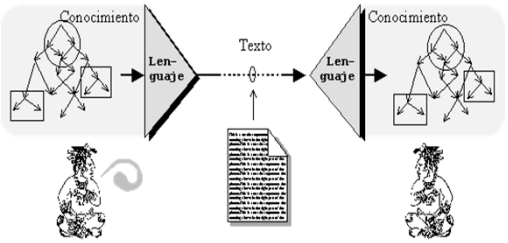
Almacenamiento y recuperación de información

El Lenguaje como Codificador-Descodificador



Introducción I

- **Procesador Lingüístico**: traduce del Lenguaje Natural (LN) a una representación formal equivalente.
- **Sistemas Expertos, Programas de Razonamiento […]:** realizan operaciones lógicas sobre esa representación.

Procesador Lingüístico

- Traduce la información entre la representación textual y la representación formal equivalente
- **Estructura** (refleja la del lenguaje):
  - Módulo fonético y fonológico
  - Módulo morfológico
  - Módulo sintáctico
  - Módulo semántico y pragmático

Introducción II

- Existen volúmenes inmensos de información en LN
- Se realizan operaciones sobre la información tales como búsqueda, comparación, traducción, …
- Los computadores son más capaces de procesar la información que las personas, pero, *¿son capaces de entenderla?*

Módulo Morfológico I

- Diccionarios: lista de palabras de una lengua, junto con diversas informaciones: morfología, definición, etimología, estadísticas, …
- **Lexicón**: forma típica de la entrada de los diccionarios que contiene información fonológica, morfológica, sintáctica y semántica
- Formalismo de representación para codificar los datos
- **Ejemplos**: los bilingües o multilingües recogen la correspondencia entre destintas lenguas

Módulo Morfológico II

- Diccionarios electrónicos:
  - Elementales: Léxico desplegado (inmanejable)
  - Lengua de expresión compleja: el lexicón proporciona la raíz y la información gramatical asociada, y un componente morfológico genera las posibles formas (ayuda a inferir funciones sintácticas)

Problemas Generales

- **Ambigüedad:** Léxica, sintáctica, …
- **Conocimiento lingüístico**: conocimiento léxico y conocimiento general
- **Conocimiento extralingüístico**: información obvia omitida
  - Diccionarios de relaciones entre objetos y de escenarios de las relaciones típicas
  - Métodos de aprendizaje semiautomático

Módulo Sintáctico I

- Las estructuras sintácticas se construyen con una gramática, una especificación mediante reglas de reescritura de las estructuras permitidas en el lenguaje.
- El tipo más común de gramáticas son las de contexto libre (CFGs)
- **CFG**: es una cuádrupla (N,T,R,S)
  - N = conjunto de símbolos No-Terminales
  - T =conjunto de símbolos Terminales
  - R= conjunto de Reglas de la forma $\alpha \rightarrow \beta$, $\alpha \in N$, $\beta \in (N \cup T)$
  - S = axioma (No-Terminal)

Almacenamiento y Recuperación de Información

- **Baeza – Yates [1999]:** Parte de la informática que estudia la recuperación de la información (no datos) de una colección de documentos escritos. Los documentos recuperados pueden satisfacer una necesidad de información de un usuario expresada normalmente en lenguaje natural.

Módulo Semántico y Pragmático II

- La *semántica* estudia el significado del texto y desarrolla los métodos para formar este significado a través de una serie de representaciones sintácticas de las oraciones.
- La *pragmática* estudia cómo las intenciones del autor del texto están expresadas en el texto, es decir, en un contexto dado

Sinónimos

- Sistema de almacenamiento y recuperación de la información (SARI).
- Sistema de recuperación de información.
- Sistema de recuperación de textos.
- Base de datos documental.
- Base de datos de información no estructurada

Diferencias SARI vs SGBD

- Un SGBD tiene un lenguaje de descripción de datos no ambiguo. Se suele expresar que la información de un SGBD es información estructurada mientras que en un SARI es información no estructurada.
- Un SGBD tiene un lenguaje de consulta formal no ambiguo.
- Hay más información no estructurada (libros, artículos, revistas, notas, e-mails, etc.) que información estructurada.

