, etc. Sin embargo, como este proceso puede disminuir la exhaustividad, en algunos motores de búsqueda no la implementan.

4) Detección de expresiones. Este paso intenta capturar el significado del texto más allá de una lista de palabras mediante la identificación de grupos de sustantivos y otras expresiones. La detección de frases se puede abordar de varias maneras, como el uso de reglas (por ejemplo, la retención de términos que no están separados por signos de puntuación), el análisis morfológico o el análisis sintáctico. Un enfoque común para la detección de expresiones se basa en el uso de **tesauros**. Alternativamente existen técnicas de aprendizaje automático, como el algoritmo de extracción de claves (KEA),[7] que identifica expresiones candidatas utilizando métodos léxicos.

[7]Por sus siglas en inglés *Key Extraction Algorithm*.

Tesauro

Un **tesauro** es una lista de palabras o términos controlados, empleados para representar conceptos. Los tesauros hechos manualmente son generalmente jerárquicos y contienen términos relacionados, ejemplos de uso y casos particulares.

5) Lematización. El siguiente paso intenta normalizar las palabras mediante la eliminación de sufijos. El objetivo es obtener la forma básica (diccionario) de cada palabra eliminando la parte flexiva de la misma que se usa para formar plurales, género, conjugaciones verbales, formas adverbiales, etc. El método clásico de abordar esto fue ideado por Porter mediante un algoritmo basado en reglas.

6) Ponderación. La fase final del procesamiento es la ponderación de los términos. Como se mencionó anteriormente, cada palabra en un texto tiene un poder descriptivo diferente y, por lo tanto, los términos se pueden ponderar de manera diferente para tener en cuenta su importancia dentro de un documento o en toda la colección de documentos.

Uno de los métodos de ponderación más usado es el denominado **TF*IDF** (*Term Frequency*, *Inverse Document Frequency*), el cual establece una relación entre la frecuencia de un término dentro de un documento y su frecuencia en todos los documentos de la colección (*N*), es decir, se obtiene la frecuencia del término t_i en el documento d_j(*TF*) y se multiplica por el recíproco de la cantidad de documentos (*n*) de la colección en los que aparece t_i(*IDF*):

$$TF * IDF_{ij} = TF_{ij} \times \log_2(N/n)$$

En el cálculo del *IDF* los valores próximos a 0 indican que el término posee poco peso y, por tanto, bajo valor de discriminación. Por el contrario, los valores alejados de 0 indican que el término es poco frecuente y resulta más adecuado para caracterizar los documentos en los cuales se encuentra.

En general la indexación se basa en el análisis de la frecuencia de los términos y su distribución en los documentos. Este análisis tiene como objeto establecer criterios que permitan determinar si una palabra es un término de indexación válido, fundamentalmente porque permite discriminar el contenido de los documentos y, de alguna forma, aporta información.

3.1.2. Leyes empíricas sobre el texto

Existen algunas propiedades interesantes en el lenguaje y su uso que pueden ser útiles para comprender el proceso de indexación pues determinan cómo se distribuyen las frecuencias de aparición de las diferentes palabras en una colección y cómo crece el tamaño del vocabulario conforme crece tal colección.

Ley de Zipf

Formulada en la década de los cuarenta del siglo pasado por George K. Zipf, lingüista de la Universidad de Harvard, quien realizó una serie de estudios empíricos que demostraron que la gente al escribir tiende a preferir palabras más conocidas a las menos conocidas. Denominó a esto *ley del mínimo esfuerzo*.

La **ley de Zipf** establece que, dada una lista de las palabras junto con la frecuencia de aparición de cada una ordenadas de mayor a menor, se cumple que la frecuencia $f(w)$ de una palabra multiplicada por su posición $r(w)$ en lista ordenada, es igual a una constante C que depende del idioma, es decir:

$$C = r(w) \times f(w)$$

Palabras más frecuentes

En español, las diez palabras más frecuentes según la RAE son *de*, *la*, *que*, *el*, *en*, *y*, *a*, *los*, *se* y *del*. En concreto *la* aparece la mitad de las veces que *de*, *que* un tercio de las veces que *de*, y así sucesivamente.

La ley de Zipf es una ley de potencias, lo que quiere decir que da igual el tamaño del texto que estemos estudiando, y que esta proporción en la frecuencia de aparición de las palabras siempre se cumple.

Además la ley de Zipf se aplica a todos los idiomas, independientemente de la familia a la que pertenezcan. Por lo tanto tiene que ver con la forma en la que el cerebro procesa el lenguaje.

Ley de Heaps

Esta ley plantea la relación entre el tamaño del texto (número de palabras) y el crecimiento del vocabulario (número de palabras únicas). En particular, determina que el tamaño del vocabulario V (y su crecimiento) es una función del tamaño del texto N (medido en palabras):

$$V = K\,N^{\beta}$$

donde K es una constante (entre 10 y 100) y β es otra constante entre 0 y 1 (normalmente entre 0,4 y 0,6).

