Recuperación de Información

Índice

Análisis Automático de Textos

Contar Palabras

FUENTE: 20858 palabras

Documento de 53 páginas sobre investigación

DICCIONARIO: 717 palabras

Palabras del capítulo de Introducción y de los apartados de Introducción del resto de capítulos (excepto nombres propios y números)

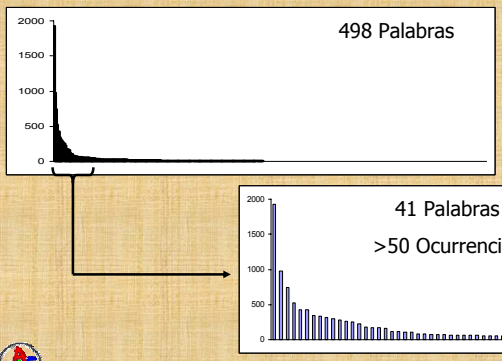
SIMPLIFICACIÓN: 498 palabras

Plurales, género y tiempos verbales

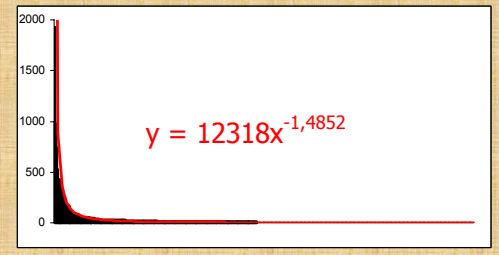
13055 ocurrencias (62.6%)

Palabras ordenadas por número de ocurrencias

Contar Palabras



Aproximación



Ley de Zipf

Frecuencia de la i-ésima palabra más frecuente $= \frac{1}{i^{\theta}}$ Frecuencia de la palabra más frecuente

El valor de θ depende del texto analizado

Aproximación: $\theta=1$

Realidad: $\theta \in (1.5,2.0)$

"Human Behaviour and the Principle of Least Effort" (1949)

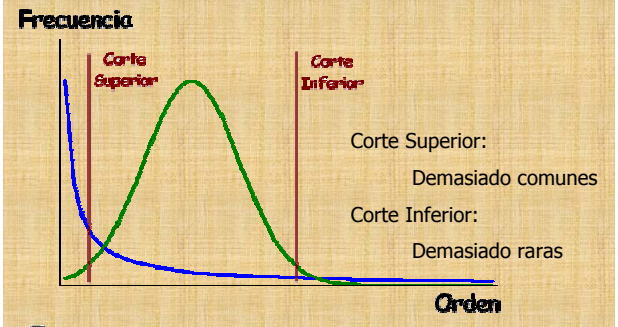
H. P. Luhn

"Se propone aquí que la frecuencia de ocurrencia de una palabra en un artículo podría ser una buena medida de la trascendencia de esa palabra en ese artículo"

The Automatic Creation of Literature Abstracts (1958)

El contenido de un artículo viene determinado por las palabras que tienen frecuencias de ocurrencia medias

Palabras Significativas



Corte Superior:

Demasiado comunes

Corte Inferior:

Demasiado raras

Análisis Léxico

Modelado

MODELADO DE LA INFORMACIÓN

1. Análisis Léxico

El Análisis Léxico es la primera etapa del proceso de Modelador de la Información

Consulta/Indexado

CONSULTA	ÍNDICE
Produce elementos en una representación adecuada para su comparación con los índices	Produce Términos de Índice candidatos que tendrán que continuar su proceso y eventualmente ser añadidos a los índices

Indización Automática

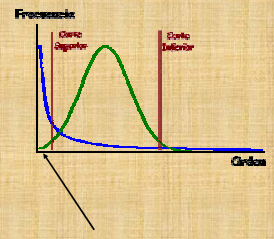
¿Qué Términos de Índice se deberían tomar para crear el esquema de indización?

Primera Hipótesis:

Tomar como Términos de Índice todas y cada una de las palabras que forman el documento

Parece una solución fácil, pero pronto aparecen problemas →

Palabras Vacías



Términos de Indización Pobres

Se genera una Lista de Palabras Vacías

Se deben eliminar como Términos de Índice

Palabras Vacías

Desde los primeros trabajos en Recuperación de Información se ha reconocido que muchas de las palabras más frecuentes ('el', 'de', 'y', ...) son desaconsejables como Términos de Índice

Una búsqueda que use cualquiera de esos términos recuperará, probablemente, casi todos los documentos de la bases de datos; independientemente de su relevancia puesto que su poder de discriminación es muy pobre

