Refinamiento de la búsqueda

El proceso de recuperación se inicia con una interrogación inicial

la respuesta se utiliza para reformular la pregunta de forma más precisa (usuario) o más adecuada (sistema).

El objetivo es obtener un conjunto nuevo de términos para recuperar un grupo mayor de documentos relevantes.

Slide 1

- Relevance feedback

  Información proporcionada por el usuario.

- Técnicas de clustering

  Información obtenida de la respuesta.

- Técnicas de tesauros

  Información de la colección de documentos que se consulta.

Sólo en el primer caso se requiere la cooperación interactiva del usuario.

Relevance feedback

Se presenta al usuario una lista con los documentos recuperados y se le pide que marque los que son relevantes.

En casos reales el usuario examina $\sim$ 10/20 documentos

Se refleja esta información mediante dos mecanismos

- Expansión de la interrogación
- Revaloración (automática) de los términos

Slide 2

Dependiendo del modelo de RI utilizado se utilizan diferentes métodos.

Relevance feedback: Modelo vectorial

Principio: documentos relevantes tienen que tener representaciones similares y documentos no relevantes representaciones diferentes de los relevantes.

Reutilizar esta información para refinar la interrogación, en este caso un vector.

Notación:

D_r es el conjunto de documentos relevantes (los identificados por el usuario).

Slide 3

D_n es el conjunto de los documentos no relevantes recuperados.

C_r conjunto de todos los documentos relevantes (en toda la colección)

Si conemos C_r la interrogación óptima es

$$\vec{q}_{opt} = \frac{1}{|C_r|} \sum_{\vec{d}_j \in C_r} \vec{d}_j - \frac{1}{N - |C_r|} \sum_{\vec{d}_j \notin C_r} \vec{d}_j .$$

Como no conocemos C_r aproximaremos el conjunto mediante un proceso iterativo de relevance feedback.

Presentamos tres propuestas que utilizan los valores

α, β y γ como parámetros constantes a ajustar.

- **Standard Rochio**

$$\vec{q}_m = \alpha \vec{q} + \frac{\beta}{|D_r|} \sum_{\vec{d}_j \in D_r} \vec{d}_j - \frac{\gamma}{N - |D_n|} \sum_{\vec{d}_j \notin D_n} \vec{d}_j$$

- **Ide Regular**

Slide 4

$$\vec{q}_m = \alpha \vec{q} + \beta \sum_{\vec{d}_j \in D_r} \vec{d}_j - \gamma \sum_{\vec{d}_j \notin D_n} \vec{d}_j$$

- **Ide Dec Hi**

$$\vec{q}_m = \alpha \vec{q} + \beta \sum_{\vec{d}_j \in D_r} \vec{d}_j - \gamma \vec{dm}$$

donde dm es el documento no relevante con mayor puntuación.

Relevance feedback: Modelo probabilista

La función de similaridad en el modelo probabilistico viene dada por la ecuación

$$sim(d_j, q) = \sum_{i=1}^{t} w_{i,q} w_{i,j} \left(\log \frac{Pr(t_i|R)}{1 - Pr(t_i|R)} + \log \frac{Pr(t_i|\overline{R})}{1 - Pr(t_i|\overline{R})} \right).$$

Como ya vimos los valores $Pr(t_i|R)$ y $Pr(t_i|\overline{R})$ son desconocidos y se aproximan mediante un proceso iterativo automático.

Slide 5 La información proporcionada por el usuario se incorpora en un proceso iterativo de estimación ligeramente diferente.

- Distribución inicial: no cambia

  $Pr(t_i|R) = 0.5$ y $Pr(t_i|\overline{R}) = \frac{n_i}{N}$

  donde N es el número total de documentos y n_i es el número de documentos que contienen el término t_i.

- Ajuste:

$$Pr(t_i|R) = \frac{|D_{r,i}|}{|D_r|} \qquad Pr(t_i|\overline{R}) = \frac{n_i - |D_{r,i}|}{N - |D_r|}$$

  donde D_r es el número de documentos relevantes recuperados y $D_{r,i}$ es el número de documentos en D_r que contienen el término t_i.

La última ecuación admite las mismas variantes que en el modelo probabilista, cuando los valores son pequeños y se puede perder precisión.

Slide 6 Principio básico: El proceso es correcto si los términos son independientes y los pesos son booleanos.

