Universidad Nacional de Rosario

Facultad de Ciencias Exactas, Ingeniería y Agrimensura

Licenciatura en Ciencias de la Computación

Bases de Datos Avanzadas

Recuperación de Información

Giustozzi Franco Nazareno

2014

Indice general.

Introducción 1
1 Hacia una defincón de Recuperación de Información 2
2 Sistemas de Recuperación de Información 3
2.1 Componentes de un Sistema de Recuperación de Información 3
2.2 Evolución de los SRI 4
3 Modelos de IR 4
3 .1 Modelos Clásicos 6
3.1.1 Modelo Booleano 6
3.1.2 Modelo Espacio vectorial 8
3.1.3 Modelo Probabilístico 10
3.1.4 Comparación entre los Modelos 11
4 Evaluación de la Recuperación de Información 11
4.1 Relevancia 11
4.2 Recall y Precisión 12
4.3 Colecciones de referencia 14
5 Lenguajes de Consulta 14
5.1 Consultas basadas en Keywords 15
5.1.1 Consultas Single-Word 15
5.1.2 Consultas Contextuales 15
5.1.3 Consultas Boolenas 16
5.1.4 Consultas en Lenguaje Natural 17
5.2 Pattern Matching 17
5.3 Consultas Estructurales 18
5.3.1 Estructura fija 19
5.3.2 Hypertexto 20
5.3.3 Estructura jerárquica 20
6 Estrategias de Búsqueda 21
6.1 Expansión semántica de la consulta 22
6.1.1 Recursos lingüísticos 23
6.1.1.1 Diccionarios 23
6.1.1.2 Tesauros 24
7 Indexado y Búsqueda 24
8 Aplicación de Recuperación de Información en Sistemas Recomendadores 27
Conclusiones 29
Referencias y Bibliografía 30

Introducción

El área de la Recuperacion de Información (Information Retrieval – IR) ha crecido mucho más alla de sus primeros objetivos como indexar y buscar documentos útiles en una colección de documentos. Hoy en día, la recuperación de información incluye el modelado, clasificación y categorización, interfaces de usuario, filtrado, idiomas, etc.

En este trabajo se hace énfasis en la integración de algunas de estas tareas que están muy relacionadas con el problema de recuperación de información.

En el capítulo 1 se hace una presentación del concepto de recuperación de información, y de las diferencias que posee con otras aplicaciones en lo relacionado con la gestión y recuperación de datos.

En el capítulo 2 se define que es un sistema de recuperación de información y su evolución a lo largo de los años. Mientras que en el capítulo 3 se exponen los distintos modelos sobre los que se basan los sistemas que permiten la recuperación de la información. También, debido a la necesidad de evaluar el desempeño de un sistema de recuperación de información (SRI) en el capítulo 4 se muestran algunas medidas que permiten cuantificar su efectividad.

En el capítulo 5 se muestran los distintos tipos de lenguajes de consultas y en el capítulo 6 se discuten las técnicas para la transformación de consultas (queries) con el objetivo de mejorar la tarea de recuperación.

Las tareas de indexado y búsqueda son desarroladas en el capítulo 7. En particular se desarrolla la técnica de índice invertido.

En el capítulo 8 se describe la aplicación de la recuperación de información en búsquedas inteligentes que utilizan sistemas recomendadores para la búsqueda de objetos de aprendizaje.

1 Hacia una definición de Recuperación de Información.

Se presenta una confusión a la hora de querer dar una definición precisa de Recuperación de Información, ya que se trata de un término que suele ser definido en un sentido muy amplio. Esto produjo, que no se encuentre bien utilizado en muchas ocasiones, como por ejemplo, usado como sinónimo de Recuperación de Datos.

En [Baeza y Ribeiro, 1999], el autor asienta las diferencias entre Recuperación de Datos y Recuperación de la Información:

Un sistema de Recuperacion de datos (base de datos relacional), trata con datos que tienen una estructura y una semántica bien definidas. Permite recuperar todos los objetos que satisfacen las condiciones especificadas en una expresion regular o en una expresion del algebra relacional. Es decir, solo recupera los datos que coinciden exactamente con el patrón ingresado por el usuario. En cambio un Sistema de Recuperación de Información, recupera datos relevantes que hagan mejor coincidencia parcial con el patrón dado. Esto se debe a que la recuperación de información generalmente trata con texto en lenguaje natural, el cual no esta siempre bien estrucutrado y podría ser semanticamente ambiguo.

	Recuperación de datos	**Recuperación de Información**
Acierto	Exacto	Parcial, el mejor
Inferencia	Algebraica	Inductiva
Lenguaje de Consulta	Fuertemente estructurado	Estrucuturado o natural
Especificación consulta	Precisa	Imprecisa

Tabla 1 : Recuperación de datos vs. Recuperación de Información. [van Rijsbergen, 1979]

A continuación se presentan algunas de las definiciones de Recuperación de Información propuestas por algunos autores destacados.

Según Grossman y Frieder, en [Grossman y Frieder, 1998] indican que recuperar información es “encontrar documentos relevantes, no encontrar simples correspondencias a unos patrones de bits”. Esta definición está muy influenciada por la informática, cuya evolución hizo que se llegase a olvidar que se puede recuperar información sin recursos informáticos (aunque no es lo más común en la actualidad).

Ricardo Baeza-Yates plantea en [Baeza y Ribeiro, 1999] que “la recuperación de información trata con la representación, el almacenamiento, la organización y el acceso a items de informacíon”. Además incorpora la siguiente reflexión: “la representación y organización debería proveer al usuario un fácil acceso a la información en la que está interesado. Desafortunadamente la caracterización de la necesidad informativa de un usuario no es un problema sencillo de resolver”.

Otro autor, Croft estima en [Croft, 1987] que la recuperación de información es “el conjunto de tareas mediante las cuales el usuario localiza y accede a los recursos de informacion 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.”.

Otra corriente de autores, no se preocupó demasiado en dar una definición concreta de Recuperación de Información, sino que profundizaron más en la explicación de los Sistemas de Recuperación de Información (SRI).

De la observación de las definiciones mencionadas se determina que la Recuperación de la Información no es algo aislado vinculado al acceso de la información, sino que constituye un proceso que incluye también la representación, la organización, la búsqueda y la localización de la información. Es decir, la Recuperación de Información intenta resolver el problema de: encontrar y rankear documentos relevantes que satisfagan la necesidad de información de un usuario, expresada en un determinado lenguaje de consulta.

2 Sistemas de Recuperación de Información.

Primero resulta necesario definir adecuadamente Sistema de Recuperación de Información.

2.1 Componentes de un Sistema de Recuperación de Información.

Cualquier Sistema de Recuperación de Información puede ser descripto como un conjunto de ítems de información (DOCS), un conjunto de peticiones (REQS) y algún mecanismo que determine qué ítems satisfacen las necesidades de información expresadas por el usuario en la consulta.

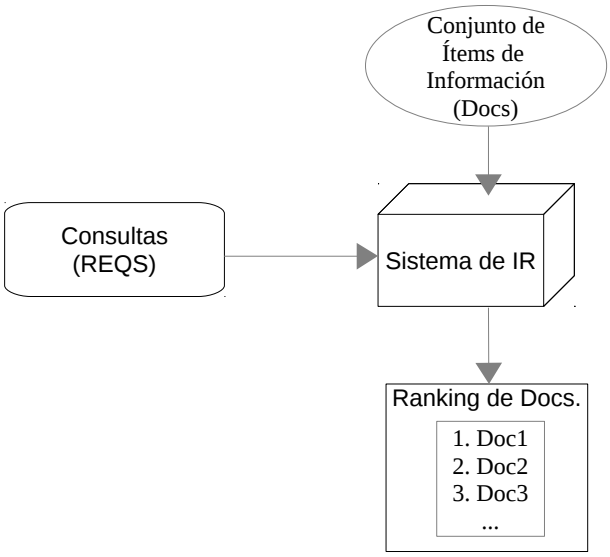


Figura 1: Arquitectura de un SRI.

El SRI trabaja con esos items de información realizando operaciones sobre ellos, tales como, remoción de palabras no significativas y stemming, para construir un archivo invertido.
Además, una interface de usuario que permita al usuario ingresar la consulta y la visualización de los resultados.
El SRI realiza modificaciones a la consulta para mejorar los resultados, por ejemplo, exapansión de la consulta y feedback de relevancia.
Los resultados se muestran al usuario con un orden (ranking) de todos los documentos recuperados según su relevancia.

