Capítulo 6

Retorno de la información

Parte 1

Retorno de la información

- El **retorno de la información** es el proceso de retornar documentos a partir de una colección de documentos en respuesta a una consulta.
  - Los documentos suelen estar en lenguaje natural no estructurado.
- La **información no estructurada:**
  - No tiene un modelo formal bien definido
  - Se basa en la comprensión del lenguaje natural
  - Se almacena en una variedad amplia de formatos estándares
- ¿Cuáles son las diferencias entre BD relacionales y retorno de la información?

Retorno de la información

Bases de Datos Relacionales	Sistemas de retorno de la información
Datos estructurados	Datos no estructurados
Dirigidos por esquemas relacionales	No hay esquemas fijos
Modelo de consultas estructurado	Modelos de consulta libres de forma
Operaciones sobre metadatos	Operaciones sobre datos
Las consultas retornan datos	Las búsquedas retornan lista de punteros a documentos.
Los resultados se basan en correspondencia exacta y son siempre correctos.	Los resultados se basan en correspondencia aproximada y medidas de efectividad.
Trabajan con transacciones	No trabajan con transacciones

Lenguajes de consulta

- Los sistemas de retorno de información (SRI) típicamente permiten **expresiones de consulta** formadas usando palabras clave y conectivos proposicionales.
- Ahora vemos algunos **ejemplos de lenguajes de consulta**:
  - **Consultas usando frases**: una frase es una secuencia de palabras
  - **Consultas de palabras clave**: se escribe texto de palabras clave y se asume and entre esas palabras.
  - **Consultas Booleanas**: las expresiones involucran términos y conectivos booleanos.
  - **Consultas basadas en expresiones regulares**: se usan expresiones regulares y búsqueda basada en correspondencia de patrones.

Lenguajes de consulta

- **Ejemplos de lenguajes de consulta (continuación):**
  - **Consultas de proximidad**: se expresa cuan cerca deben estar entre sí ciertos términos. En algunos casos se pide respetar el orden de las palabras. Se usan sentencias como: near, adjacent, after, within. También se puede especificar la máxima distancia entre palabras específicas.
    - **Ejemplo**:“ciencia NEAR tecnología”, encuentra documentos donde ambos términos aparecen próximos entre sí.
    - **Ejemplo**: queremos que inteligencia y artificial aparezcan en la misma oración. "inteligencia WITHIN/10 artificial“ busca ambos términos con una separación máxima de 10 palabras.
  - **Consulta en lenguaje natural**: La consulta es texto en lenguaje natural.
    - Requiere entender la estructura y el significado de la consulta.
    - La consulta puede ser una pregunta o una narrativa.

Ejemplo con el buscador de Google

- Google utiliza un lenguaje de consulta específico llamado **Google Search**; el cual está diseñado para entender consultas en lenguaje natural.
  - Puedes escribir tus preguntas o términos de búsqueda como hablarías normalmente.
- **Google Search usa operadores:**
  - Para buscar una frase exacta ponerla entre comillas (“ ”).
  - Usar OR para buscar por un término u otro. Ejemplo: cats OR dogs.
  - Para excluir términos de una búsqueda usar el signo ‘-’. Ejemplo: jaguar – car. Retorna resultados acerca de jaguares que no son autos.
  - Para restringir resultados a un sitio específico usar site::. Ejemplo: site::wikipedia.org muestra resultados solo de Wikipedia.
  - Para buscar por paginas con una palabra específica en el título usar intitle:: . Ejemplo: intitle::apple retorna páginas con apple en el título.

Resultados de una consulta

- Los **resultados de una búsqueda** pueden ser una lista de identificadores de documentos y también algunas piezas de texto.
- Los documentos suelen retornarse en orden decreciente de puntaje de relevancia.
  - Usualmente se retornan unos pocos documentos del resultado y no todos.
- Es necesario definir cuándo un documento es más relevante que otro.