Evaluación de la recuperación

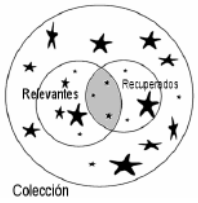
- Eficiencia: Grado de rendimiento de un sistema referido al uso de sus recursos computacionales.
- Eficacia: Grado de rendimiento de un sistema referido a la consecución del objetivo de recuperación medido por el usuario.

Conceptos básicos

- Componentes básicos de un SARI:
  - Almacenamiento de información.
  - Caracterización de las preguntas.
  - Identificación de documentos relevantes a las preguntas.

Tabla de contingencia

	Recuperados	No Recuperados	
Relevante	RvRc	RvNRc	Rv = RvRc + RvNRc
No Relevante	NRvRc	NRvNRc	
	Rc = RvRc+NRvRc		N



12

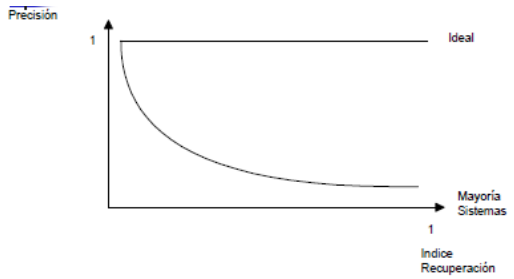
Definiciones

- **Documento**: Ítem de almacenamiento y recuperación.
- **Documento estructurado**: Documento con una división interna describiendo elementos externos al contenido del documento (fecha de creación, lugar, ...) o describiendo la jerarquía de contenidos (capítulos, secciones, subsecciones,...).
- **Documento de texto completo**: Documento almacenado en un SARI con el contenido completo sin resumir.
- **Palabra clave**: Una palabra elegida por el autor, editor o automáticamente, para representar el contenido del documento. Sinónimo: término.
- **Relevante**: Importante, significativo. Sobresaliente, excelente.

Métricas simples

- Precisión: Proporción entre documentos recuperados relevantes y documentos recuperados. P = RvRc/Rc
- Índice de recuperación (recall): Proporción entre documentos recuperados relevantes y documentos relevantes en la colección. R = RvRc/Rv
- Índice de relevancia (generality): Proporción entre documentos relevantes y el tamaño de la colección. G=Rv/N
- Índice de fallos (fallout): Proporción entre documentos no relevantes recuperados y documentos no relevantes en la colección. F = NRvRc/N-Rv

Métricas Simples



Representación

- Los documentos se representan por un conjunto de términos de indexación
- El modelo de representación de los documentos (D)
- Método de representación de las preguntas (P)
- Una función S: D × P → R
  - Para cada par (documento,pregunta) asigna un valor real de similitud

Perspectiva histórica

- **Sistemas pre-informáticos:** Creación manual de índices para clasificar el contenido de libros (Bibliotecas).
- **1ª generación:** Mecanización de las fichas bibliográficas.
- **2ª generación:** Búsquedas más sofisticadas por palabras claves, etc...
- **3ª generación:** Interfaces gráficos, hipertexto, sistemas distribuidos, almacenamiento de documentos de texto completo.
- **Futuro:** Bibliotecas digitales
  - Ventajas: Bajo coste, acceso generalizado, libertad de publicación.
  - Problemas a resolver: Protección de copyright, pago por acceso, interoperabilidad entre bibliotecas digitales.