2.2 Evolución de los SRI

Baeza-Yates destaca tres fases fundamentales en la evolución de estos sistemas [Baeza y Ribeiro, 1999]

1. *Desarrollos iniciales:* Ya existían métodos de recuperación de información en las antiguas colecciones de papiros. Otro ejemplo clásico que se ha venido utilizando es la tabla de contenidos de un libro, sustituida por otras estructuras más complejas a medida que ha crecido el volumen de información. La evolución lógica de la tabla de contenidos es el índice, estructura que aún constituye el nucleo de los SRI actuales.
2. *Recuperación de información en las bibliotecas:* Fueron las primeras instituciones en adoptar estos sistemas. Originalemente fueron desarrollados por ellas mismas y posteriormente se ha crado un mercado informático altamente especializado, en el que participan empresas e instituciones.
3. *La WORLD WIDE WEB:* la evolución lógica de los sistemas de recuperación de información ha sido gracias a la Web, donde han encontrado una alta aplicación práctica y un aumento en el número de usuarios, especialmente en el campo de motores de búsqueda. La consolidación de la web está siendo favorecida por el gran abaratamiento de las tecnologías informáticas, por el desarrollo de las telcomunicaciones y por la facilidad de publicación de cualquier documento que un autor considere interesante, sin tener que pasar por las editoriales.

Resulta necesario destacar que este proceso de evolución no es un proceso finalizado, sino más bien un proceso en realización.

3 Modelos de IR

El diseño de un SRI se realiza bajo un modelo, que es una idealización o abstracción del proceso real de recuperación, donde queda definido como se obtienen las representaciones

de los documentos y de la consulta, la estrategia para evaluar la relevancia de un documento respecto a una consulta y los métodos para establecer la importancia (orden) de los documentos de salida.

Baeza-Yates clasifica los modelos de recuperación de información en base a la tarea que realiza el usuario en el sistema: recuperar información por medio de una consulta (Recuperación) o dedicar un tiempo a navegar (browse) los documentos en la búsqueda.

A su vez divide a los modelos basados en la recuperación en dos grupos: clásicos y estructurados. En los clásicos incluye a los modelos booleanos, espacio vectorial y probabilístico. Posteriormente, presenta una serie de alternativas para cada modelo: conjuntos difusos y booleano extendido, redes neuronales, redes de inferencia. Los modelos estructurados corresponden a listas sin solapamiento y a nodos próximos (son modelos que no se tratan en este trabajo).

Los modelos basados en navegación son de tres tipos: estructura plana, estrucutra guiada e hipertexto. El primero es un lectura de un documento aislado del contexto, el segundo incorpora la posibilidad de facilitar la exploración organizando los documentos en una estructura con jerarquía de clases y subclases, y el tercero se basa en la idea de un sistema de información que da la posibilidad de adquirir información de forma no secuencial sino a través de nodos y enlaces.

<table>
<tr><td colspan="4"></td><td>Extensiones</td></tr>
<tr><td rowspan="10">Tarea del Usuario</td><td rowspan="8">Recuperación</td><td rowspan="6">Modelos Clásicos</td><td rowspan="2">Boolean</td><td>Conjuntos difusos</td></tr>
<tr><td>Booleano extendido</td></tr>
<tr><td rowspan="2">Espacio Vectorial</td><td>Vector generalizado</td></tr>
<tr><td>Redes neuronales</td></tr>
<tr><td rowspan="2">Probabilístico</td><td>Redes de inferencia</td></tr>
<tr><td>Redes de inferencia</td></tr>
<tr><td rowspan="2">Modelos Estructurados</td><td>Non-Overlapping List</td><td rowspan="5"></td></tr>
<tr><td>Proximal Nodes</td></tr>
<tr><td rowspan="3">Navegación</td><td rowspan="3">Navegación</td><td>Flat</td></tr>
<tr><td>Structure</td></tr>
<tr><td>Hypertext</td></tr>
</table>

TABLA 2: Modelos de IR

Ya vista la clasificación de los modelos, se da una definición más formal de modelo.

Definición: Un modelo de Recuperación de Información es una cuadrupla $(D, Q, F, R(q,d))$, donde:

- D es una representación de la colección de documentos.
- Q es una representación de la información que necesita el usuario.

- F es el entorno de trabajo para modelar la colección de documentos, las consultas y las relaciones que hay entre ellos.
- $R\ (q,d)$ es una función que devuelve un número real que permite asociar la consulta $q\ (q\ \epsilon\ Q)$ y la representación de la colección de documentos $d\ (d\ \epsilon\ D)$. Ese número, representa la probabilidad de que d resulte relevante para q.

Vale la pena aclarar que un Sistema de Recuperación de Información no trabaja con los documentos propiamente dichos, sino con una representación más manejable de los mismos.

Baeza-Yates presenta en detalle cada uno de los modelos en [Baeza y Ribeiro, 1999]. En este trabajo se presentan únicamente los modelos clásicos.

3.1 Modelos Clásicos

Los modelos clásicos, consideran que cada documento se describe por un conjunto de palabras claves llamadas *index terms*. Un *index term* es una palabra cuya semántica ayuda a recordar los temas principales del documento, es decir los *index terms* se usan para indicar los contenidos del documento. En general, los *index terms* son sustantivos ya que estos tiene significado por si solos.

Ahora veamos que dado un conjunto de *index terms* para un documento, no todos los términos son útiles para describir el contenido del docuemento. Por ejemplo, considere cientos de miles de documentos. Una palabra que aparezca en todos estos documentos es inútil como un *index term*, en cambio una palabra que aparezca en diez documentos sería útil ya que reduce bastante el número de documentos. Por lo tanto, es evidente que cada *index term* tiene un grado de relevencia. Ésto se expresa asignando diferentes *pesos* a esos *index terms* en cada documento.

A continuación, se da la definición formal:

Definición: Sea t el número de *index terms* en el sistema, sea k_i un *index term* y $K = \{k_1,..., k_t\}$ el conjunto de todos los *index terms*. Se asocia un peso w_{ij} a cada *index term* k_i de un documento d_j. Para un *index term* que no aparece en un documento se tiene $w_{ij} = 0$. Con un documento d_j esta asociado un vector representado por $\vec{d_j} = (w_{1j},w_{2j},...,w_{ij})$.

3.1.1 Modelo Booleano

El modelo Booleano es un modelo simple, basado en la teoría de conjuntos y el álgebra de Boole.

Los documentos se representan por el conjunto de términos contenidos en ellos.

Las consultas se expresan como expresiones booleanas entre términos, usando los operadores lógicos más conocidos (AND, OR y NOT). Esto permite que las consultas tengan una semántica concreta. Por ejemplo, sean t_1 y t_2 dos términos. Entonces:

- t_1 AND t_2 = conjunto de documentos cuyas representaciones contienen al término t_1 y al término t_2.

- t_1 OR t_2 = conjunto de documentos cuyas representaciones contienen al término t_1 o al término t_2.

- NOT t_1 = conjunto de documentos cuyas representaciones no contienen al término t_1.

Desafortunadamente, el modelo booleano sufre una desventaja importante. Pues su estrategia de recuperación está basada en un criterio de decisión binario, es decir un documento es relevante o no es relevante, no posee la noción de importancia relativa con respecto a una consulta. Esto sucede debido a que un index term puede tener peso 1 (si está en el documento) o peso 0 (si no está en el documento). Además se muestra un ejemplo que ayudará para entender mejor lo dicho.

La ventaja principal de este modelo es que posee un formalismo muy simple y una semántica clara y concisa para la formulación de las consultas. Las desventajas son: (1) provee escaso control sobre el volumen del resultado producido por una consulta concreta, por lo que se hacen necesarias reformulaciones de la pregunta para lograr un volumen aceptable de resultados. (2) Todos los registros recuperados son supuestamente de la misma utilidad para el usuario, no existen mecanismos que permitan ordenarlos en función de su relevancia, por lo que no permite reflejar la importancia relativa de los diferentes componentes de la pregunta ya que todos los términos tienen un peso 1 o 0, dependiendo de si están o no presentes en la pregunta.

Algunas extensiones de este modelo son: el modelo booleano extendido y el modelo booleano usando conjuntos difusos y la lógica difusa (fuzzy).

Veamos un ejemplo:

Documento 1: “los coches tienen ruedas y circulan por cualquier vía”

Documento 2: “por la autopista pueden circular coches, motos...”

Los términos son:

K={*coches, ruedas, vía, autopista, motos*}

Los documentos quedan descriptos por (los pesos):

$\overline{d}_1$ = *(1,1,1,0,0)*

$\overline{d}_2$ = *(1,0,0,1,1)*

Algunas consultas:

q_1 : $ruedas$ AND ($autopista$ OR $coches$) = $\overline{d}_1$

q_2 : $coches$ AND $motos$ = $\overline{d}_2$

3.1.2 Modelo Espacio Vectorial

El modelo espacio vectorial utiliza un mecanismo a través del cual es posible un matching parcial. Logra esto mediante la asignación de pesos no binarios a los index terms en la consulta y en la representación de los documentos. Estos pesos se usan para calcular el grado de similitud entre cada documento y la consulta del usuario, permitiendo ordenar los documentos recuperados en orden decreciente con respecto al grado de similitud. El resultado principal de esto es el rankeo mucho más preciso (mejor coincidencia con la necesidad de información del usuario) comparado con el del modelo booleano.