Resultados de una consulta

- La **relevancia** se basa en **factores** como:
  - **Frecuencia de términos**: frecuencia de ocurrencia de término de una consulta en un documento.
  - **Frecuencia inversa de documentos**: ¿en cuántos documentos ocurre la palabra? Si ocurre en menos documentos se le da más importancia a la palabra.
  - **Enlaces a documentos**: Si hay más enlaces a un documento, el documento es más importante.

Resultados de una consulta

- **La mayoría de los sistemas de retorno de la información (SRI) agregan:**
  - Las palabras que ocurren en el título, lista de autores, títulos se les da más importancia.
  - Las palabras cuya primera ocurrencia es tarde en el documento se les da poca importancia.
  - **Proximidad:** si las palabras de una consulta aparecen cerca entre sí en el documento, el documento tiene más importancia que si las palabras ocurren bien lejos unas de otras.

Enfoques estadísticos

- Un SRI al procesar una consulta no accede directamente a los documentos, sino que se usa una representación de cada documento.
  - Se construye para cada documento una **estructura que resume lo que contiene** y es relevante para procesamiento de consultas.
- En un **enfoque estadístico** los documentos son analizados y descompuestos en **piezas** que pueden ser palabras, frases, n-gramas (i.e. n palabras consecutivas).
  - Cada **pieza** se la cuenta, pesa y mide para determinar su relevancia e importancia.
  - Dada una consulta se comparan los términos de la consulta con las piezas para determinar un **grado potencial de correspondencia** y para producir una **lista ordenada de documentos resultantes**.

Enfoques estadísticos

- **Ejemplos de enfoques estadísticos** son: Booleanos, Espacio vectorial y probabilístico.
- **Modelo Booleano**:
  - Los documentos se representan como un conjunto de términos.
  - Se usan consultas booleanas.
  - Los documentos retornados son una correspondencia exacta.
    - O el documento sirve o no sirve para la consulta.
  - No hay noción de ordenar los documentos por relevancia.

Enfoques estadísticos

- **Modelo de espacio vectorial:**
  - Cada documento se representa con un **vector de valores** de dimensión n.
  - Cada término es una **dimensión**.
  - Para cada dimensión del vector hay un número que puede representar; un valor booleano (indicando la presencia del término en el documento), o la frecuencia del término en el documento, o el peso (p.ej. usando TF-IDF).
  - Se usa una **función de similitud de vectores**.
  - Se puede establecer la **medida de relevancia** de un documento con respecto a una consulta.
  - Una consulta también se representa como un vector de términos.
  - El **vector de una consulta** es comparado con los vectores de los documentos para **estimación de similitud/relevancia**.
  - Las respuestas a una consulta se ordenan por relevancia.

Modelo de espacio vectorial

- Los términos tienen peso en los vectores.
- Una opción es que el peso sea la **frecuencia** de cada término en el documento/consulta.
- Otra posibilidad es usar **TF-IDF** (frecuencia de término-frecuencia inversa de documento).
- TF-IDF es usado para evaluar la importancia de una palabra en una colección de documentos.
- **Idea de TF-IDF**: términos que capturan la esencia de un documento aparecen frecuentemente en el mismo; pero para que un término sea bueno para discriminar un documento de los demás, entonces debe ocurrir en unos pocos documentos de la colección.

Modelo de espacio vectorial

- Sea término i y documento Dj
- $TF\text{-}IDF_{i,j} = Tf_{i,j} * IDF_i$

$$TF_{ij} = f_{ij} \Big/ \sum_{i=1 \text{ to } |V|} f_{ij}$$

$$IDF_i = \log\left(N / n_i\right)$$

  - $Tf_{i,j}$ es la frecuencia del término i en el documento Dj normalizada.
  - f_{ij} es la cantidad de ocurrencias del término i en el documento Dj.
  - N es el número de documentos en la colección.
  - n_i es el número de documentos donde el término i ocurre.

Modelo de espacio vectorial

- Se puede calcular la relevancia de un documento D_j con respecto a una consulta Q de la siguiente manera:

$$\mathrm{rel}(D_j, Q) = \sum_{i \in Q} TF_{ij} \times IDF_i$$

- **Observación**: si la consulta dice que el término i no debe considerarse, entonces ese término no debería estar en la fórmula.