Archivos

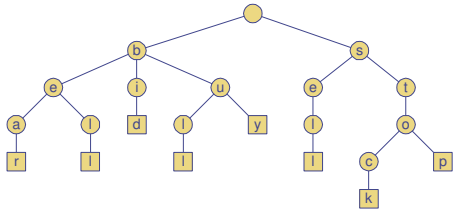
- Tenemos un archivo con todos los documentos
- Una solución es recorrer el archivo buscando palabras (es Ineficiente).
- Otra es almacenar las palabras en alguna estructura para indexación de palabras

Modelos de Recuperación de la Información

- 3 Modelos clásicos
  - Booleano
  - Vectorial
  - Probabilístico
- Normalmente se basan en términos para indexar y también para recuperar información

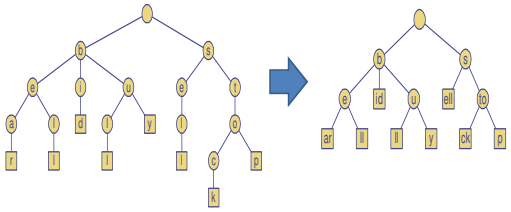
Archivo

- Usar arboles digitales Tries
  - Es un estructura de datos compacta para almacenar conjunto de palabras
  - S = { bear, bell, bid, bull, buy, sell, stock, stop }



Archivo

- Se pueden comprimir para evitar que un nodo solo tenga un hijo



Archivos: Archivo Invertido

- El tamaño de los registros aumenta con el número de documentos
- La matriz está llena de 0's
- Solución: Partir el archivo en 2:
  - Diccionario
  - Archivo de almacenamiento de listas (Postings file)

Archivos

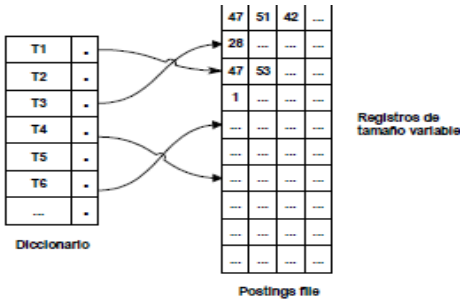
- Otra forma es tener un registro para cada documento con un 0 ó 1 por cada término

	T1	T2	T3	T4	T5
D1	0	0	0	1	0
D2	0	1	0	1	0
D3	1	0	1	1	0
D4	0	1	1	0	1

Archivo Directo

- Para saber si un documento tiene un término miramos en la tabla

Archivos: Archivo Invertido



Archivos

	D1	D2	D3	D4	D5
T1	0	0	0	1	0
T2	0	1	0	1	0
T3	1	0	1	1	0
T4	0	1	1	0	1

Archivo Invertido

- Es más eficiente pues al buscar un término sólo miramos un registro

Modelo Booleano

- Modelo clásico basado en la teoría de conjuntos y el álgebra de Boole.
- Es el modelo más simple.
- Los documentos se representan por conjuntos de términos contenidos en ellos.
- Las consultas se expresan como expresiones booleanas con una semántica clara y concreta.
- Adoptado por muchos de los SRI tempranos.

30

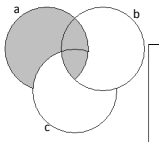
Modelo Booleano

- Presenta algunos problemas:
  - Decisión binaria, sin escala de relevancia. → $w_{ij} \in \{0,1\}$
  - Se basa más en data retrieval que en information retrieval.
  - Difícil traducir a una expresión booleana.
  - Las consultas son combinaciones de términos usando operadores and, or y not. Además, hay que buscar una representación óptima

31

Modelo Booleano

- Ejemplo de consulta en FND:



Consulta genérica

$q = k_a \wedge (k_b \vee \neg k_c)$

Consulta FND

$q = k_a \wedge (k_b \vee \neg k_c)$

$q = (k_a \wedge k_b) \vee (k_a \wedge \neg k_c)$

$q_{fnd} = (k_a \wedge k_b \wedge k_c) \vee (k_a \wedge k_b \wedge \neg k_c) \vee (k_a \wedge \neg k_b \wedge \neg k_c)$