Palabras Vacías

Las palabras vacías constituyen una fracción muy importante del texto en la mayoría de documentos (las 10 palabras más frecuentes del Castellano suelen ocupar entre el 20% y el 30% de las palabras de un documento)

1 palabra	14.75 %	6 palabras	38.42 %
2 palabras	22.21 %	7 palabras	41.06 %
3 palabras	27.87 %	8 palabras	43.62 %
4 palabras	31.90 %	9 palabras	46.04 %
5 palabras	35.18 %	10 palabras	48.36 %

Palabras Vacías

No está muy claro qué palabras deben ser incluidas

En inglés, entre las 200 palabras más frecuentes están **'time' 'war' 'home' 'life' 'water' 'world'**

La lista de palabras vacías depende de la base de datos y de los potenciales usuarios

Las listas de palabras vacías en los sistemas comerciales tienden a ser conservadores

En Castellano, sólo incluyen los determinantes y las preposiciones

Números

La mayoría de los números no son buenos Términos de Índice y por ello no deben ser incluidos ↔

En algunos entornos pueden ser importantes:

-Número de caso en bases de datos legales

-Bases de Datos de documentos ténicos

Solución Parcial:

Permitir Términos de Índice que contengan números pero sin que empiecen por ellos

Problema:

Fechas cuando interesa buscar por ellas

Signos de Puntuación

Los guiones se usan para romper una palabra en sílabas al final de una línea

Los guiones pueden forman parte de un nombre (F-16)

Los puntos pueden formar parte de nombre de ficheros (COMMAND.COM)

La barra puede aparecer como parte de un nombre (OS/2)

En la Consulta deben tratarse como un error

Mayúsculas/Minúsculas

La aparición de Mayúsculas y/o Mayúsculas mezcladas no suele ser un problema para elegir los Términos de Índice

Normalmente, los analizadores léxicos convierten los caracteres a un único tipo de letra

En determinados entornos es importante mantener la distinción entre Mayúsculas y Minúsculas (lenguajes de programación)

Diferencias en la Consulta

El diseño para procesar consultas depende del diseño empleado para la indización

Los Términos de Índice de la consulta deben coincidir con los obtenidos en la indización

El analizador de consultas debe reconocer, además, los distintos operadores que puedan incluirse en la consulta (por ejemplo, operadores Booleanos)

Política del Analizador Léxico

Normas de Indización que se utilizan

Los resultados de recuperación dependerán directamente de la política utilizada en el analizador léxico

Coste Computacional

El análisis léxico es costoso, en comparación con las etapas posteriores, porque requiere examinar cada carácter de la entrada

No existen estudios exhaustivos → **50 %**

Es importante diseñar analizadores léxicos tan eficientes como sea posible

Ejercicio

Bloque 2 - Ejercicio 2.1

Extraer el corpus de palabras características de la colección de Documentos de la Tabla 1 (considerar que sólo son significativas las palabras que están resaltadas en el texto).

Lexematización

Modelado

MODELADO DE LA INFORMACIÓN

El proceso de lexematización simplifica la profundidad de los índices asociando variedades morfológicas

1. **Análisis Léxico**
2. **Lexematización**

¿Cuándo Lexematizar?

Durante la Indización

Mayor eficiencia y menor tamaño de índices

Durante la Consulta

No se pierde información sobre los términos completos

Taxonomía

Lexematización

- **Manual** → Usando expresiones regulares
- **Automática** → Programas lexematizadores

Una buena lexematización reduce el tamaño de los índices hasta en un 50%

Taxonomía

Lexematización

- Manual
- **Automática**
  - Eliminación de Afijos
    Eliminan sufijos o prefijos
  - Variedad de Sucesores
    Usan las frecuencias de las secuencias de letras
  - Búsqueda en Tabla
    Utilizan tablas que relacionan los términos y sus lexemas
  - N-Gramas
    Unen términos en función del número de n-gramas que comparten

Valoración

Exactitud, Efectividad de la recuperación y Nivel de compresión de los índices

No suelen ser juzgados por su exactitud lingüística

Hiperlexematización: eliminar demasiados caracteres (puede juntar términos que no están relacionados y causar la recuperación de documentos no relevantes

Hipolexematización: eliminar menos caracteres de lo debido (puede impedir que se junten términos relacionados y causar la no recuperación de documentos relevantes)

Eliminación de Afijos

Se eliminan los prefijos y/o los sufijos

La mayoría de lexematizadores que emplean esta técnica son de eliminación de afijos por coincidencia más larga (eliminan de una palabra la cadena de caracteres más larga posible de acuerdo a un conjunto de reglas)

El conjunto de reglas empleado es crítico en la calidad del lexematizador

Búsqueda en Tabla

Consiste en almacenar en una tabla todos los Términos de Índice y su correspondiente lexema

Término	Lexema
representación	representar
representante	representar
representar	representar
...	...