Modelo de espacio vectorial

- Se usa **función de similitud de vectores.**
- La función del coseno del ángulo entre los vectores de la consulta y el documento se usa frecuentemente para estimar la similitud.

$$\text{cosine}(d_j, q) = \frac{\langle d_j \times q \rangle}{\| d_j \| \times \| q \|} = \frac{\sum_{i=1}^{|V|} w_{ij} \times w_{iq}}{\sqrt{\sum_{i=1}^{|V|} w_{ij}^2} \times \sqrt{\sum_{i=1}^{|V|} w_{iq}^2}}$$

- Donde d_j es el vector del documento j, q es el vector de la consulta.
- $W_{i,j}$ es el peso del término i en el documento j, $w_{i,q}$ es el peso del término i en el vector de la consulta q.
- |V| es el número de dimensiones en el vector.

Modelo de espacio vectorial

- **Interpretación del resultado de la función de similitud de vectores:**
  - Si el resultado es cercano a 1, significa que los vectores son muy similares.
  - Si el resultado es cercano a 0, los vectores no tienen casi nada en común.
  - Si el resultado es -1, entonces los vectores son opuestos.
- La función de similitud de vectores se puede usar para calcular la similitud entre el vector de una consulta y el vector de un documento.
  - Este valor puede usarse también como medida de relevancia de un documento con respecto a una consulta.

Selección de términos relevantes de un documento

- Antes de construir la representación de un documento es importante encontrar los términos relevantes.
- Hace falta preprocesar los documentos de la colección para encontrar los términos relevantes.
- No todos los términos son relevantes:
- **Stopwords** son palabras que se espera que ocurran en el 80% o más de los documentos de la colección.
  - Ejemplos: él, la, lo, de, un, y, para, etc.
  - No contribuyen a la relevancia.
  - Se deben ignorar las stopword.
  - En las consultas se deben remover las stopword.

Selección de términos relevantes de un documento

- A veces un término aparece de muchas maneras y no vale la pena tener todas sus variantes.
  - **Por ejemplo**: un mismo verbo conjugado de muchas maneras
  - **Por ejemplo**: usar 'comput' para *computer*, *computing*, *computable*, and *computation*.
- El **algoritmo de Martin Porter** hace este tipo de trabajo.
  - Básicamente se toma una palabra y reduce su forma a su raíz o base común. Por ejemplo: convierte corriendo, corre y correr a la raíz en común 'corr'.
  - Entonces agrupa palabras similares bajo una misma raíz mejorando la eficiencia de la recuperación de la información.

Selección de términos de un documento

- A veces aparecen **sinónimos de un término.**
  - Se puede usar un término por concepto en lugar de todos los sinónimos.
  - Esto no es tan simple porque una palabra puede tener diferentes significados en diferentes contextos.
    - E.g., document: "motorcycle repair", query: "motorcycle maintenance"
      - Need to realize that "maintenance" and "repair" are synonyms
    - System can extend query as "motorcycle *and* (repair *or* maintenance)"

Selección de términos de un documento

- **Solución: WordNet** agrupa palabras en conjuntos de sinónimos llamados synsets.
  - Los synsets se dividen en **categorías**: sustantivo, verbo, adjetivo y adverbio.
  - Dentro de categoría los synsets se organizan usando **relaciones** clase/subclase (o ES).
- **El uso de sinónimos en SRI tiene varias ventajas significativas:**
  - **Mejora de la precisión**: Los SRI pueden entender mejor la intención del usuario y ofrecer resultados más relevantes, incluso si las palabras exactas no coinciden.
  - **Aumento de cobertura**: permite encontrar información que usa diferentes términos para referirse a lo mismo, asegurando que no se descarten documentos importantes.
  - **Experiencia del usuario optimizada**: los usuarios no tienen que adivinar el término exacto que se usó en los documentos, lo que facilita y agiliza la búsqueda de información.