$q_{fnd} = (1,1,1) \vee (1,1,0) \vee (1,0,0)$

32

Modelo Booleano

- **Definición.** Para el modelo booleano, los pesos de los términos son binarios ($w_{ij} \in \{0,1\}$). Una consulta es una expresión booleana convencional. Si q_{fnd} es la forma normal disjunta de una consulta, y q_{cc} alguno de los componentes de esta fnd, la similitud de un documento d_j con una consulta q se define como:

$$sim(d_j,q) = \begin{cases} 1 \text{ si } \exists\, q_{cc} \mid (q_{cc} \in q_{fnd}) \wedge (\forall k_i, g_i(d_j) = g_i(q_{cc})) \\ 0 \text{ en otro caso} \end{cases}$$

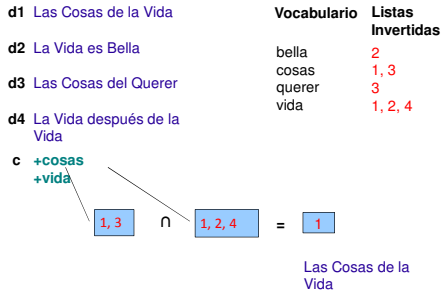
Si $sim(d_j,q)=1$, entonces el documento se predice como relevante. En cualquier otro caso, el documento no es relevante.

33

Consultas en el Modelo Booleano

1. Se buscan los términos de la consulta en el índice
2. Se recuperan las listas invertidas correspondientes a cada término
3. Se realizan las operaciones de conjuntos correspondientes sobre las listas (unión, intersección y/o diferencia)
4. Las listas están ordenadas en orden creciente de número de documento (distancias) → se puede operar recorriéndolas secuencialmente → los documentos se recuperan ordenados por número de documento

34



Vocabulario	Listas Invertidas
bella	2
cosas	1, 3
querer	3
vida	1, 2, 4

35

Modelo Vectorial

- Asume que el uso de pesos binarios es limitativo y propone un marco con posibilidad de relevancia parcial.
- Por tanto, se asignan pesos no binarios a los términos en los documentos
- Se pretende computar el grado de similitud entre documentos y consultas de forma gradual, y no absoluta.
- El resultado será un conjunto de documentos respuesta a una consulta ordenados en ranking de relevancia.

36

Modelo Vectorial

- **Definición.** En el modelo vectorial, el peso w_{ij} que se asocia a un par (k_i,d_j) es positivo y no binario. De igual modo, los pesos de los términos en una consulta se someten a los mismos pesos, de modo que $w_{iq} \geq 0$ es el peso asociado asociado al par $[k_i,q]$. El vector q se define como $q=(w_{1q}, w_{2q},..., w_{tq})$ siendo t el número total de términos indexados en el sistema. De igual forma, el vector documento se representa por $d_j=(w_{1j},w_{2j},...,w_{tj})$
- Por tanto, un documento y una consulta se representan como vectores t-dimensionales (vectores en un espacio de t dimensiones, siendo t el número de términos indexados en la colección de documentos).

37

Modelo Vectorial

- La similitud entre documentos y consultas se evalua a través de la correlación de los vectores que los representan, q y d_j.
- La correlación se puede definir a través del coseno del ángulo entre los vectores:

$$sim(\vec{d}_j,\vec{q}) = \frac{\vec{d}_j \bullet \vec{q}}{\left|\vec{d}_j\right| \times \left|\vec{q}\right|} = \frac{\sum_{i=1}^{t} w_{ij} \times w_{iq}}{\sqrt{\sum_{i=1}^{t} w_{ij}^2} \times \sqrt{\sum_{i=1}^{t} w_{iq}^2}}$$

38

Modelo Vectorial

- Sobre la fórmula del coseno
  - La norma del vector consulta no afecta al ranking porque es igual para todos los documentos, cosa que no pasa con la norma del vector documento
  - La similitud varía entre 0 y +1 puesto que así lo hacen los pesos de los términos de los vectores

39

Modelo Vectorial

- En este modelo, en lugar de predecir si un documento es o no relevante, se proporciona un grado de relevancia.
- Un documento podría ser recuperado sólo con una coincidencia parcial.
- Se establece un umbral de relevancia para decidir cuando mostrar un documento como relevante.
- El problema para obtener la relevancia consistirá en la forma de asignar pesos.