Los términos de las consultas y de los índices se lexematizan a través d búsquedas en las tablas

Problemas

No existen tablas estándar

(ni para Castellano ni para Inglés)

Si existieran, muchos de los términos encontrados en los documentos no estarían representados porque son dependientes de la temática y no pertenecen al vocabulario estándar

Sobrecarga de almacenamiento

Variedad de Sucesores

Están basados en trabajos de **Lingüística Estructural** que intentan determinar los límites de las palabras y los morfemas basándose en la distribución de fonemas en un gran cuerpo de pronunciaciones

Las versiones simplificadas del método usan letras en lugar de fonemas

Definición

Sea α una palabra de longitud n:

α_i es un prefijo de longitud i de α

Sea D el corpus de palabras:

$D(\alpha_i)$ es el subconjunto de D que contiene aquellos términos cuyas primeras i letras coinciden con α_i.

La variedad de sucesores de α_i, $S(\alpha_i)$, se define como el número de letras distintas que ocupan la posición i+1 en las palabras de $D(\alpha_i)$

Definición

Una palabra de longitud n tienen n variedades de sucesores:

$S(\alpha_1)$, $S(\alpha_2)$, ..., $S(\alpha_i)$, ..., $S(\alpha_n)$

La variedad de sucesores de una cadena es el número de caracteres diferentes que siguen a esa cadena en las palabras que forman el cuerpo de un texto

Ejemplo

Cuerpo de Texto:

danza decir dedo delta
dispar dolor duro

Variedad de Sucesores **decir**

d (5)	a,e,i,o,u
de (3)	c,d,l
dec (1)	i
deci (1)	r
decir (1)	∅

Ejemplo

Cuerpo de Texto:

ente	entender	entendido
entendimiento	entereza	enternecedor
enternecer	entero	enterrar

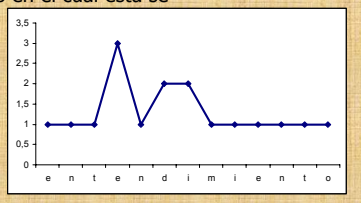
Variedad de Sucesores **entendimiento**

e (1)	n	entendi (2)	m,d
en (1)	t	entendim (1)	i
ent (1)	e	entendimi (1)	e
ente (3)	∅,n,r	entendimie (1)	n
enten (1)	d	entendimien (1)	t
entend (2)	i,e	entendimient (1)	o
		entendimiento (1)	∅

Segmentación de Palabras

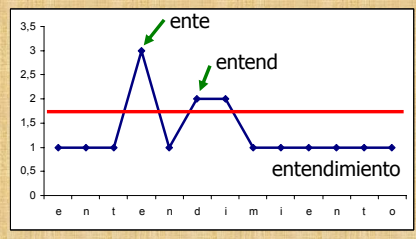
Cuando se lleva a cabo el proceso de extracción de variedad de sucesores en un cuerpo de texto suficientemente grande, la variedad de sucesores de una subcadena de un término disminuye según se le añaden caracteres hasta que se alcanza el **límite de un segmento** momento en el cual esta se incrementa súbitamente

Esta información puede usarse para segmentar la palabra



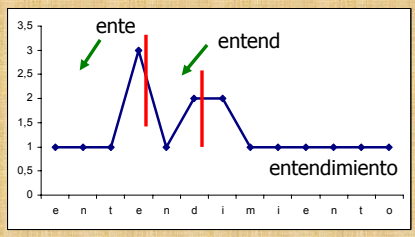
Valor de Corte

Método del Valor de Corte: Se selecciona un valor de corte para las variedades de sucesores y se identifica un límite cada vez que se alcanza ese valor



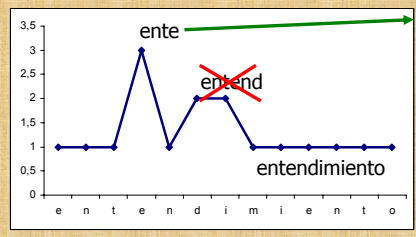
Picos y Valles

Método de Picos y Valles: Se hace el corte del segmento después de los caracteres cuya variedad de sucesores excede a la del carácter que lo precede y a la del que lo sigue



Palabras Completa

Método de Palabra Completa: Se hace el corte después de un segmento si éste es una palabra completa en el corpus



ente
entender
entendido
entendimiento
entereza
enternecedor
enternecer
entero
enterrar

Entropía

Método de la Entropía: Este método aprovecha la distribución de probabilidad de las variedades de sucesores

Ejercicio

Bloque 2 - Ejercicio 2.2

Dado el corpus de palabras del ejercicio 2.1, elegir los Términos Índice que las representan utilizando la técnica de Variedad de Sucesores.