Selección de términos de un documento

- A veces interesa recolectar **entidades** en lugar de términos.
  - P.ej: hechos, eventos, lugares, relaciones, nombres de personas, etc.
  - Se extrae contenido estructurado a partir de texto.
  - Para esto se puede usar un **enfoques basados en reglas** con expresiones regulares o gramática lingüística específica del idioma;
    - además, para hacerlo más poderoso se pueden usar sinónimos.
  - Se usan técnicas de análisis sintáctico o correspondencia de patrones.

Selección de términos de un documento

- **El retorno de entidades es muy útil para:**
  - **Mejorar la precisión en las búsquedas:** Usando esto, un SRI puede entender mejor las búsquedas del usuario y entregar resultados más relevantes.
  - **Agrupar información relacionada**: facilita la identificación de documentos que mencionan las mismas entidades, aunque usen términos diferentes.

Índices invertidos

- **Problema**: ¿Cómo buscar las ocurrencias de un término en los documentos de una colección?
- **Solución 1**: cuando la colección es pequeña se puede escanear secuencialmente cada documento.
- Si las colecciones de documentos son grandes hay una solución mejor.
- **Solución 2**: Se usa una estructura de datos de **índice invertido**.
  - Esta estructura puede tener para cada término los identificadores de los documentos donde aparece el término y
  - también puede tener las posiciones en el documento donde aparece el término (para relevancia basada en proximidad de términos).

Índices invertidos

- **Implementación de índices invertidos**:
  - La lista invertida de un término puede requerir varios bloques de disco.
  - Para eficiencia en el acceso, se puede tener una lista invertida de un término en un conjunto consecutivo de bloques.
  - Se puede usar un índice de árbol B+ para mapear cada término a su lista invertida asociada.

Índices invertidos

Document 1

This example shows an example of an inverted index.

Document 2

Inverted index is a data structure for associating terms to documents.

Document 3

Stock market index is used for capturing the sentiments of the financial market.

ID	Term	Document: position
1.	example	1:2, 1:5
2.	inverted	1:8, 2:1
3.	index	1:9, 2:2, 3:3
4.	market	3:2, 3:13

Índices invertidos

- **Construcción de índices invertidos:**
  1. Extraer el vocabulario (términos) de los documentos de la colección
  2. Armar estadísticas para cada documento dependiendo del modelo usado.
  3. Invertir el stream de documentos con sus términos en un stream de términos y sus documentos.
     - Aquí se puede agregar información adicional como frecuencias de términos, posiciones de términos y pesos de términos.