Un dodumento d_j y una consulta q son representados como un vector con tantas dimensiones como index terms se tengan. Cada componente del vector representa al index term correspondiente. El modelo evalúa el grado de similaridad del documento d_j y de la consulta q mediante la comparación de los vectores $\overline{d}_j$ y $\overline{q}$. Esta comparación se puede cuantificar, por ejemplo usando el coseno del ángulo entre los dos vectores. Es decir,

$sim\ (\overline{d}_j, \overline{q})$ = ($\overline{d}_j$ • $\overline{q}$) / (| $\overline{d}_j$ | x | $\overline{q}$ |), donde • es la operación producto interno.

Existen varios métodos para cuantificar la comparación entre los vectores. Otras formas son por ejemplo: Coeficiente de Dice, Coeficiente de Jaccard, etc.

Otro asunto, no menor, es la asignación de los pesos a los términos. Un esquema habitual está basado en consideraciones estadísticas para representar la importancia relativa de un término dentro de un documento. Se tiene en cuenta la cantidad de veces que aparece un término en un documento (***tf*** - term frequency), suponiendo que las palabras que más aparecen son mas representativas del contenido del mismo. Se define así:

$$\boldsymbol{tf}\ (t_i, d_j) = freq\ (t_i, d_j) / max\ freq\ (t_i, d_j)$$

(Número de veces que un término aparece en un documento, normalizado por la máxima frecuencia de cualquier término en ese documento).

Pero cuando se considera como peso de un término directamente a su frecuencia de aparición surge el siguiente inconveniente, se le confiere igual importancia a todos los términos que aparecen en la colección. Si se supone que los términos que aparecen en pocos documentos son buenos discriminadores y que los términos más comunes son menos

útiles a la hora de decidir la relevancia de un documento, es natural asignar a los primeros un peso más alto que a los segundos. Para resolver este problema, se utiliza un esquema de asignación de pesos denominado *TF-IDF* (Term frecuency - Inverse Document Frecuency) y es uno de los más utilizados dentro de los sistemas de recuperación de información construidos usando el modelo espacio vectorial.

La frecuencia de documento inversa (IDF: Inverse Document Frecuency) de un término está relacionada con la cantidad de documentos de la colección en que aparece dicha palabra. Como el término se considera más importante cuando aparece en menos documentos, la frecuencia de documento inversa se define como: $\boldsymbol{idf}(t_i) = \log(N/n_i)$ donde N es la cantidad total de documentos y n_i es el número de documentos en los que aparece el término t_i. El logaritmo se incluye simplemente para evitar el crecimiento númerico de la función. De esta manera, un término que aparezca en todos los documentos de la colección tendrá una ***idf*** igual a cero, indicando que carece de valor para ser un buen discriminador.

El peso de un término t_i dentro de un documento d_j se define entonces así:

$$w(t_i, d_j) = \boldsymbol{tf}(t_i, d_j) \times \boldsymbol{idf}(t_i)$$

Utilizando este esquema, la importancia de cada término influirá en el ranking de documentos recuperados frente a una consulta. Veamos un ejemplo.

Supongamos que un SRI cuenta con un documento que contiene este texto: “... La República Argentina ha sido nominada para la realización del X Congreso Americano de Epidemiología en Zonas de Desastre. El evento se realizará ...”. Un usuario realiza la consulta ”Argentina congreso epidemiología”. En la tabla siguiente se observa parte de la matriz término-documento con pesos normalizados entre 0 y 1. La última fila de la Tabla3 es la consulta del usuario.

	argentina	...	**congreso**	**epidemiología**	...
d_1	0.5		0.3	0.2	
...					
d_j					
q	0.4		0.3	0.3	

Tabla 3: tabla término-documento con pesos entre 0 y 1.

La similitud entre la consulta y el documento se mide con el valor del coseno del ángulo entre ambos vectores.

$$\boldsymbol{sim}(\vec{d}_1, \vec{q}) = (\vec{d}_1 \bullet \vec{q}) / (|\vec{d}_1| \times |\vec{q}|) =$$

$$= (0.5*0.4 + 0.3*0.3 + 0.2*0.3) / (0.5^2*0.3^2*0.2^2) * (0.4^2*0.3^2*0.3^2) = \mathbf{0.504}$$

Las ventajas de este modelo son muchas, ha obtenido muy buenos resultados en experimentos y aplicaciones, sobre todo con grandes colecciones de documentos. Y la principal es que permite tener en cuenta el grado de relevancia de los documentos recuperados y permite matching parcial entre la consulta y los documentos. Una desventaja es que no tiene en cuenta información de contexto y términos ambiguos.
Algunas extensiones de este modelo son: el modelo vectorial generalizado y el uso de las redes neuronales.

3.1.3 Modelo Probabilístico

El modelo probabilístico clásico fue introducido en 1976 por Robertson y Spark Jones [Baeza y Ribeiro, 1999]. Este modelo también se conoce como modelo de recuperación de independencia binaria o BIR. Está basado en la teoría de probabilidades. Utiliza una representación binaria de los documentos, de forma que un documento cualquiera será representado como un conjunto de unos y ceros que indican la presencia o ausencia de los términos de indización. Las consultas se expresan como una enumeración de términos.
La idea fundamental de este modelo consiste en asumir que "para una determinada consulta q de usuario existe un conjunto de documentos (llamado conjunto ideal) que comprende exactamente los documentos relevantes y no otros". El problema de esto es que no resulta simple describir con certeza las características iniciales de este conjunto. Por lo tanto, resulta necesario realizar ciertas supociones sobre las características de dicho conjunto e intentar refinar dichas características consulta tras consulta. Es decir, luego de cada consulta el usuario identificará los documentos recuperados que resultaron relevantes, con lo que se refinará la descripción del conjunto ideal.
Podemos expresar la función de relevancia que estima el parecido de un documento con una consulta en términos de probabilidad como el resultado de dividir la probabilidad de que un documento sea relevante a la consulta lanzada por el usuario entre la probabilidad de que dicho documento no sea relevante a la consulta. Esto se puede formalizar como:

$$sim(d_j,q) = \frac{P(R \mid d_j)}{P(R' \mid d_j)} = \frac{\frac{P(d_j \mid R) \times P(R)}{P(d_j)}}{\frac{P(d_j \mid R') \times P(R')}{P(d_j)}} = \frac{P(d_j \mid R)}{P(d_j \mid R')}$$

donde:

$P(R|d_j)$ es la probabilidad de que el documento d_j sea relevante a la consulta q.

$P(R'|d_j)$ es la probabilidad de que el documento d_j no sea relevante a la consulta q.

$P(d_j|R)$ es la probabilidad de seleccionar al documento d_j de entre los relevantes .

$P(R)$ es la probabilidad de que seleccionando algún documento aleatoriamente de la

colección, sea relevante.

$P(d_j)$ es la probabilidad de obtener el documento d_j aleatoriamente seleccionando uno de entre toda la colección.

$P(R'|d_j)$, $P(d_j|R')$, $P(R')$ son los análogos, aplicados a la no relevancia .

Entonces, un documento d_j será considerado como relevante si:

$$P(R|d_j) > P(R'|d_j) \quad \text{ó} \quad P(d_j|R) > P(d_j|R')$$

De esta manera, se pueden ordenar los documentos de la colección en orden descendente de probabilidad de relevancia en relación a la consulta.

Las principales ventajas que ofrece este modelo son que ordena los resultados por relevancia y el hecho de que utiliza un razonamiento matemático basado en probabilidades lo que permite que tenga extensiones populares. Como redes bayesianas y redes de inferencias bayesianas. Sin embargo, posee varias desventajas, asigna pesos binarios a los index terms, supone independencia de términos y no es posible conocer de antemano el conjunto de documentos relevantes.

3.1.4 Comparación entre los modelos

El modelo Booleano es considerado el más débil de los modelos clásicos. Su principal problema es la incapacidad de reconocer matching parciales, lo que lleva a pobres performances. En [Baeza y Ribeiro, 1999] el autor destaca que Croft realizó algunos experimentos que retornaron que el modelo probabilístico tiene una mejor performance que el modelo espacio vectorial. Sin embargo, luego Salton en [Salton y Mc Gill, 1983] y Buckeley refutaron esos resultados. Esto parece ser lo dominante, ya que en la comunidad de la Web el modelo espacio vectorial es el más utilizado.

4 Evaluación de la Recuperación de Información

4.1 Relevancia

Antes de la implementación final de un sistema de recuperación de la información, se lleva a cabo una evaluación del sistema. El tipo de evaluación depende de los objetivos que tenga el sistema de recuperación. Por supuesto, cualquier sistema de software debe proveer la funcionalidad para la cual fue concebido. Por lo tanto, el primer tipo de evaluación será probar las funcionalidades una por una. Una vez realizada esta evaluación se procede con la evaluación de la performance del sistema.

En los sistemas de recuperación de datos, el tiempo de respuesta y el espacio requerido son las métricas consideradas.

En los sistemas de recuperación de información, otras métricas, además del tiempo de respuesta y el espacio requerido, son también de interés. Baeza-Yates en [Baeza y Ribeiro, 1999] señala tres criterios de evaluación de los sistemas de recuperación de información:

- **Eficacia en la ejecución:** medida del tiempo que tarda un sistema de recuperación de información en realizar una operación.
- **Eficiencia del almacenamiento:** medida del espacio que se precisa para almacenar los datos.
- **Efectividad en la recuperación de la información:** medida del éxito en satisfacer la demanda de información de los usuarios, basada en la relevancia. Ésta no está presente en la evaluación de sistemas de recuperación de datos.