Selección de Lexemas

Algunos investigadores han propuesto la siguiente regla:

si	el primer segmento aparece en más de 12 palabras del corups
entonces	el primer segmento es el lexema
sino	el segundo segmento es el lexema

La condición está basada en la observación de que si un segmento aparece en más de 12 palabras será con mucha probabilidad un prefijo

No se considera lexema a ningún segmento más allá del segundo debido a la escasez de prefijos múltiples

N-gramas

di-grama	2 letras consecutivas de una palabra
tri-grama	3 letras consecutivas de una palabra
...	
n-grama	n letras consecutivas de una palabra

Medidas de asociación entre pares de términos

basadas

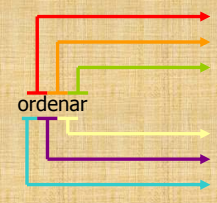
n-gramas únicos compartidos

Tradicionalmente se incluye como método de lexematización

En realidad no se produce tal lexematización

Di-gramas Únicos

ordenar
ordenación
radiación
radioactividad



di-gramas únicos de cada palabra

ordenar	ar, de, en, na, or, rd
ordenación	ac, ci, de, en, io, na, on, or, rd
radiación	ac, ad, ci, di, ia, io, on, ra
radioactividad	ac, ad, ct, da, di, id, io, iv, oa, ra, ti, vi

Medida de Similitud

Una vez se han identificado los di-gramas únicos se calcula una medida de similitud para dos palabras basada en el **coeficiente de Dice**

$$S = \frac{2C}{A + B}$$

A	número de di-gramas únicos de la primera palabra
B	número de di-gramas únicos de la segunda palabra
C	número de di-gramas únicos compartidos

Matriz de Similaridad

S

	ordenar	ordenación	radiación	radioactividad
ordenar	**1**	**0.67**		
ordenación	**0.67**	**1**	**0.47**	**0.19**
radiación		**0.47**	**1**	**0.5**
radioactividad		**0.19**	**0.5**	**1**

Simétrica y Escasa

Si se usa un valor umbral de **0.6**, la mayoría de los agrupamientos que se forman son correctos y en casi ningún caso se producen asociaciones falsas

Ejercicio

Bloque 2 - Ejercicio 2.3

Elegir los Términos Índice que representan el corpus del Problema 2.1 utilizando la Técnica de N-Gramas.

Compresión

Dado que un lexema es normalmente más corto que la palabra a la cual corresponde, almacenar lexemas en lugar de palabras decrementa el tamaño del fichero de índice

Se ha comprobado experimentalmente que los porcentajes de disminución oscilan entre el **26%** y el **50%** en función del lexematizador utilizado y de los textos empleados

Ejercicio

Bloque 2 - Ejercicio 2.4

Comparar los resultados obtenidos en los Problemas 2.2 y 2.3. ¿Qué conjunto de Términos Índice es mejor? ¿Por qué?

Tesauros

Función del Tesauro

En los sistemas de Recuperación de Información se usan para coordinar los procesos básicos de indización y recuperación

vocabulario común, preciso y controlado

\+ Si la búsqueda no recupera suficientes documentos se puede usar el tesauro para expandir la petición a través de sus enlaces

\- Si la búsqueda recupera demasiados documentos el tesauro puede sugerir términos de búsqueda más específicos

Los tesauros son dependientes del dominio

Construcción

Existe una vasta literatura sobre los principios, metodología y problemas implicados en la construcción de tesauros

Sólo una pequeña parte está dedicada a la construcción automática de tesauros

Estado actual de la construcción de tesauros

Abundancia de tesauros generados de forma manual

Escepticismo sobre la posibilidad de automatizar completamente la construcción de tesauros

Características

Nivel de Coordinación

La coordinación hace referencia a la construcción de frases a partir de términos individuales

Relaciones entre Términos

Las relaciones entre términos constituyen el aspecto más importante del tesauro dado que proporcionan interconexiones semánticas en el vocabulario

Coordinación

Tesauro Precoordinado

Las frases están disponibles para la indización y la consulta; Hace el vocabulario muy preciso e incorpora frases hechas; Más frecuente en tesauros manuales

Tesauro Postcoordinado

Las frases se construyen durante la consulta; Simplifica el proceso de consulta porque el buscador no necesita conocer las reglas de construcción de frases; Más frecuente en tesauros automáticos