Índices invertidos

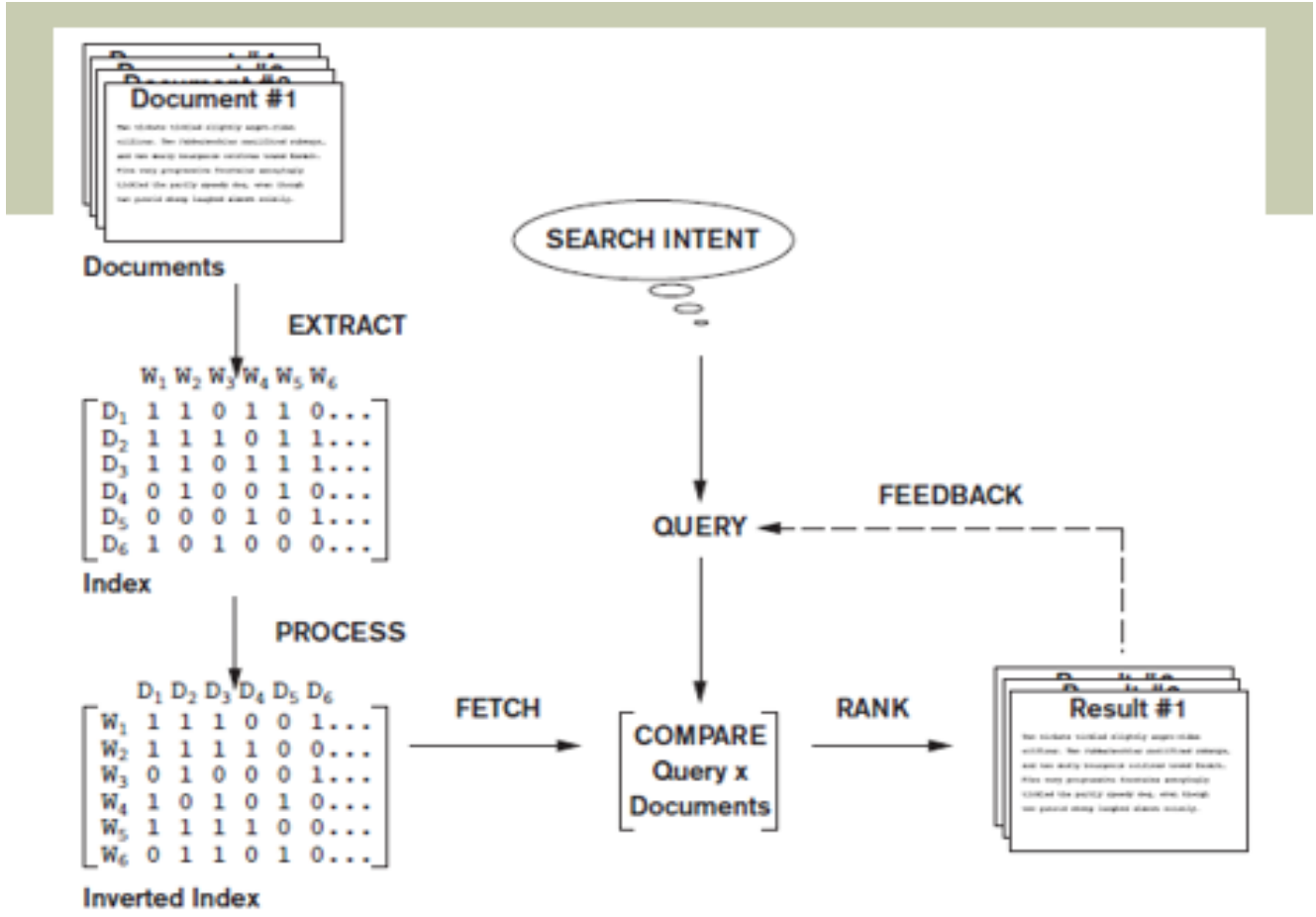


Figure 27.2 Simplified IR process pipeline

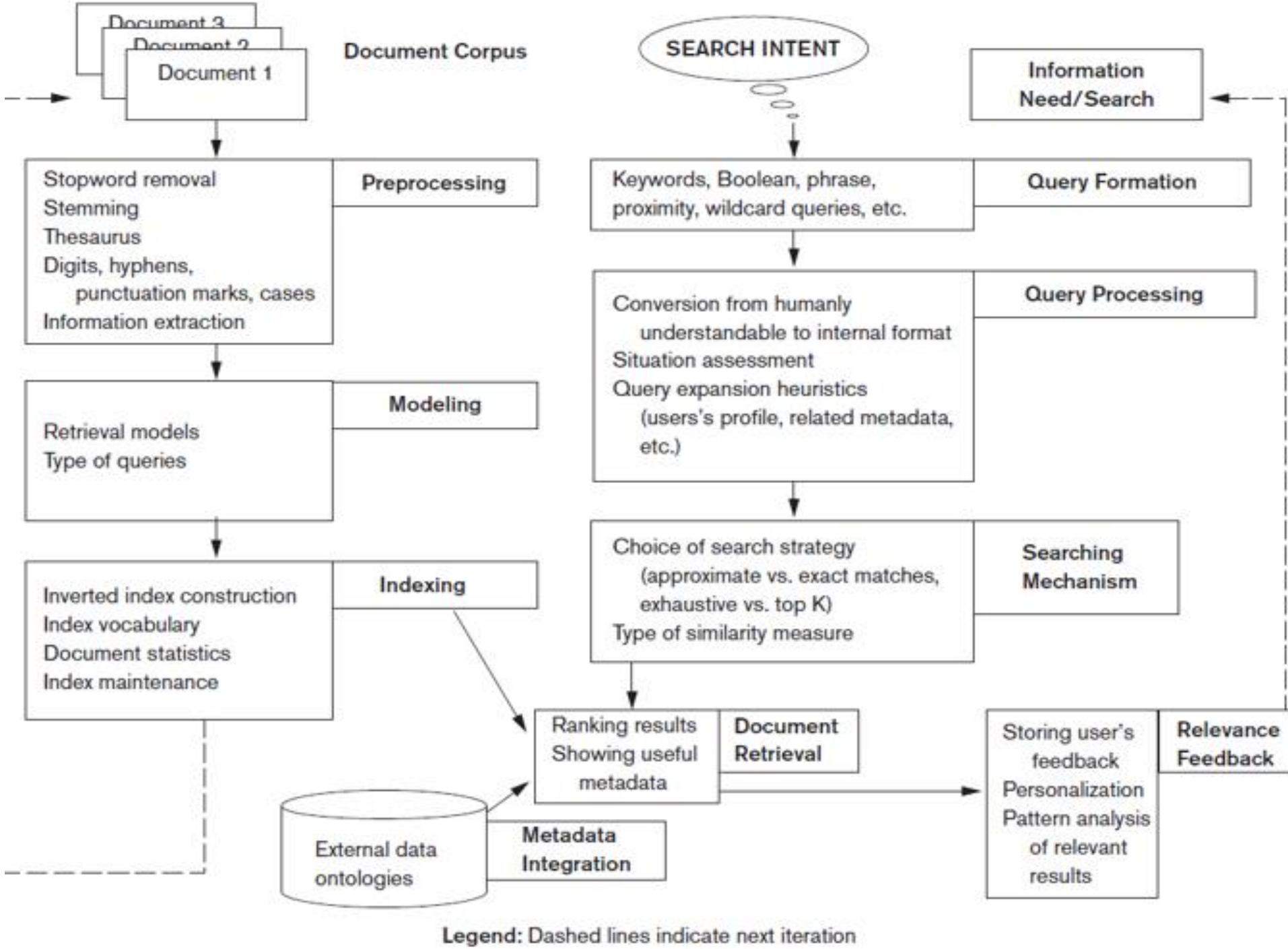
Búsqueda de documentos relevantes a partir de consulta usando índice invertido

- Se trata de un proceso de 3 pasos:
  1. **Búsqueda de vocabulario:** Cada término de la consulta se busca en el vocabulario. Los términos se pueden ordenar lexicográficamente para mejorar eficiencia.
  2. **Retorno de la información de documentos:** se retorna la información del documento para cada término.
  3. **Manipulación de la información retornada:** la información de cada documento es procesada para incorporar las distintas formas de lógica de consulta.
     - Ilustramos este paso para consultas booleanas.