El último criterio mencionado plantea ciertas inquietudes. Ya que un documento recuperado es relevante cuando el contenido del mismo se ajusta a la necesidad de información del usuario (relevancia objetiva). Pero, los juicios de relevancia son realizados por los usuarios, esto hace dificultoso determinar el grado de relevancia del documento, pues un mismo documento puede ser considerado relevante o no por dos personas distintas, ya sea por motivos de la búsqueda o grado de conocimiento, incluso se puede recibir distinta evaluación por el mismo usuario en dos momentos diferentes. Por eso parece más apropiado utilizar el término pertinencia, relación de utilidad entre un documento recuperado y una necesidad de información individual (relevancia subjetiva).

La evaluación del sistema, se basa en un test reference collection (conjunto de documentos, conjunto de consultas y conjunto de documentos relevantes para cada consulta) y en una medida de evaluación. De tal manera que, dada una estrategia de recuperación $\mathcal{E}$, la evaluación cuantifica la similaridad entre el conjunto de documentos obtenidos usando $\mathcal{E}$ y el conjunto de documentos relevantes para esa consulta.

Existen muchas métricas para medir cuantitativamente la performance de los sistemas de recuperación de información clásicos [Baeza y Ribeiro, 1999] [Losse,1998]. En en este trabajo se explican sólo dos, las más utilizadas. Estas son *Recall* y *Precisón*.

4.2 Recall y Precisión

Las medidas recall y precisión son definidas de la siguiente forma:

- **Recall:** es la fracción de documentos relevantes que han sido recuperados.
- **Precisión:** es la fracción de documentos recuperados que resultaron relevantes.

Tanto Recall como Precisión, según la definicion dada, asumen que todos los documentos del conjunto respuesta a una consulta han sido examinados o vistos. Pero en realidad lo que se hace es primero ordenar los documentos del conjunto respuesta usando algún ranking de

relevancia. Así, el usuario recorre la lista de arriba hacia abajo, por lo que las medidas de recall y precisión varían. Por ejemplo, cuando esté más abajo la precisión va a decaer.

Para dejar más en claro los conceptos consideremos lo siguiente:

Sea $\boldsymbol{q}$ una consulta (de una colección de referencia) y su correspondiente conjunto $\boldsymbol{R}$ de documentos relevantes. Supongamos que dada una estrategia de recuperación (la cual es evaluada) procesa la consulta $\boldsymbol{q}$ y genera el conjunto respuesta $\boldsymbol{A}$.

Luego,

$$Recall = |\boldsymbol{R} \cap \boldsymbol{A}| \,/\, |\boldsymbol{R}| \qquad \text{(F1)}$$

$$Precisión = |\boldsymbol{R} \cap \boldsymbol{A}| \,/\, |\boldsymbol{A}| \qquad \text{(F2)}$$

Los resultados de estas operaciones están entre 0 y 1.

Cuando el resultado de (F1) más se acerque a 0, mayor será el número de documentos recuperados que no le sean útiles al usuario. Si el resultado de (F2) es 1 significa que se obtuvo la exhaustividad máxima, es decir, se encontró todo lo relevante que había, por lo que la recuperación es perfecta.

Estas medidas se comportan de manera antagónica, ya que cuando se incrementa la recuperación tiende a disminuir la precisión, y viceversa.

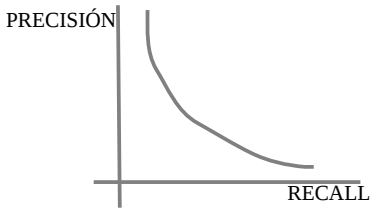


Figura 2: relación precision y recall.

Es decir, búsquedas específicas obtienen resultados muy precisos, pero habrán perdido documentos por ese alto nivel de especificación. Se reduce el Recall. Por ejemplo,

$\boldsymbol{q}_1$:“contaminación de agua en los ríos”

$\boldsymbol{q}_2$: “contaminación en los ríos”

Por otro lado, búsquedas generales recuperan la mayoría de los documentos relevantes, con el tema, pero también otros que no lo son. Se reduce la Precisión. Por ejemplo,

$\boldsymbol{q}_1$:“contaminación”

$\boldsymbol{q}_2$: “contaminación en los ríos”

Para que una búsqueda sea óptima el conjunto de documentos recuperados debe coincidir totalmente con el conjunto de documentos relevantes para la consulta realizada. Es decir, cuando todos los documentos recuperados sean relevantes y todos los documentos relevantes sean recuperados.

Dependiendo de la aplicación del sistema de recuperación de información, será requerida

mayor precisión y menor recall, o menor precisión y mayor recall.
Estos indicadores, se aplican a la recuperación de información en la Web. En el contexto de la web, los motores de búsqueda generalista (google[1], etc.) proporcionan altas tasas de exhaustividad (recuperan muchos documentos relevantes), pero su tasa de precisión es muy baja, ya que sólo una parte muy reducida de los documentos recuperados es relevante.
Los sistemas muy especializados, como las agencias de selección y evaluación de recursos digitales (Go-Geo[2], Intute[3], etc.), proporcionan muchos menos recursos, probablemente tasas de exhaustividad muy bajas, pero las tasas de precisión se aproximan al 100%.

4.3 Colecciones de referencia

Existen muchas colecciones de referencia que han sido utilizadas para la evaluación de los sistemas de recuperación. Una de ellas es la colección TREC (Text Retrieval Conference), la cual es una de las más destacadas debido a su gran tamaño y a su experimentación minuciosa. Otras colecciones son CACM (Communications of the ACM) y ISI (Institute of Scientific Information), tambíen consideradas de gran importancia. En [Baeza y Ribeiro, 1999] el autor analiza con más detalle diferentes colecciones de referencia.

5 Lenguajes de Consulta

En esta sección se tratarán los diferentes tipos de consultas que normalmente se le plantean a los sistemas de recuperación de la Información, en la siguiente sección se discutirá como se resuelven esas consultas. Las consultas planteadas dependen en parte del modelo que adopta el sistema (capítulo 3).
Un punto importante a tener en cuenta es que la mayoria de los lenguajes de consulta tratan de usar el contenido (semántica) y la estructura del texto (sintaxis) para encontrar documentos relevantes. En este sentido, el sistema podría fallar en encontrar respuestas relevantes , como ya se mencionó anteriormente en la sección 4. Por esta razón, se usan varias técnicas que intentan mejorar esto. Como por ejemplo, la expansión de una palabra a un conjunto de sinónimos o el uso de tesauros o stemming o eliminación de stopwords.
Primero se tratan las consultas que se pueden formular con Keyword-based query languages, que tienen como objetivo recuperar información incluyendo palabras simples, frases y combinaciones de éstas usando operadores booleanos.
Segundo, se exponen consultas que usan pattern matching, que incluye consultas más complejas y generalmente tienen como objetivo complementar las búsquedas con mayores

1 www.google.com
2 *www.*gogeo.*ac.uk*
3 *www.*intute.*ac.uk*

capacidades de recuperación de datos.
Por último, se abordan consultas sobre la estructura del texto.

5.1 Consultas basadas en Keywords

En su forma más simple, una consulta está compuesta por una combinación de palabras que se utilizan para determinar cuáles documentos son relevantes. En general, son fáciles de expresar. Se las puede clasificar en consultas de una palabra, consultas contextuales, consultas booleanas y consultas en lenguaje natural.

5.1.1 Consultas Single-Word

La consulta más simple que se puede formular en un sistema de recuperación de documentos es simplemente una palabra. Los documentos de textos se asumen como grandes secuencias de palabras. Algunos modelos pueden ver la división interna de palabras en letras, veremos que estos últimos permiten la búsqueda de otro tipo de patrones (pattern matching).
Una palabra es una secuencia de letras, rodeada por espacios. Modelos más complejos permiten especificar otro tipo de caracteres que nos son letras ni separan una palabra. Por ejemplo, el caso de la palabra on-line.
La división del texto en palabras no es arbitraria, ya que las palabras llevan significado en lenguaje natural. Debido a esto, muchos modelos están completamente estructurados en el concepto de palabra y palabras son las únicas consultas permitidas (modelo vectorial).
El resultado de consultas de palabras puede ser el conjunto de documentos que contienen al menos una de las palabras de la consulta, y el ranking se arma siguiendo algún método, por ejemplo , $TF\text{-}IDF$. O también, puede ser el conjunto de documentos que contienen todas las palabras de la consulta. Esto es útil cuando el número de documentos recuperados para una sola palabra resulta muy grande.