Este hallazgo es muy importante para la escalabilidad del proceso de indexación puesto que establece que el tamaño del vocabulario (y el tamaño del índice) presenta un crecimiento sub-lineal con respecto al crecimiento del número de documentos.

3.2. Estructuras de datos para indexación

En este apartado se presentan las estructuras de datos básicas para la implementación de sistemas de recuperación de información. A partir de los conceptos y de las técnicas expuestas en el apartado anterior sobre preprocesamiento, resulta necesario contar con estructuras de datos eficientes que soporten las estrategias de búsquedas. La justificación de la indexación es que el coste (en términos de tiempo y espacio de almacenamiento) dedicado a la creación del índice se recupera en la ejecución de múltiples consultas.

Por lo tanto la primera pregunta que debe abordarse al afrontar la indexación es qué estructura de almacenamiento debería usar para maximizar la eficiencia de recuperación. Una primera solución simplista utilizaría una matriz de documentos y términos, es decir, una matriz donde las filas corresponden a términos y las columnas corresponden a documentos de la colección. De este modo, cada celda w_{ij} representa el peso del término t_i en el documento d_j.

Tabla 3. Matriz de asociación término-documento

	d_1	d_2	d_3	...	d_n
t_1	w_{11}	w_{12}	w_{13}	...	w_{1n}
t_2	w_{21}	w_{22}	w_{23}	...	w_{2n}
t_3	w_{31}	w_{32}	w_{33}	...	w_{3n}
...	...	...	...	...	...
t_m	w_{m1}	w_{m2}	w_{m3}	...	w_{mn}

Sin embargo, en el caso de grandes colecciones de documentos, este criterio daría como resultado una matriz con muy pocos valores (y muchos huecos), ya que la probabilidad de que cada palabra aparezca en un documento de la colección disminuye con el número de documentos.

A continuación se presentan algunas mejoras a este planteamiento.

3.2.1. Índices invertidos

El fundamento de un índice invertido es muy sencillo. Primero hay que crear un diccionario de términos V (denominado también *vocabulario*) que contiene todos los términos únicos de la colección de documentos.

A continuación, para cada término $t_i \in V$ se crea una lista L_i que contiene la referencia a cada documento d_j en el que t_i aparece. Esta lista L_i se denomina **lista de publicación** o **lista invertida** y puede contener información adicional, como la frecuencia (TF*IDF) o la posición de t_i dentro de d_j.

El conjunto del diccionario de términos y todas sus listas invertidas se denomina **índice invertido**.

Tabla 4. Índice invertido con información de frecuencia

Vocabulario	Listas invertidas
t_1	(d_1, 1), (d_3, 4), (d_5, 2), ..., (d_n, 1)
t_2	(d_3, 2)
t_3	(d_1, 1), (d_2, 6)
...	...
t_m	(d_2, 3)

Los índices invertidos no tienen rival en cuanto a eficiencia: de hecho, como un término generalmente aparece en muchos documentos, se reducen las necesidades de almacenamiento. Además, todas estas estructuras admiten compresión de forma que puedan caber en memoria.

Dada esta estructura de índice invertido, el proceso de búsqueda consta de cuatro pasos principales:

1) Acceder al diccionario de términos para identificar los términos de la consulta.

2) Para cada término de la consulta se recuperan las listas invertidas.

3) Filtrar los resultados: si la consulta se compone de varios términos (posiblemente conectados por operadores lógicos), es necesario fusionar las listas de resultados parciales.

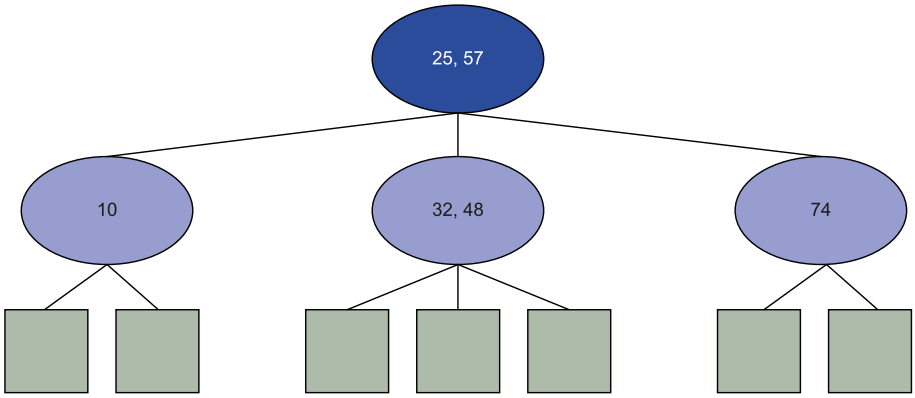
4) Finalmente, entregar la lista de resultados.