Relaciones

Identificar las relaciones requiere conocimiento intenso del dominio para el que se está diseñando el tesauro

Relaciones de Equivalencia:

'genética'	'herencia'	por significado
'aspereza'	'suavidad'	por valor de propiedad

Relaciones Jerárquicas:

'perro'	'pastor alemán'	por género-especie

Relaciones no-Jerárquicas:

'autobús'	'asiento'	por cosa-parte
'rosa'	'fragancia'	por cosa-atributo

Difíciles de identificar por métodos automáticos

Construcción Manual

El proceso de construir tesauros es un arte y una ciencia

Se han de definir los límites del área sujeto; Se particiona el dominio en divisiones o subáreas; Esto implica la identificación de áreas centrales y periféricas dado que es improbable que todos los temas sean de igual importancia

Se deciden las características que se desean para el tesauro

Se recolectan términos de cada subárea (fuentes: índices, enciclopedias, manuales, libros de texto, periódicos, revistas, catálogos, tesauros previos existentes, expertos, ...)

Se analiza cada término para localizar su vocabulario relacionado (sinónimos, términos ensanchadores, términos estrechadores, definiciones,...)

Construcción Manual

En cada subárea, los términos y sus relaciones se organizan en estructuras jerárquicas (este proceso puede revelar la existencia de huecos que lleven a identificar la necesidad de nuevos niveles en la jerarquía, a agrupar sinónimos que no fueron reconocidos, a sugerir nuevas relaciones, a reducir el tamaño del vocabulario, ...)

Se organiza el tesauro alfabéticamente

Se revisa el tesauro, verificando la consistencia de las relaciones y la creación de frases

Se prueba por expertos y se incorporan sus sugerencias

Se actualiza periódicamente (proceso usualmente lento)

Construcción Automática

Alternativas:

Generar tesauros a partir de una colección de documentos aplicando métodos estadísticos sobre la información sintáctica para extraer información semántica

Generar tesauros mezclando tesauros existentes

Generar tesauros mediante herramientas de sistemas expertos usando información obtenida del usuario

Caracterización Formal

Recuperar Información

Los sistemas de Recuperación de Información por contenido tradicionales asumen que **la semántica**, de cada documento de la base de datos y de las necesidades de información de cada usuario, **puede expresarse mediante un conjunto de Términos Índice**

Es un simplificación del problema real

La comparación entre cada documento y la consulta realizada por el usuario se lleva a cabo en este espacio de Términos Índice que resulta muy impreciso

No es raro que los documentos recuperados en una consulta de usuario expresada como un conjunto de palabras resulten frecuentemente irrelevantes

Ranking

El problema central de los sistemas de Recuperación de Información es predecir que documentos son los más relevante

Los algoritmos de Ranking caracterizan la noción de documento relevante en función de una serie de premisas

Según las premisas que se adopten se producirán los distintos modelos de Recuperación de Información

Definición

Un Modelo de Recuperación de Información es una cuadrupla [D, Q, F, R(q_i, d_j)] donde

D es un conjunto compuesto por representaciones de los documentos de la colección a analizar

Q es un conjunto compuesto por representaciones de las necesidades de información de los usuarios

F es un espacio de trabajo para modelar las representaciones de los documentos, de las consultas y de sus relaciones

R(qi, dj) es una función de ranking que asocia un número real a una consulta $q_i \in Q$ y a una representación de documento $d_j \in D$

Planteamiento

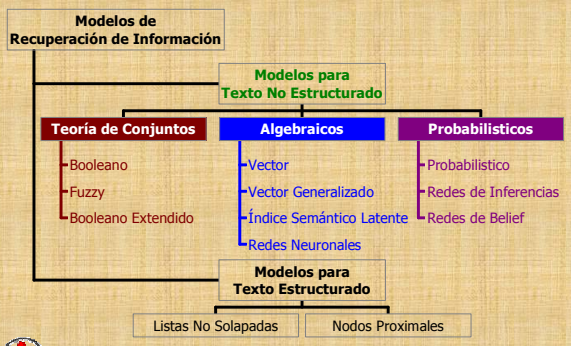
Para poder plantear un modelo de Recuperación de Información necesitamos varios aspectos

Modelado de la Información: Una representación de los documentos y de las necesidades de información del usuario