5.1.2 Consultas Contextuales

Muchos sistemas complementan el anterior con la habilidad de buscar palabras en un contexto dado, esto es, cerca de otras palabras. Es decir, palabras cercanas a otras pueden tener más relevancia que si aparecen en otra parte.
Supongamos que se quiere encontrar palabras que estén próximas en el texto. Se distinguen dos tipos de consultas:

- **Phrase:** es una secuencia de consultas single-word. Una ocurrencia de la phrase es una secuencia de palabras. En este tipo de consultas normalmente se entiende que

los separadores en el texto no necesariamente son los mismos que en la consulta (por ejemplo, dos espacios en blanco contra uno) y las palabras no interesantes no son consideradas. Aunque la noción de phrase es una característica muy útil en la mayoría de los casos, no todos los sistemas la implementan.

- **Proximity:** es una versión más relajada que las consultas phrase. En este caso, se da una secuencia de palabras o frases, con la distancia máxima permitida entre ellas. La distancia puede ser medida en caracteres o palabras dependiendo el sistema. Las palabras y las frases pueden aparecer en el mismo orden o no que en la consulta, según se requiera.

Las consultas phrases se pueden rankear de la misma forma que las consultas de single-word.

Las consultas de proximity pueden rankearse de la misma manera si no dependen de la distancia física. No está claro como hacer un mejor ranking, la distancia física tiene un significado semántico. Esto es porque en la mayoria de los casos la proximidad siginifica que las palabras están en un mismo párrafo y por lo tanto están relacionadas de alguna manera.

5.1.3 Consultas Booleanas

La forma más antigua y aún usada para combinar consultas es usar operadores booleanos.

Una consulta booleana tiene una sintaxis compuesta por átomos (consultas básicas) que devuelven documentos, y operadores booleanos que trabajan sobre conjuntos de documentos y devuelven conjuntos de documentos. Claramente, este es un esquema composicional, por lo que se define un árbol de sintaxis de la consulta, donde las hojas corresponden a las consultas básicas y los nodos internos son los operadores booleanos.

Estos son los operadores booleanos: OR, AND, BUT y NOT.

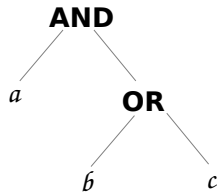


Figura 3: Árbol - va a recuperar todos los documentos que contengan la palabra a y, la palabra b o c.

En la sección 3.1 se mencionó que los sistemas booleanos clásicos no hacen un ranking de los documentos recuperados. Un documento satisface una consulta booleana (es devuelto) o no la satisface (no es devuelto). Esta es una limitación porque no permite el matcheo parcial entre el documento y la consulta del usuario.

Para sobreponerse a esta limitación, lo que se hace es relajar en cierto modo a los

operadores booleanos. Por ejemplo, un documento que parcialmente satisface un AND sera recuperado.

En [Baeza y Ribeiro, 1999] se menciona la propuesta de un conjunto fuzzy-Booleano de operadores. La idea principal es relajar el significado de AND y OR, es decir, en lugar de forzar a un elemento a aparecer en todos los operandos (documentos) (con AND) o al menos en uno de los operandos (con OR), estos devuelven documentos donde aparecen algunos de los elementos. De esta manera, se pueden rankear los documentos cuando tienen mayor número de elementos en común con la consulta.

5.1.4 Consultas en Lenguaje Natural

Si se relaja aún más el significado de los operadores del conjunto fuzzy-Booleano. La distinción entre AND y OR se puede confundir completamente, por lo que una consulta se transforma en una consulta con una enumeración de palabras y contextos.

Para manejar el operador negación (NOT) se permite que el usuario exprese algunas palabras que no desea. Y luego, los documentos que contienen esas palabras, son penalizados a la hora de hacer el ranking .

Todos los documentos que matchean con la consulta del usuario son recuperados. Están rankeados más altos aquellos que matchean con más partes de las consultas. Los documentos que no matchean con tantas partes directamente no son recueprados.

De esta manera, lo que se hace es eliminar cualquier referencia a un modelo booleano, entrando en el campo de consultas en lenguaje natural.

Se puede considerar a las consultas booleanas como una abstracción simplificada a las consultas en lenguaje natural.

5.2 Pattern Matching

En esta sección se tratan formulaciones más específicas de consultas, basadas en el concepto de patrón, no en el concepto de palabras. Estas consultas permiten la recuperación de pedazos de textos que tienen alguna propiedad. Los cuales pueden ser utilizados para formular consultas de frases o de proximidad, como las anteriormente descriptas.

Un patrón es un conjunto de características sintácticas que deben ocurrir en un segmento de texto. Aquellos segmentos que satisfacen la especificación del patrón se dicen que “matchean” con el patrón.

Cada sistema permite la especificación de diferentes tipos de patrones. En general, mientras más poderoso sea el conjunto de patrones permitidos por el sistema, mejores consultas

podrá formular el usuario. Los tipos de patrones más usados son:

- **Words:** Es el patrón más básico, una cadena de caracteres que está en el texto.
- **Prefijos:** Una cadena que debe formar el principio de una palabra del texto. Por ejemplo, dado el prefijo 'comput' todos los documentos que contengan las palabras como 'computadora', 'computación', 'computar', etc. son devueltos.
- **Sufijos:** Una cadena que debe formar el fin de una palabra del texto. Por ejemplo, dado el sufijo 'logia' todos los documentos que contengan las palabras que terminan en 'logia' serán recuperados.
- **Subcadenas:** Una cadena que puede aparecer dentro de una palabra del texto, además se puede extender a subcadenas que pertenezcan a una parte del texto.
- **Rangos:** Una cadena que matchee cualquier cadena que esté entre dos cadenas (orden lexicográfico).
- **Allowing errors:** Una palabra junto con un umbral de error. Este permite la recuperación de todos los documentos cuyas palabras sean “similares” a la dada. Es útil principalmente para el caso en que el texto tenga errores, ya sea de tipeo, o traducción.
- **Expresiones Regulares:** Un patrón bastante general construido por cadenas simples. Se combina a esas cadenas simples usando los siguientes operadores:
  - *unión*: si $\mathbf{e_1}$ y $\mathbf{e_2}$ son expresiones regulares, entonces ($\mathbf{e_1}$ | $\mathbf{e_2}$) matchea con lo que $\mathbf{e_1}$ y $\mathbf{e_2}$ matchean.
  - *concatenación: si* $\mathbf{e_1}$ y $\mathbf{e_2}$ son expresiones regulares, las ocurrencias de ($\mathbf{e_1}$ $\mathbf{e_2}$) están formadas por las ocurrencias de $\mathbf{e_1}$ inmediatamante seguidas por la de $\mathbf{e_2}$.
  - *repetición: si* **e** es una expresión regular, luego (**e***) matchea con una secuencia de cero o más ocurrencias contiguas de **e**.
- **Patrones extendidos:** Un lenguaje de consulta más fácil de usar para representar algunos casos comunes de expresiones regulares. Son un subconjunto de las expresiones regulares que se puede representar con una sintaxis más simple. Algunos ejemplos son:
  - *expresiones condicionales*: una parte de un patrón puede aparecer o no.
  - *combinaciones* que permitan que algunas partes del patrón matcheen exactamente y otras no.

5.3 Consultas Estructurales

En las secciones anteriores sólo se consideró a la colección de texto como un conjunto de documentos que pueden ser consultados con respecto a su contenido. Este modelo es

incapaz de usar las nuevas ventajas que están apareciendo, como la estructura del texto.
Las colecciones de textos tienden a tener alguna estructura, la cual permite al usuario hacer consultas basadas en esas estructuras. Por ejemplo, HTML (HyperText Markup Language) es un lenguaje para representar la estructura de los textos.
Mezclar contenido y estructura permiten hacer consultas muy poderosas. Por lo que si se utiliza un lenguaje que soporte ambas, se mejora notablemente la recuperación.
Este mecanismo está construido sobre consultas básicas, entonces estas seleccionan un conjunto de documentos que satisfacen ciertas restricciones en sus contenidos (expresadas usando palabras, frases o patrones que los documentos deben tener). Y sobre esto, se pueden expresar algunas restricciones sobre los elementos de la estructura de los documentos (capitulos, secciones, etc).
Los tres tipos de estructura principales son:

a) Estructura Fija.

b) Hypertexto.

c) Estructura Jerárquica.

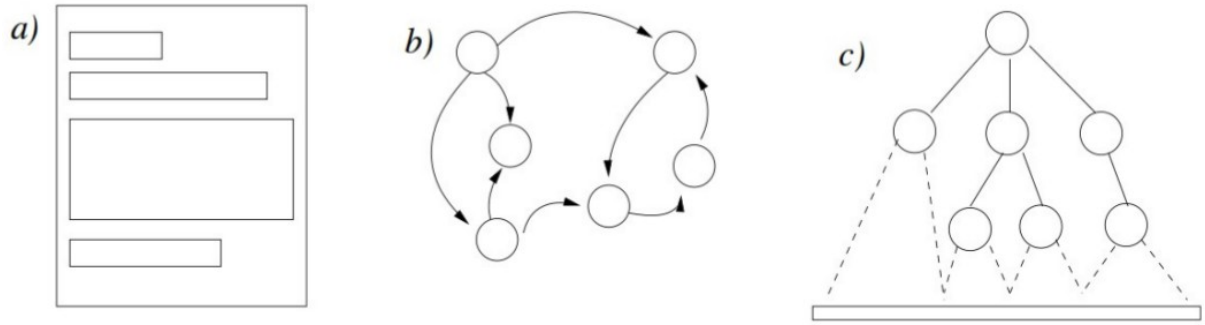


Figura 4: (a) Estructura fija. (b) Hypertexto. (c) Estructura jerárquica.