3.2.2. Árboles B y B+

Los **árboles B** constituyen una categoría muy importante de estructuras de datos que permiten una implementación eficiente de conjuntos y diccionarios para operaciones de consulta. Existe una gran variedad de árboles B: los árboles **B**, **B+** y **B***; aunque todas ellas están basadas en la misma idea, se incluye la utilización de árboles de búsqueda no binarios y con condición de balanceo.

Los **árboles B+** en concreto son ampliamente utilizados en la representación de índices en bases de datos. De hecho, este tipo de árboles está diseñado específicamente para estas aplicaciones, en las que la característica fundamental es el tiempo en las operaciones de acceso a datos.

Figura 4. Árbol B balanceado



Los **árboles B** se usan mucho en bases de datos debido a su tiempo de acceso reducido: de hecho, el número máximo de accesos para un árbol B está limitado a la profundidad del árbol. Un inconveniente de los árboles B es su bajo rendimiento en las búsquedas secuenciales. Este problema puede ser mitigado por la variante del árbol B+, en la que los nodos hoja o finales están vinculados formando una cadena que sigue un orden.

Otro problema que suelen plantear los árboles B es que pueden perder el balanceo después de muchas inserciones. Esto puede modificarse adoptando procedimientos de regeneración del balanceo del árbol.

4. Modelos de recuperación de información

En este apartado se presentan tres modelos clásicos de recuperación de información: el **booleano**, el de **espacio vectorial** y el **probabilístico**. Estos modelos proporcionan los fundamentos de la evaluación de consultas, que es el proceso que recupera los documentos relevantes según la consulta de un usuario.

4.1. Modelo booleano

El **modelo de recuperación booleana** es un modelo de recuperación basado en la teoría de conjuntos y en el álgebra de Boole, mediante el cual las consultas se definen como expresiones booleanas con términos de índice (y utilizando los operadores booleanos `AND`, `OR` y `NOT`); por ejemplo, «foto AND montaña OR nieve».

En el modelo booleano la representación de la colección de documentos se realiza sobre una matriz binaria documento-término, donde los términos han sido extraídos manual o automáticamente de los documentos y representan su contenido.

Tabla 5. Matriz binaria término-documento

	d_1	d_2	d_3	...	d_n
t_1	0	0	1	...	0
t_2	1	0	1	...	1
t_3	0	1	0	...	0
...	...	...	...	...	...
t_m	0	0	0	...	1

Una consulta booleana q se puede resolver recuperando todos los documentos que contienen los términos de la consulta y creando una lista para cada término. Una vez que tales listas estén disponibles, los operadores booleanos deben manejarse de la siguiente manera:

- q_1 `OR` q_2: requiere construir la **unión** de las listas de q_1 y q_2.
- q_1 `AND` q_2: requiere construir la **intersección** de las listas de q_1 y q_2.
- q_1 `AND NOT` q_2: requiere construir la **diferencia** de las listas de q_1 y q_2.

Se trata de un modelo muy sencillo, pero mediante algunas extensiones se permite usar el carácter comodín * para indicar la aceptación de coincidencias de términos parciales (`mont* OR nie*`). Otras extensiones incluyen el operador de proximidad `NEAR`, es decir, una forma de expresar que dos términos en una consulta deben aparecer cerca uno del otro en un documento (*rock NEAR roll*).

4.2. Modelo de espacio vectorial

El **modelo de espacio vectorial** representa documentos y consultas como vectores de un espacio vectorial en el que cada dimensión corresponde a un término del vocabulario.

El fundamento de este modelo es que cada término t_i del diccionario V se representa como un vector de un espacio vectorial euclídeo de dimensión $|V|$, en el que cada t_i tiene todas sus componentes iguales a 0 excepto la correspondiente a la dimensión asociada a t_i en ese espacio que vale 1.

Así, en un espacio vectorial de dimensión 5 tendríamos $t_{\text{rock}} = [1, 0, 0, 0, 0]$, $t_{\text{pop}} = [0, 1, 0, 0, 0]$, $t_{\text{jazz}} = [0, 0, 1, 0, 0]$, $t_{\text{heavy}} = [0, 0, 0, 1, 0]$ y $t_{\text{dance}} = [0, 0, 0, 0, 1]$.

De esta forma, cualquier consulta q y cualquier documento $d_j \in D$ se pueden representar en el espacio vectorial como:

$$q = \sum_{i=1}^{|V|} w_{iq} \cdot t_i$$

$$d_j = \sum_{i=1}^{|V|} w_{ij} \cdot t_i$$

donde w_{iq} y w_{ij} son los pesos asignados al término t_i para la consulta q y para cada documento d_j.