Modelado del Sistema: Un espacio de trabajo

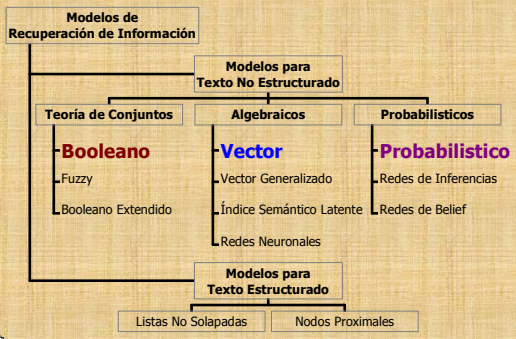
Función de Ranking: Define el orden de relevancia de los documentos respecto a una consulta concreta

Taxonomía



Modelos Clásicos

Taxonomía



Términos Índice

Los modelos clásicos asumen que cada documento puede ser **representado por un conjunto de palabras** clave llamadas Términos Índice

Suelen ser los sustantivos y verbos del texto puesto que, normalmente, los adjetivos, adverbios y conectores tienen menos información semántica

Algunos buscadores Web utilizan todas las palabras distintas de los documentos como Términos Índice

Relevancia Relativa

No todos los Términos Índice tienen igual importancia en la descripción de un documento

Decidir la importancia de cada Término Índice a la hora de resumir el contenido de un documento no es tarea fácil

Se asocia un peso a cada Término Índice

Pesos de los Términos Índice

Sea t el número de Términos Índice en el sistema y sea k_i un Término Índice genérico

$K=(k_1, k_2, \ldots, k_i, \ldots, k_t)$ conjunto formado por todos los Términos Índice

Cada Término Índice k_i en cada documento d_j tiene asociado un peso $w_{i,j} \geq 0$

Cada documento tiene asociado un vector de Términos Índice

$$\vec{d}_j = (w_{1,j}, w_{2,j}, \ldots, w_{t,j}) \qquad g_i(\vec{d}_j) = w_{i,j}$$

Característica

Se asume que los pesos de los Términos Índice son mutuamente independientes

Esto es una simplificación porque en realidad la ocurrencia de Términos Índice en los documentos tiene valores de correlación no nulos

Extraer información de la correlación de los Términos Índice para mejorar los algoritmos de ranking no es una tarea sencilla

Modelo Booleano

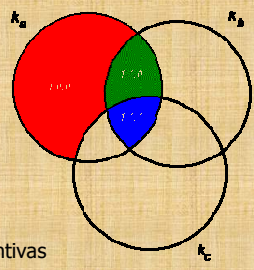
Los pesos de los Términos Índice son binarios $w_{i,j} \in \{0,1\}$

Las consultas q son expresiones Booleanas convencionales en forma normal disjuntiva

$$q = k_a \cap (k_b \cup \bar{k}_c)$$

$$\vec{q}_{fnd} = (1,1,1) \cup (1,1,0) \cup (1,0,0)$$

componentes conjuntivas



Similaridad

La **similaridad** de un documento d_j respecto a una consulta q se define como

$$sim(d_j, q) = \begin{cases} 1 & \text{si } \exists \vec{q}_{cc} \in \vec{q}_{fnd} \mid \forall k_i \; g_i(\vec{d}_j) = g_i(\vec{q}_{cc}) \\ 0 & \text{otro caso} \end{cases}$$

Si la **similaridad entre d_j y q es 1**, el Modelo Booleano predice que el documento **d_j es relevante para la consulta q**

Consultas

Un documento es **relevante si y sólo si**...

Una palabra:	contiene la palabra
A and B:	contiene las dos palabras
A or B:	contiene una de las dos palabras
A pero no B:	contiene A pero no contiene B

Es el modelo **más primitivo**, y **bastante malo**

"Necesito investigar sobre los Griegos y los Romanos"
→ Griegos and Romanos

Ventajas / Problemas

Simplicidad

Formalismo claro

Suele recuperar o muy pocos o demasiados documentos

No los ordena por relevancia

Modelo Vector

Los pesos de los Términos Índice son **positivos** y **no-binarios** $w_{i,j} \geq 0$

Las consultas q vienen definidas como

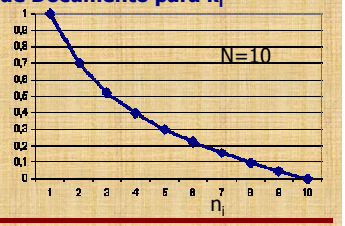
$$\vec{q} = (w_{1,q}, w_{2,q}, \ldots, w_{t,q})$$

Frecuencia Inversa (D)

Sea **N** el número total de documentos existentes en el sistema y sea n_i el número de documentos en los cuales aparece el Término Índice k_i

La **Frecuencia Inversa de Documento para k_i** viene dada por

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



Frecuencia Normalizada (TI)