Se debe tener en claro la diferencia entre la estructura que puede tener un texto y lo que puede ser consultado sobre la estructura. En general, cualquier texto en lenguaje natural puede tener la estructura que uno desee. Sin embargo, diferentes modelos permiten consultas de sólo algunos aspectos de la estructura real. Cuando solo los aspectos que siguen esas restricciones pueden ser consultados, se dice que la estructura es restringida.

5.3.1 Estructura Fija

Los documentos poseen un conjunto fijo de *campos*, cada campo tiene algo de texto adentro y pueden existir campos que no estén presentes en todos los documentos. Un documento no puede tener texto sin clasificar, es decir cada porción del texto debe estar en un campo según corresponda. No se permite que los campos estén anidados o superpuestos.

Claramente, este tipo de estructura es bastante restrictiva. Sin embargo, muchos sistemas comerciales la utilizan.

La actividad de recuperación está restringida a especificar que un patrón dado sea encontrado en el campo.

A continuación se muestra un ejemplo en el cual es razonable aplicar este modelo. Un usuario puede consultar un conjunto de mails enviados a otra persona con la palabra clave '`informe`' en el asunto del correo. Esto resulta sencillo ya que cada mail tiene una estructura fija: destinatarios, un origen, una fecha, un asunto, un cuerpo, etc. Por otro lado, este modelo no es el adecuado para otra situaciones, por ejemplo para representar la estructura jerárquica presente en un documento HTML.

Se pueden dividir los campos de forma tan rígida que su contenido podrá ser interpretado no como simplemente texto sino como fechas, nombres, número, etc. y de esta manera se permiten hacer consultas más específicas (consultar rango de fechas, edades). Está división se asemeja mucho al modelo relacional, donde cada campo corresponde a una columna en la tabla de la base de datos. Pero las bases de datos hacen mejor uso de su conocimiento usando los tipos de datos para así construir índices más eficientes. Hay muchas propuestas que buscan combinar estas dos ideas. Muchas de ellas tienen como fin extender SQL (Structured Query Language) para permitir recuperación full-text, una es SFQL (Strucured Full-text Query Language). En [Baeza y Ribeiro, 1999] el autor menciona y detalla algunas más.

5.3.2 Hypertexto

Un Hypertexto puede verse como un digrafo (grafo dirigido) donde los nodos contienen algo de texto y los links representan las conexiones entre los nodos. La estructura de Hypertexto es la que representa mayor libertad con respecto a la estructuración, es decir se pueden construir estructuras más flexibles. Desde la explosión de la Web, los hypertextos han ganado mucha popularidad.

La manera en que se recupera desde un hypertexto es la siguiente, un usuario debe recorrer los nodos siguiendo los links para buscar la necesidad de información manualmente (tarea de navegación). A diferencia de las otras estructuras, no es posible consultar basándose en la estructura sino que se busca por el contenido en los nodos vecinos del nodo actual.

Existen muchas propuestas que combinan la navegación y la búsqueda. Una muy interesante que lo hace sobre la Web es WebGlimpse[1]. Esta permite la navegación clásica más la habilidad de buscar contenido en los nodos vecinos al nodo actual.

1 http://webglimpse.net/

5.3.3 Estructura Jerárquica

Esta estructura representa una descomposición recursiva del texto, se utiliza en muchas colecciones de texto como libros, articulos, etc. Se ubica entre los tipos de estructura fija y la estructura de hypertexto. Sólo que a diferencia del hypertexto permite aplicar algoritmos más eficientes para resolver consultas. De aquí surge la siguiente regla: *mientras más poderoso el modelo, más dificil de implementarlo eficientemente.*

A continuación se muestra un ejemplo:

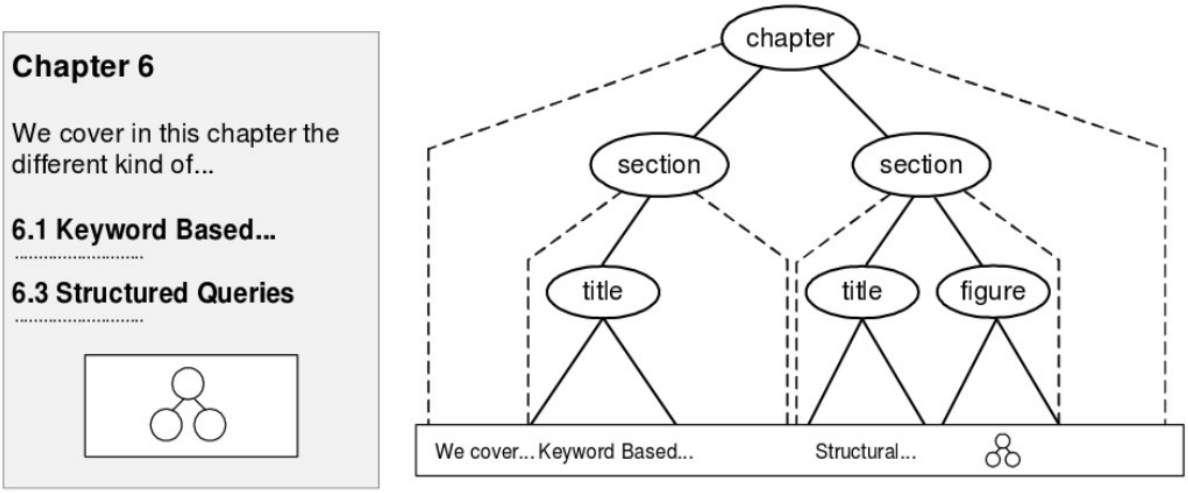
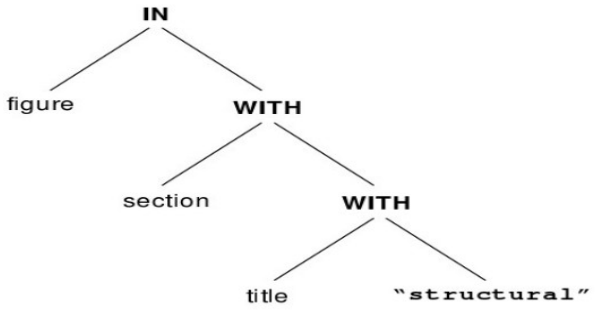
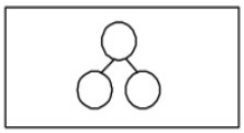


Figura 5: Ejemplo de estructura jerárquica. Fuente: [Baeza y Ribeiro, 1999].

Consideremos la siguiente consulta sobre la estructura:



La cual generará este resultado:



En [Baeza y Ribeiro, 1999] el autor presenta y analiza las disitintas propuestas presentes sobre modelos jerárquicos. Las más significativas son: Expresiones PAT, Listas

superpuestas, Lista de referencias, Nodos proximales y Tree matching.

6 Estrategias de Búsqueda

El objetivo principal en un Sistema de Recuperación de Información es que dada una consulta de un usuario se maximice la cantidad de documentos relevantes para esa consulta y ese usuario. En otras palabras, se trata de reducir la cantidad de documentos recuperados si se recuperan demasiados y se trata de aumentar la cantidad si no se recupera la cantidad de documentos suficientes. Para lograr esto se depende en gran parte de la correcta preparación de una *estrategia de búsqueda*.

Definición: una estrategia de búsqueda es una expresión lógica compuesta por distintos conceptos combinados usando conectores lógicos.

Una estrategia de búsqueda conocida es la *expansión semántica de la consulta*, y se explica en la siguiente sección.

6.1 Expansión semántica de la consulta

La expansión semántica consiste en incorporar a la búsqueda términos que sean conceptualmente equivalentes, como sinónimos, términos relacionados y hasta incluso términos traducidos a otros idiomas. Para la realización de esta estrategia, se utilizan varios recursos lingüísticos. Antes de presentar a los distintos recursos lingüisticos, se muestra una arquitectura de la expansión semántica de una consulta.