Este modelo supone que dos vectores documento que estén «cerca» en el espacio vectorial «tratan» del mismo tema. Por lo tanto, para resolver los vectores consulta buscará vectores documento cercanos en el espacio vectorial. Tal similitud puede representarse intuitivamente como la proyección de un vector sobre otro, idea que se expresa matemáticamente en términos del **producto escalar**:

$$sim(d_j, q) = d_j \cdot q$$

La métrica de similitud más utilizada es la **similitud del coseno**, que es una medida de la similitud existente entre dos vectores en los que se evalúa el valor del coseno del ángulo que forman. Dicho matemáticamente:

$$d_j \cdot q = \|d_j\| \times \|q\| \times \cos(\alpha)$$

La similitud de coseno es una función del ángulo α formado entre d_j y q en el espacio vectorial:

$$Sim_{\cos}(d_j, q) = \cos(\alpha) = \frac{d_j \cdot q}{\|d_j\| \times \|q\|} = \frac{\sum_{i=1}^{|V|}(w_{ij} \times w_{iq})}{\sqrt{\sum_{i=1}^{|V|}(w_{ij})^2}\sqrt{\sum_{i=1}^{|V|}(w_{iq})^2}}$$

La medida del coseno realiza una normalización de la longitud del vector (representada como $\|\cdot\|$), lo que permite descartar errores cuando los vectores de longitudes muy diferentes generan proyecciones insignificantes.

Algunas ventajas del modelo de espacio vectorial con respecto al modelo booleano residen en su interpretación geométrica más intuitiva y la posibilidad de ponderar las representaciones de las consultas y los documentos.

4.3. Modelo probabilístico

La recuperación de información es un proceso incierto: todo el proceso en sí mismo dista mucho de ser exacto.

Un modelo probabilístico intenta representar la probabilidad de relevancia de un documento dada una consulta, es decir, calcula la similitud entre consultas y documentos como la probabilidad de que un documento d_j sea relevante para una consulta q.

En otras palabras, sea r la relevancia (en binario) con respecto a un conjunto de documentos D en relación con una consulta q. El modelo probabilístico calcula la similitud:

$$Sim(q, d_j) = P(d_j = 1|\mathrm{q}, d_j),\ \forall d_j \in D$$

El documento que maximiza dicha probabilidad se recuperará como el mejor resultado, y a continuación se recuperarán otros documentos en orden decreciente de probabilidad de relevancia.

Cabe señalar que el conjunto de documentos con relevancia máxima es desconocido *a priori*; por lo tanto, estimar esta probabilidad no es una tarea trivial. Una estrategia consiste en calcular la probabilidad de relevancia mediante la coincidencia de términos en la consulta y en los documentos. En este punto puede llevarse a cabo una fase de procesamiento iterativo (opcionalmente utilizando la retroalimentación del usuario) con el fin de mejorar el conjunto de respuesta.

El modelo de recuperación probabilístico clásico es el modelo de independencia binario, en el que los documentos (y consultas) se representan como vectores, es decir, $d_j = [w_{1j}, ..., w_{|V|}]$ de modo que $w_{ij} = 1$ si y solo si el documento d_j contiene el término t_i, y $w_{ij} = 0$ en caso contrario. El modelo supone que las apariciones de un término en los documentos son independientes, una hipótesis que generalmente funciona en la práctica.

El modelo probabilístico tiene la ventaja de clasificar los documentos de acuerdo con su probabilidad decreciente de ser relevantes. Si además cuenta con retroalimentación de relevancia (del usuario), es ciertamente una ventaja. Sin embargo hay que estimar la relevancia inicial de los documentos, una tarea que podría no ser fácil ni precisa.

Bibliografía

Baeza-Yates, R. A.; Ribeiro-Neto, B. A. (2011). *Modern Information Retrieval—the Concepts and Technology Behind Search* (2.ª ed.). Harlow: Pearson Education.

Ceri, S.; Bozzon, A.; Brambilla, M.; Della Valle, E.; Fraternali, P.; Quarteroni, S. (2013). *Web Information Retrieval*. Heidelberg: Springer-Verlag.

Croft, W. B. (1987). «Approaches to intelligent information retrieval». *Information Proccesing & Management* (vol. 23, núm. 4, págs. 249-254).

Grossman D. A.; Frieder O. (2004). *Information Retrieval: Algorithms and Heuristics* (vol. 15). Kluwer Academic, Norwell.

Korfhage, R. R. (1997). *Information Storage and Retrieval*. Nueva York: Wiley Computer Publishing.

Manning, C. D.; Raghavan, P.; Schütze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press. Disponible en: https://nlp.stanford.edu/IR-book/

Witten, I.; Moffat, A.; Bell, T. (1994). *Managing Gigabytes: Compressing and Indexing Documents and Images*. Massachusetts: Morgan Kaufmann.