Evaluación de métodos con relevance feedback

- La forma normal de medir precisión y recall no proporciona una medida adecuada, la medida se desvirtúa ya que se concen algunos documentos relevantes para el usuario y esto hace que mejoren notablemente las curvas.
- Se valoran precisión y recall sobre el conjunto residual, todos los documentos menos los documentos proporcionados por el usuario.

Slide 7

Métodos basados en clustering

Obtención automática de nuevos términos a añadir a la consulta a partir de los documentos recuperados o de la colección.

Se pretende identificar términos que estén relacionados con los que aparecen en la interrogación. Por ejemplo, sinónimos, stemming variations, o términos que aparecen en el texto próximos a los de la interrogación (máximo k palabras), o una combinación de varios criterios.

Slide 8

Estos métodos se implemetan sin intervención del usuario, requieren acceso a los documentos.

Marco general

Sea V un conjunto no vacío de palabras donde todas ellas son una variación gramatical de la misma raíz s (stem).

$$V = \{polish, polishing, polished\} \qquad s = polish$$

Supondremos que los documentos han sido procesadod de manera que todos los términos son stems.

Slide 9

Notación:

D el conjunto de documentos.

V formado por todas las palabras que aparecen en D.

S son los stems de V.

Objetivo: asociar a cada $s \in S$ un conjunto (cluster) $S_s(n)$ de stems que se utilizarán como sinónimos de búsqueda.

Veremos tres tipos de clusters asociación, métricos y escalares

La técnica es siempre la misma, definir una matriz M que contine los valores de correlación a decuados y asociar a cada stem los stems con mayor correlación.

Slide 10

Clusters de asociación

Intenta explotar información de co-ocurrencia de términos.

$f(s, d)$ es la frecuencia del stem s en el documento d.

M es una matriz con $|S|$ filas y $|D|$ columnas donde $m_{i,j} = f(s_i, d_j)$.

La matriz $C = M \times M^{\mathrm{T}}$ se interpreta como la matriz de correlación de stems,

Slide 11

$$c_{a,b} = \sum_{d \in D} f(s_a, d) f(s_b, d),$$

que cuantifica la frecuencia absoluta de co-ocurrencia de stems.

Se define la matriz de asociación Q que puede ser C o su versión normalizada:

$$q_{a,b} = \frac{c_{a,b}}{c_{a,a} + c_{b,b} - c_{a,b}}.$$

Sea $S_a(n)$ una función que toma la fila a-ésima de Q y devuelve el conjunto de stems que corresponde a los n valores mayores de $q_{a,b}$.

$S_a(n)$ es el cluster de asociación alrededor de s_a.

Se habla de cluster no normalizado cuando $Q = C$

y de cluster normalizado cuando Q se obtiene a partir de los valores normalizados.

Slide 12

Clusters métricos

Intenta utilizar distancia (número de palabras) entre términos.

Definimos la distancia entre palabras:

$r(u, v) =$ el número mínimo de palabras entre ellas si ambas aparecen juntas en algún documento en d, ∞ en caso contrario.

Y la correlación entre dos stems como

Slide 13

$$c_{a,b} = \sum_{u \in V(s_a)} \sum_{u \in V(s_b)} \frac{1}{r(u, v)}$$

A partir de C se define la matriz Q que puede ser C o su versión normalizada

$$q_{a,b} = \frac{c_{a,b}}{|V(s_a)| \times |V(s_b)|}.$$

Sea $M_a(n)$ una función que toma la fila a-ésima de Q y devuelve el conjunto de stems que corresponde a los n valores mayores de $m_{a,b}$.

$M_a(n)$ es el cluster métrico alrrededor de s_a.

Cluster escalares

Intenta valorar la aparición de sinónimos comunes.

Si s_a y s_b son dos stems se pretende comparan los valores de co-ocurrencia.

Utiliza los mismos principios que el modelo vectorial.

A cada stem s se le asocia un vector $\vec{s}$: la fila correspondiente en la matriz de correlación.

Via producto escalar se cuantifica la correlación entre stems

Slide 14

$$s_{a,b} = \frac{\vec{s_a} \cdot \vec{s_b}}{|\vec{s_a}| \times |\vec{s_b}|}$$

Sea $E_a(n)$ una función que devuelve el conjunto de stems que corresponde a los n valores mayores de $s_{a,b}$.

$E_a(n)$ es el cluster escalar alrrededor de s_a.

Reformulación automática

Partimos de la interrogación q, D es el conjunto de documentos recuperados.

V y S se definen como antes a partir de D.