40

Modelo Vectorial

- Problema de clustering en IR: definir que documentos son relevantes y que documentos no lo son. Se pueden usar dos medidas para ello:
  - Similitud intra-cluster. Se puede utilizar como medida la frecuencia de términos (tf).
  - Diferencia inter-cluster. Se puede utilizar como medida la frecuencia de documento inversa (idf).
- Estas medidas (tf, idf) se pueden aplicar para el cálculo de los pesos de los términos.

41

Modelo Vectorial

- **Definición.** Sea N el total de documentos de una colección, y n_i los documentos en los que aparece el término k_i. La frecuencia del término k_i en el documento d_j la denotamos por $freq_{ij}$. La frecuencia normalizada del término k_i en el documento d_j es f. El máximo se obtiene sobre los términos del documento. La frecuencia de documento inversa será idf.

$$f = \frac{freq_{ij}}{\max freq_j}$$

$$idf_i = \log \frac{N}{n_i}$$

El peso del término en documentos y consultas se calcula con estas fórmulas empíricas:

$$w_{ij} = f_{ij} \times \log \frac{N}{n_i}$$

$$w_{iq} = \left(0{,}5 + \frac{0{,}5 \cdot freq_{iq}}{\max freq_q}\right) \times \log \frac{N}{n_i}$$

42

Modelo Vectorial

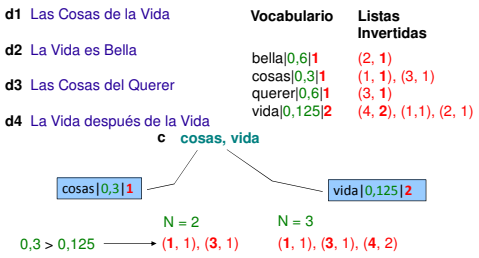
- Las principales ventajas del modelo son:
  - Se mejora el rendimiento con las fórmulas de obtención de pesos.
  - Se pueden recuperar documentos que se 'aproximen' a la consulta.
  - La fórmula del coseno proporciona, además, un ranking sobre la respuesta.
- La principal desventaja es que considera los términos como independientes, lo que puede causar bajo rendimiento (en teoría).

Modelo Vectorial

- Como conclusión:
  - Es muy elástico como estrategia de ranking en colecciones generales.
  - Es difícil de mejorar sin expansión de consultas o relevance feedback.
  - En comparación con otros modelos, es superior o igual en rendimiento a las alternativas.
  - Es simple y rápido.
  - Hoy en día, es uno de los más utilizados.

Consultas en el Modelo Vectorial

- Una consulta puede tener muchos términos, y sólo interesa obtener los N documentos más relevantes
- La lista de documentos de cada término se almacena ordenada por orden decreciente de apariciones del término en cada documento
- Se quiere obtener los N documentos con mayor similitud con la consulta
- Se comienza con la lista correspondiente al término con mayor peso global y se consideran los primeros N números de documento de la lista
- Si la lista tiene menos de N números de documento se sigue con la lista de otro término con peso global menos o igual al del anterior



$Sim(1, c) = p(cosas, 1) * 0,3 + p(vida, 1) * 0,125 = 1 * 0,3^2 + 1 * 0,125^2 = 0,1$
$Sim(3, c) = p(cosas, 3) * 0,3 + p(vida, 3) * 0,125 = 1 * 0,3^2 + 0 * 0,125^2 = 0,09$
$Sim(4, c) = p(cosas, 4) * 0,3 + p(vida, 4) * 0,125 = 0 * 0,3^2 + 2 * 0,125^2 = 0,03$