Sea $freq_{i,j}$ la frecuencia del Término Índice k_i en el documento d_j (por ejemplo, el número de veces que aparece el Término Índice en el texto del Documento)

La **Frecuencia Normalizada del Término Índice k_i** en el documento d_j viene dada por

$$f_{i,j} = \frac{freq_{i,j}}{\max_l freq_{l,j}}$$

Pesos

Los **pesos para los documentos** vienen dados por

$$w_{i,j} = f_{i,j} fid_i = \frac{freq_{i,j}}{\max_l freq_{l,j}} \log \frac{N}{n_i}$$

Los **pesos para las consultas** vienen dados por

$$w_{i,q} = \left(0.5 + \frac{0.5\, freq_{i,q}}{\max_l freq_{l,q}}\right) \log \frac{N}{n_i}$$

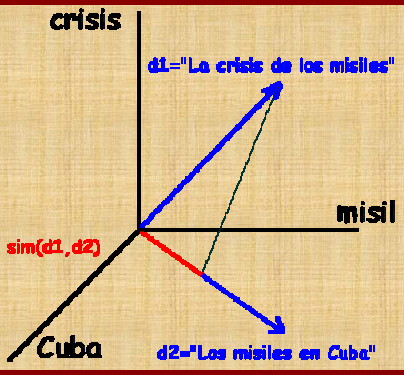
Similaridad

La **similaridad** de un documento d_j respecto a una consulta q se define como

$$sim(d_j, q) = \frac{\vec{d}_j \bullet \vec{q}}{|\vec{d}_j||\vec{q}|} = \frac{\sum_{i=1}^{t} w_{i,j} w_{i,q}}{\sqrt{\sum_{i=1}^{t} w_{i,j}^2} \sqrt{\sum_{i=1}^{t} w_{i,q}^2}}$$

La **similaridad varía entre 0 y 1** y el Modelo Vector no sólo predice que el documento **d_j es relevante para la consulta q** sino que además **ordena los documentos relevantes** de acuerdo a su grado de similaridad respecto a la consulta

Ejemplo



Ventajas / Problemas

El esquema de pesos que utiliza mejora el rendimiento de la recuperación

Se estrategia de comparación parcial permite recuperar documentos que aproximan las condiciones de la consulta

Su fórmula de ranking ordena los documentos en función de su similaridad a la consulta

En el cálculo de los pesos se asume la independencia mutua entre los Términos Índice

Ejercicio

Bloque 3 - Ejercicio 3.1

Suponiendo el conjunto de Términos Índice elegido en el ejercicio 2.4. ¿Cuál será el documento más relevante de la tabla 1, frente a la consulta "dar una pastilla a un gato en casa" si el Modelo empleado es el Modelo Vector?

Modelo Probabilístico

Los pesos de los Términos Índice son **binarios** $w_{i,j} \in \{0,1\}$

Una consulta es un subconjunto de Términos Índice

Los pesos de las consultas son binarios $w_{i,q} \in \{0,1\}$

Probabilidad

Para una consulta q

Conjunto de **documentos relevantes** R

Conjunto de **documentos no relevante** $\overline{R}$

Probabilidad de que el documento d_j sea **relevante**

$$P(R|\vec{d}_j)$$

Probabilidad de que el documento d_j sea **no relevante**

$$P(\overline{R}|\vec{d}_j)$$

Similaridad

La **similaridad** de un documento d_j respecto a una consulta q se define como la relación

$$sim(d_j,q) = \frac{P(R|\vec{d}_j)}{P(\overline{R}|\vec{d}_j)}$$

$$P(X|Y) = \frac{P(Y|X)P(X)}{P(Y)}$$

$$sim(d_j,q) = \frac{P(\vec{d}_j|R)P(R)}{P(\vec{d}_j)} \frac{P(\vec{d}_j)}{P(\vec{d}_j|\overline{R})P(\overline{R})}$$

Regla de Bayes

$$sim(d_j,q) = \frac{P(\vec{d}_j|R)P(R)}{P(\vec{d}_j|\overline{R})P(\overline{R})}$$

Similaridad

$$sim(d_j,q) = \frac{P(\vec{d}_j|R)P(R)}{P(\vec{d}_j|\overline{R})P(\overline{R})}$$

$P(\vec{d}_j|R)$ $P(\vec{d}_j|\overline{R})$ Probabilidad de que el documento d_j pertenezca a los conjuntos R $\overline{R}$

$P(R)$ $P(\overline{R})$ Probabilidad de que un documento seleccionado aleatoriamente sea relevante / no relevante