Búsqueda de documentos relevantes a partir de consulta usando índice invertido

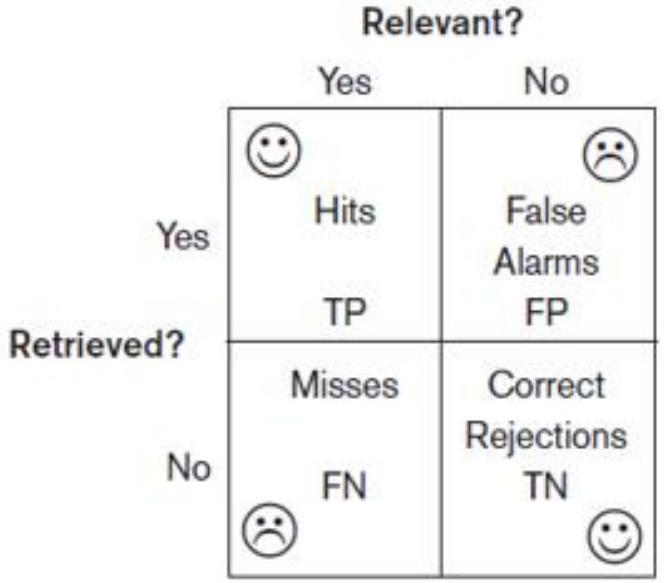
- **Tratamiento de consultas booleanas:**
  - **Suposiciones:** La consulta involucra n términos. S_i es conjunto de identificadores de documentos donde aparece término i ($i \in \{1,\ldots,n\}$).
  - **Operación and**: Los documentos deseados son: $S_1 \cap S_2 \cap \ldots \cap S_n$.
  - **Operación or**: Los documentos deseados son: $S_1 \cup S_2 \cup \ldots \cup S_n$.
  - **Operación not**: Sea t_i término i. Explicamos not t_i. Sea S el conjunto de todos los identificadores de documentos. Podemos eliminar los documentos que contienen término i haciendo: $S - S_i$.
- **Otra forma de procesar operación and**: se retornan los documentos conteniendo al menos uno de los términos de la consulta, pero ordenados por su medida de relevancia.

Framework para retorno de la información



Medición de la relevancia de los resultados de una consulta

- Los SRI soportan solo retorno aproximado.
- Se pueden dar las siguientes situaciones:
  - **Falsos negativos**: algunos documentos relevantes pueden no ser retornados.
  - **Falsos positivos**: algunos documentos irrelevantes pueden ser retornados.
  - Los documentos relevantes retornados se llaman **verdaderos positivos** y los resultados irrelevantes no retornados se llaman **verdaderos negativos**.



Medición de relevancia de los resultados de una consulta

- Métricas de desempeño relevantes:
  - **Precisión**: ¿qué porcentaje de los documentos retornados son relevantes para consulta?
    - Número de documentos relevantes retornados por la consulta divididos por el número total de documentos retornados por la consulta.
  - **Cobertura**: ¿qué porcentaje de los documentos relevantes para la consulta son retornados?
    - Número de documentos relevantes retornados por la consulta divididos por el número total de documentos relevantes en la base de datos.
- La cobertura puede ser incrementada al presentar más resultados al usuario; pero existe el riesgo de que disminuya la precisión.

Medición de relevancia de los resultados de una consulta

- **Problema**: ¿Cómo definir qué documentos son realmente relevantes y cuáles no?
  - ¿Se podrá automatizar esto?
  - Esto requeriría comprender el lenguaje natural y comprender el propósito de una consulta.
- Al final se termina creando consultas y etiquetando manualmente documentos como relevantes e irrelevantes.

Medición de relevancia de los resultados de una consulta

- **Problema**: Alta precisión es lograda casi siempre a expensas de cobertura y recíprocamente.
- La medición **F-score** es usada como una medida que combina precisión y cobertura para comparar distintos conjuntos de resultados (es el promedio armónico de los dos números).
- $$F = \frac{2pr}{p+r}$$ equivalentemente: $$F = \frac{2}{\frac{1}{p} + \frac{1}{r}}$$
- F tiende a ser cercano al más pequeño de p y r. Para F alto tanto r como p deben ser altos.

Lucene

- **Lucene** es una máquina de búsqueda e indexado popular en la industria y la academia.
- Los documentos no estructurados pasan por un proceso de indexado antes de estar disponibles para consultas. Puede manejar colecciones grandes.
- Un documento de Lucene se forma de **campos**; los campos tienen un **tipo** que puede ser: binario, numérico o texto.
  - Un campo de texto puede ser texto no tokenizado o un stream de símbolos (tokens).
- Lucene posee una **API de consultas.**
  - Las consultas retornan una lista ordenada de documentos por rango, usando variante de TF-IDF para dar valor a documento de resultado de una consulta.
  - La API de consultas es configurable: se pueden crear consultas para búsquedas por expresiones booleanas, expresiones regulares o por proximidad.
- Lucene usa modelo de espacio vectorial