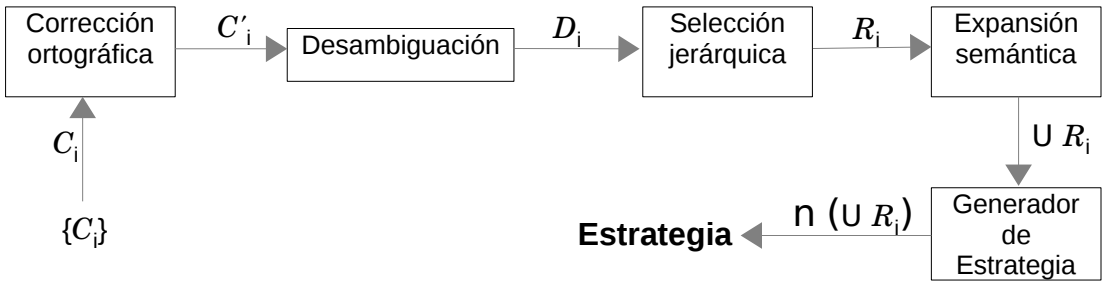


Figura 6: Arquitectura de la expansión semántica de una consulta. Fuente: [Bender y Deco, 2014]

1. El usuario ingresa un conjunto de conceptos $\{C_i\}$. Se supone $\#\{C_i\} = n$, $n \in \mathbb{N}$.
2. Para cada concepto C_i se obtiene el concepto C'_i corregido otrográficamente.
3. Se obtiene el concepto D_i desambiguado.
4. Se obtiene el concepto R_i jerarquicamente relacionado.
5. Se obtiene el resultado de la expansión semántica que es el conjunto union de sinónimos y términos relacionados, claramente no todos los conceptos van a tener el mismo número de sinónimos o términos relacionados.

6. Los conjuntos ingresan al generador, cuya salida es la intersección de esas uniones.

Por ejemplo, un usuario realiza la siguiente consulta: *"relación de la aspirina en el tratamiento del cáncer de pulmón ".*

Los conceptos son: aspirina – tratamiento – cáncer de pulmón.

Y la estrategia provista por el generador es:

(lung neoplasm **OR** lung cancer **OR** cancer de pulmon **OR** carcinoma of the lungs)

AND

(aspirina **OR** aspirin **OR** acido acetil salicilico)

AND

(tratamiento **OR** treatment)

6.1.1 Recursos lingüísticos

Como se mencionó anteriormente, los recursos lingüísticos ayudan en la preparación de estrategias de búsqueda. Estos pueden ser de tipo general o especializado en algún dominio del conocimiento. Se utilizan principalmente para:

- Desambiguar los conceptos.
- Permitir la selección de conceptos jerárquicamente relacionados.
- Expandir semántica y multilingualmente cada concepto.
- Mejorar la recuperación de información.

Además, pueden usarse para:

- Clasificación de la información: Reconocer conceptos similares.
  Por ejemplo, clasificación de las páginas Web a través de un buscador.
- Integración de la información: Permite unificar conceptos expresados con distinta terminología y reconocer coincidencia de autores o instituciones que pueden estar expresadas de distinta manera.
  Por ejemplo, reconocer que dos documentos provienen de la misma institución. "UNR" - "Universidad Nacional de Rosario".

A continuación se detallan algunos de los recursos más utilizados.

6.1.1.1 Diccionarios

Un diccionario indica las distintas acepciones de un término, lo que permite su expansión con:

- *Sinónimos:* relación entre términos con un mismo significado. Por ejemplo: cancer, neoplasma.

- *Merónimos:* relación semántica entre un término que denota una parte y el que denota el correspondiente todo. Por ejemplo, brazo es merónimo de cuerpo.
- *Hipónimos:* relación de subordinación entre términos, es decir un término es un hipónimo de otro si su significado está incluido en el primero. Por ejemplo: gorrion es un hipónimo de pájaro.
- *Hiperónimos:* relación de superordenación entre términos, es decir un término es un hiperónimo de otro si su significado incluye al del segundo. Por ejemplo: animal es un hiperónimo de pájaro.

Si a esto se le suma la ampliación de cada concepto en los idiomas deseados por los usuarios mediante la utilización de diccionarios multilingües, se logra aumentar el número de documentos recuperados.

Agunos de los diccionarios más usados son: WordNet, El diccionario de la Real Academia Española[1], ForeingWord.com[2] (multilingüe), Diccionarios.com[3] (multilingüe), Wordreference.com[4] (multilingüe).

6.1.1.2 Tesauros

Un tesauro es una lista que contiene los términos empleados para representar los conceptos, temas o contenidos de los documentos, con el fin de traducir a un lenguaje más estricto que el lenguaje natural que permita mejorar el canal de acceso y comunicación entre los usuarios y las unidades de Información. En otras palabras, es un vocabulario controlado de términos relacionados semántica y genéricamente.

Un tesauro está estructurado formalmente para hacer explícitas las relaciones entre conceptos. Estas relaciones pueden ser:

- *Jerárquicas:* indican términos más amplios o más específicos de cada concepto.
- *Preferenciales:* indican cual es el término preferido entre un grupo de sinónimos .
- *De afinidad:* términos relacionados conceptualmente, pero que no están ni jerárquica ni preferencialmente relacionados.

Los términos de un tesauro se clasifican en:

- *Descriptores (o términos preferentes):* es una palabra o grupo de palabras escogidas de entre un conjunto de términos equivalentes para representar sin ambigüedad un concepto contenido en un documento o en una petición de búsqueda documental.
- *No descriptores (o términos no preferentes o términos equivalentes):* es un sinónimo

1 www.rae.es
2 www.foreignworld.com
3 www.diccionario.com
4 www.wordreference.com

de un descriptor presente en el tesauro. No puede ser utilizado para indexar documentos ni para formular consultas, pero reenvía al descriptor aceptado.

Algunos de los tesauros más difundidos son: Tesauros del CINDOC[1](Centro de Información y Documentación Científica) y Tesauro de la UNESCO[2].

7 Indexado y Búsqueda

En [Baeza y Ribeiro, 1999] el autor describe que dado un cuerpo de documentos, las tareas de un Sistema de Recuperación de Información (SRI) son la indexación, la búsqueda y la visualización de dichos documentos. A continuación se detalla la tarea de indexación y luego la tarea de búsqueda.

Hasta el momento la eficiencia parece un tema secundario comparado con la efectividad, sin embargo, siempre se debe tener en cuenta a la hora del diseño de un SRI. La eficiencia en los SRI es procesar consultas de usuarios con mínimos requerimientos computacionales.

Cuando se trata de aplicaciones de gran tamaño, la eficiencia se vuelve más y más importante. Por ejemplo, los motores de búsqueda en la Web indexan teras de información y responden a miles de consultas por segundo.

Para llevar a cabo esta tarea, se utiliza una estructura de datos construida a partir del documento, esta estructura permite acelerar el proceso de búsqueda y se la denomina ***índice***.

El método más común de construcción de indices es por palabras, es decir, se representa a los documentos como una bolsa con las palabras que contienen. Otra alternativa es la utilización de frases pero la construcción se vuelve muy complicada y no mejora mucho la eficiencia.

Dentro de este método, la implementación más utilizada son los ***indices invertidos***. La estructura de índice invertido está compuesta por dos elementos:

- El *vocabulario* (conjunto de todas las palabras diferentes en el texto).
- Las *ocurrencias*.

Por cada palabra en el *vocabulario*, el índice almacena los documentos que contienen esa palabra (de aquí, índice invertido).

Para obtener el vocabulario de un conjunto de documentos, se aplican diferentes procesos a los documentos. Algunos de estos son:

- *Tokenization:* extracción de las palabras de los documentos.
- *Stemming:* obtención de la raíz de la palabra, de forma que el proceso de búsqueda

1 http://thes.cindoc.csic.es
2 http://databases.unesco.org/thessp/

se realice sobre las raices y no sobre las palabras originales.

- *Lemmatization:* se utiliza un diccioanario para reemplaza una palabra por su raíz. Comunmente se aplica para llevar los verbos a infinitivo.
- *Remoción de stopwords:* eliminación de palabras que aparecen en muchos documentos.

Las *ocurrencias* mantienen referencia de las posiciones de cada palabra en cada documento que aparezcan. De esta manera, se puede responder a consultas con frases o proximidad.
Claramente, la construcción del índice parece ser una tarea muy costosa. Por lo que surge la siguiente pregunta ¿Qué sucede con el índice cuando un documento se modifica? La respuesta es que se debe actualizar el índice. Por lo tanto, el hecho de que los documentos cambian frecuentemente es un incoveniente. Sin embargo, la mayoría de las colecciones de documentos, incluyendo la Web, son colecciones semi-estáticas (colecciones actualizadas cada intervalos razonables de tiempo, por ejemplo, diariamente).
Para evaluar la eficiencia de un SRI, se emplean las siguientes métricas:

- *Indexing time:* Tiempo necesitado para construir el índice.
- *Indexing space:* Espacio usado durante la construcción del índice.
- *Index storage:* Espacio requerido para almacenar el índice.
- *Query latency:* Intervalo de tiempo entre la llegada de una consulta y la generación de la respuesta.
- *Query throughput:* Número promedio de consultas procesadas por segundo.

El **proceso de búsqueda** en los índices invertidos esta dividido en tres etapas:

1. Se buscan las palabras de la consulta en el *vocabulario*.
   Si la consulta está compuesta por varias palabras, como una frase, se la divide en los términos que la forman y se procede de esta forma separadamente para cada palabra.
2. Se recuperan los documentos en que ocurren dichas palabras.
   - Si la consulta estaba compuesta por una palabra el proceso termina y se muestran los resultados ordenados por un ranking.
   - Si la consulta estaba compuesta por varias palabras, en la tercera etapa se termina de resolver la consulta procesando los datos obtenidos.
3.
   - Si la consulta era una frase, se busca entre los documentos recuperados en el paso dos, aquellos en que las palabras que componen la frase aparezcan en forma consecutiva. Esto se verifica usando la información sobre las posiciones de

las palabras.