$P(R)$ $P(\overline{R})$ Iguales para todos los documentos

$$sim(d_j,q) \approx \frac{P(\vec{d}_j|R)}{P(\vec{d}_j|\overline{R})}$$

Similaridad

...asumiendo independencia probabilística entre los Términos Índice

$$sim(d_j,q) \approx \frac{\prod_{g_i(\vec{d}_j)=1} P(k_i|R) \prod_{g_i(\vec{d}_j)=0} P(\overline{k}_i|R)}{\prod_{g_i(\vec{d}_j)=1} P(k_i|\overline{R}) \prod_{g_i(\vec{d}_j)=0} P(\overline{k}_i|\overline{R})}$$

$P(k_i|R)$ $P(\overline{k}_i|R)$ $P(k_i|\overline{R})$ $P(\overline{k}_i|\overline{R})$ Probabilidad de que el Término Índice k_i esté o no presente en un documento seleccionado aleatoriamente de los conjuntos R $\overline{R}$

Similaridad

...tomando logarítmos, ignorando factores constantes para todos los documentos en el contexto de una consulta dada y aplicando que $P(k_i|R)+P(\overline{k}_i|R)=1$

$$sim(d_j,q)\approx\sum_{i=1}^{t} w_{i,q}w_{i,j}\left[\log\frac{P(k_i|R)}{1-P(k_i|R)}+\log\frac{1-P(k_i|\overline{R})}{P(k_i|\overline{R})}\right]$$

¿ $P(k_i|R), P(k_i|\overline{R})$?

Estimación de Probabilidades

Estado inicial (después de especificar la consulta)

Se **asume** que la probabilidad de que k_i esté en un documento seleccionado del conjunto R es constante para todos los k_i

$$P(k_i|R)=0.5$$

Se **asume** que la probabilidad de que k_i esté en un documento seleccionado del conjunto $\overline{R}$ puede aproximarse mediante la distribución de Términos Índice sobre todos los documentos

$$P(k_i|\overline{R})=\frac{n_i}{N}$$

n_i es el número de documentos que contienen k_i
N es el número total de documentos

Estimación de Probabilidades

Sea V el subconjunto de documentos recuperados por la etapa inicial (por ejemplo, los r primeros según ranking)

Sea V_i el subconjunto de V compuesto por los documentos que contienen a k_i

$$P(k_i|R)=\frac{\#V_i}{\#V} \qquad P(k_i|\overline{R})=\frac{n_i-\#V_i}{N-\#V}$$

$$P(k_i|R)=\frac{\#V_i+0.5}{\#V+1} \qquad P(k_i|\overline{R})=\frac{n_i-\#V_i+0.5}{N-\#V+1}$$

$$P(k_i|R)=\frac{\#V_i+\frac{n_i}{N}}{\#V+1} \qquad P(k_i|\overline{R})=\frac{n_i-\#V_i+\frac{n_i}{N}}{N-\#V+1}$$

Ventajas / Problemas

Los documentos se recuperan ordenados por su probabilidad de ser relevantes

Necesita una separación inicial entre documentos relevantes y no relevantes

El método no tiene en cuenta la frecuencia con la cual los Términos Índice aparecen en lños documentos pues los pesos son binarios

En el cálculo de las probabilidades se asume la independencia mutua entre los Términos Índice

Comparación

El modelo Booleano se considera cómo el método clásico por excelencia; aún se usa en algunos sistemas de Recuperación por Contenido

No existe acuerdo a la hora de comparar el rendimiento los modelos vector y probabilístico

Según algunas medidas realizadas, el modelo vector se comporta mejor que el modelo probabilístico para colecciones generales y peor para temáticas

El modelo dominante actual es el Modelo Vector y sus variantes

Bibliografía

Libros Castellano

Libros Inglés

Modern Information Retrieval

Ricardo Baeza-Yates y Berthier Ribeiro-Neto; Addison Wesley Longman Limited; ISBN: 0-201-39829-X; 1999

Information Retrieval

C. J. van Rijsbergen; Departament of Computing Science; University of Glasgow; 1979
http://www.dcs.gla.ac.uk/Keith/Preface.html

Artículos

Páginas web

Las Posibilidades de la Ley de Zipf en la Indización Automática

Rubén Urbizagástegui Alvarado; www.debiblioteconomia.com

Almacenamiento y Recuperación de Información Textual

Octavio Santana Suárez y Octavio Mayor González; Grupo de Estructuras de Datos y Lingüística Computacional del Departamento de Informática y Sistemas de la Universidad de Las Palmas de Gran Canaria.
http://protos.dis.ulpgc.es/docencia/seminarios/rit/