Para cada stem s presente en q, seleccionamos m stems del cluster asociado $S_s(n)$ y los añadimos a la interrogación.

Así obtenemos una nueva interrogación q'.

Slide 15

Parámetros y elementos a ajustar

- n y m.
- clusters ¿normalizados o no? o ¿ambos?
- tipo de clustering

Técnicas de tesauro

Construcción de un tesauro basado en

- relaciones termino-término, o
- categorización de documentos

El objetivo es obtener una definición de concepto por agrupación de elementos similares.

Son técnicas globales, necesitan toda la colección de documentos,

Slide 16

es una técnica computacionalmente costosa,

pero puede ser útil en colecciones estables, ya que sólo se tiene que calcular una vez.

Tesauro por similaridad de términos

Se trata de hacer un clustering sobre términos, de forma similar a como se hizo con documentos.

Ahora el espacio es el conjunto de términos en vez del de documentos.

Notación

k es el número de términos en C.

Slide 17

N el número de documentos.

$f_{i,j}$ la frecuencia del término t_i en el documento d_j.

k_j el número de términos de índice que aparecen en el documento d_j.

$itf_j = \log \frac{k}{k_j}$ inverse term frequency del documento d_j.

A cada término se le asocia un vector $\vec{t_i} = (w_{i,1}, \ldots, w_{i,N})$ donde

$$w_{i,j} = \frac{(0.5 + 0.5\frac{f_{i,j}}{\max_j(f_{i,j})})itf_j}{\sqrt{\sum_{l=1}^{N}(0.5 + 0.5\frac{f_{i,j}}{\max_j(f_{i,j})})^2 itf_j^2}}$$

La correlación entre dos términos k_a y k_b se mide como en el modelo vectorial

$$c_{a,b} = \vec{t_a} \cdot \vec{t_b}.$$

Slide 18

Es similar al caso de clusters de asociación

Para cada par de términos se calcula el valor de correlación.

El cluster se define a partir the un umbral.

Expansión de la interrogación

El proceso sigue tres fases

- Representar la interrogación en el espacio de conceptos.

$$\vec{q} = \sum_{t_i \in q} w_{i,q} \cdot \vec{t_i}.$$

- Calcular la similaridad entre todos los términos correlacionados con los de la interrogación y ella misma.

Slide 19

$$sim(q, t_a) = \vec{q} \cdot \vec{t_a}.$$

- Expandir la interrogación añadiendo los r términos más puntuados. A los nuevos términos se les asigna componente

$$w_{a,q'} = \frac{sim(q, t_a)}{\sum_{t_b \in q} w_{b,q}}.$$

Categorización de documentos

Complete link algorithm

- Inicialmente, cada documento es un cluster.
- Se calcula la similaridad entre todos los pares de clusters
- Determinar el par de clusters con la similaridad inter-clusters máxima.
- Fusionar los clusters obtenidos en el paso anterior
- Verificar el criterio de parada, si no se alcanza volver al segundo paso.
- Devolver la jerarquía de clusters obtenida.

Slide 20

La similaridad inter-cluster se calcula como el mínimo de la similaridad de pares de documentos uno de cada cluster.

La jerarquía se representa como un árbol (o bosque) binario en cuyas hojas están los documentos. Los nodos internos representan clusters, formados por las hojas del subárbol asociado. En los nodos internos se guarda la similaridad inter-cluster, de los dos hijos.

Dada una jerarquía de documentos se obtiene un tesauro como sigue

- Obtener del usuario: threshold class (TC), número de documentos (NDC) y valor mínimo de inverse document frequency ($MIDF$).
- Usar el parámetro TC para descartar todos los nodos con similaridad menor que TC.
- Usar NDC para descartar todos los nodos con más de NDC hojas.
- Usar $MIDF$ para eliminar de cada cluster los términos con idf mayor que $MIDF$.

Slide 21

Al acabar hemos obtenido una cierto números de clases de términos que constituyen el tesauro final.

Desde un punto de vista estadístico se garantiza que sólo términos con baja frecuencia apareceran en las clases.

Expansión de la interrogación

Partimos de una clasificación de términos.

Para cada clase C, se calcula el peso medio de términos

$$wt_c = \frac{\sum_{t \in q} w(t, C)}{|C|},$$

donde $w(t, C)$ es el peso asociado al término t en la clase C.

Slide 22

Finalmente se incorporan a la interrogación los términos de las clases con mayor valoración.