- En el caso de que aparezcan operadores booleanos AND, OR, NOT se realizará la intersección, union, complemento de los conjuntos de documentos recuperados, respectivamente.

Entre las ventajas de los índices invertidos se destaca que poseen un formalismo simple y eficiente y cuentan con la capacidad de manejar pesos no binarios y así medir la similitud entre un documento y una consulta de manera gradual. Pero al asumir a los documentos como secuencias de palabras, las consultas complejas (como frases) se vuelven muy costosas de resolver y limitan las búsquedas.

8 Aplicación de Recuperación de Información en Sistemas Recomendadores

Son variados los campos en donde se puede aplicar la Recuperación de Información. Este capítulo se enfoca en la aplicación de la Recuperación de Información en búsquedas inteligentes que utilizan **sistemas recomendadores** para la búsqueda de **objetos de aprendizaje**. Primero se dan los conceptos previos necesarios y luego un breve estado del arte del tema.

Según [Terveen y Hill, 2001] los sistemas recomendadores son capaces de seleccionar, de forma automática y personalizada, el material que mejor se adapte a las preferencias o necesidades de un usuario. Realiza esa tarea comparando el perfil del usuario con algunas características de referencia de los temas (metadatos) y busca predecir el ranking o ponderación que el usuario le daría a un ítem que el sistema no hubiera considerado.

En el dominio de la educación existe gran cantidad y diversidad de material que puede contribuir a los procesos de enseñanza y aprendizaje. Según [Wiley, 2002] un Objeto de Aprendizaje (OA) es cualquier recurso digital que puede ser utilizado repetidamente para facilitar el aprendizaje. Pueden adquirir diferentes formatos y pueden ser reutilizados, actualizados, combinados, separados, referenciados y sistematizados.

Utilizando buscadores se puede obtener todo tipo de material desde la Web pero existen problemas que dificultan la tarea, como la sobrecarga de información disponible, la falta de estructura en la misma y el hecho de que no siempre el resultado es el esperado por el usuario (no se consideran las preferencias personales). Existen formas de solucionar estos inconvenientes:

- Utilización de repositorios (para acceder a información más estrucurada).
- Utilización de sistemas recomendadores para que se tengan en cuenta las preferencias de los usuarios en el momento de la búsqueda.

Un **repositorio de objetos de aprendizaje** es una gran colección de Objetos de Aprendizajes estructurados como una base de datos, con metadatos asociados y que en la mayoría de los casos se puedan acceder vía Web. Los metadatos son un conjunto de atributos necesarios para describir las principales características de un objeto. LOM (Learning Object Metadata) es el estándar de metadatos de la IEEE para los OAs; especifica la sintaxis y la semántica de un conjunto mínimo de metadatos necesarios para identificar, administrar, localizar y evaluar un OA.

Algunos repositorios conocidos son:

- ARIADNE[1] European Association open to the World, for Knowledge Sharing and Reuse.
- OER Commons[2] Open Educational Resources.
- FLOR[3] Federación Latinoamericana de Repositorios.
- MERLOT[4] Multimedia Educational Resource for Learning and Online Teaching.

En la actualidad hay una gran cantidad de enfoques para llevar a cabo la construcción de sistemas recomendadores que asisten la búsqueda personalizada de objetos de aprendizaje (diferentes arquitecturas, etc.).

Por ejemplo, en [Casali et. al., 2012] los autores presentan la arquitectura e implementación de un prototipo de un sistema recomendador para recuperar objetos de aprendizaje que se adecuen con el perfil del usuario. Utilizaron arquitecturas de agentes g-BDI5 [Casali et. al., 2005] para el sistema recomendador. La búsqueda se realiza en repositorios accesibles vía Web y con metadatos. Además, hicieron una experimentación del prototipo en la cual usaron el repositotio Ariadne, obteniendo resultados promisorios respecto a los rankings de OAs recomendados por el sistema.

En [Cazella et. al., 2010] se presenta un sistema recomendador de objetos de aprendizaje que utiliza un mecanismo de filtrado colaborativo basado en competencias. El modelo permite a los estudiantes recibir recomendaciones de objetos de aprendizaje de forma automática, de acuerdo con los intereses de los estudiantes, y atendiendo las competencias que deseen desarrollar. El prototipo implementado fue capaz de recomendar contenidos relevantes para los estudiantes logrando un buen nivel de precisión para las sugerencias hechas.

---

1 http://www.ariadne-eu.org/
2 http://www.oercommons.org
3 http://ariadne.cti.espol.edu.ec/FederatedClient
4 www.merlot.org
5 Gradual Belief-Desire-Intention.

Otros enfoques, tales como [Chen y Duh, 2008] han aplicado la lógica difusa para recomendar cursos con diferentes grados de dificultad para alumnos de acuerdo a respuestas inciertas/difusas obtenidas mediante retroalimentación. En [Hsieh et. al., 2013] el autor utiliza también lógica difusa para la construcción de un camino de aprendizaje adecuado en función de las ideas erróneas de los alumnos para recomendar materiales más adecuados.

Conclusiones

En esta monografía se han presentado algunos de los conceptos más importantes en el área de la Recuperacion de Información como las diferencias que posee con otras aplicaciones en lo relacionado con la recuperación de datos, los distintos modelos sobre los que se basan los sistemas que permiten la recuperación de la información, las tareas de indexado y búsqueda, los distintos tipos de lenguajes de consultas y como ha evolucionado mucho más alla de sus primeros objetivos como indexar y buscar documentos útiles en una colección de documentos. También, debido a la necesidad de evaluar el desempeño de un sistema de recuperación de información se mostraron algunas medidas que permiten cuantificar su efectividad.

Además se da un estado del arte sobre la aplicación de la recuperación de información en búsquedas inteligentes que utilizan sistemas recomendadores para la búsqueda de objetos de aprendizaje. Como consecuencia de esa sección, se ha percibido que el campo se está moviendo y están surgiendo nuevos enfoques de investigación en lo que tiene que ver con el aporte de la recuperación de información en los sistemas recomendadores de varios dominios y no sólo en la recuperación de objetos de aprendizaje.

Referencias y Bibliografía

[Baeza y Ribeiro, 1999] Baeza-Yates,R., Ribeiro-Neto, B. (eds.), Modern Information Retrieval. New York. ACM Press, 1999.

[Bender y Deco, 2014] Bender, C.M., Deco C. Tópicos avanzados de Bases de datos, 2014.

[Casali et. al., 2005] A. Casali, L. Godo, C. Sierra. “Graded BDI Models For Agent Architectures”, in CLIMA V, edited by J. Leite and P. Torroni LNAI 3487, Springer-Verlag, Berling Heidelberg, 126-143, 2005.

[Casali et. al., 2012] Casali, A., Gerling, V., Deco, C., Bender, C.: A Recommender System for Learning Objects Personalized Retrieval. (eds) Educational Recommender Systems and Technologies: Practices and Challenges, pp. 182-210. (2012).

[Cazella et. al., 2010] Cazella, S.C., Reategui, E.B., Behar, P.A.: Recommendation of Learning Objects Applying Collaborative Filtering and Competencies. Key Competencies in the Knowledge Society pp. 35-43 (2010).

[Chen y Duh, 2008] Chen, C.M., Duh, L.-J.: Personalized web-based tutoring system based on fuzzy item response theory, Expert Systems with Applications, Volume 34, Issue 4, May 2008, pp. 2298-2315, ISSN 0957-4174, (2008).

[Croft, 1987] Croft, W. B. 'Approaches to intelligent information retrieval'. Information Proccesing & Managment, 23, 4, 1987.

[Grossman y Frieder, 1998] Grossman, D.A., Frieder, O. Information retrieval: algorithms and heuristics. Boston: Kluwer Academia Publishers, 1998.

[Hsieh et. al., 2013] Hsieh, T.-C., Lee, M.-C., Su, C.-Y.: Designing and implementing a personalized remedial learning system for enhancing the programming learning. Educational Technology & Society 16(4): 32-46 (2013).

[Losse,1998] Losee, R., Text Retrieval and Filtering: Analytic Models or Performance, Kluwer, Boston, 1998.

[Salton y Mc Gill, 1983] Salton, G., Mc Gill, M.J. Introduction to modern Information Retrieval. New York: Mc Graw-Hill Computer Series, 1983.

[Terveen y Hill, 2001] Terveen L. G. and Hill W., Beyond Recommender Systems: Helping People Help Each Other. In Carroll, J. (Ed.), HCI in the New Millenium. Addison Wesley, 2001.

[van Rijsbergen, 1979] Van Rijsbergen, C. J. Information Retrieval. Butterworths, 1979.

[Wiley, 2002] Wiley, D. Connecting Learning Objects to Instructional Design Theory: A definition, a metaphor, and a taxonomy. In D. A. Wiley (ed.) Instructional Use of Learning Objects. Editorial Association for Instructional Technology, 2002.