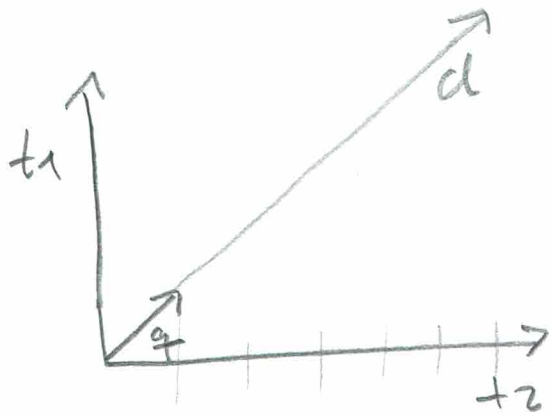
RI - CHAPTER Retrieval Models



$q = t_1 t_2$

$d = t_1 t_1 t_1 t_1 t_1 t_1 t_2 t_2 t_2 t_2 t_2 t_2$

$\vec{q}$ y $\vec{d}$ hablan de lo mismo pero de manera verbosa. Distancia Euclide de q y d es grande, intuitivamente no es una buena medida para lo que pretendemos

$\vec{d} \cdot \vec{q} = |d| \cdot |q| \cdot \cos(d, q)$ → Espacio vectorial con t términos

$$\cos(d,q) = \frac{d \cdot q}{|d| \cdot |q|} = \frac{\sum_{j=1}^{t} d_j \cdot q_j}{\sqrt{\sum_{j=1}^{t} d_j^2 \sum_{j=1}^{t} q_j^2}}$$ rank

rank

$$\frac{\sum_{j=1}^{t} d_j \cdot q_j}{\sqrt{\sum_{j=1}^{t} d_j^2}} = \frac{d \cdot q}{|d|}$$

- term frequency para el término k en el documento i

$$tf_{ik}$$

- nº ocurrencias del término k en el documento i

$$f_{ik} \quad \underline{\text{raw tf}}$$

- Heurísticas

tf normalizado $\quad tf_{ik} = \dfrac{f_{ik}}{\sum_{j=1}^{t} f_{ij}}$

Suma de las ocurrencias de todos los términos del documento i

tf logarítmico $\quad tf_{ik} = \begin{cases} 0 & \text{si } f_{ik}=0 \\ 1+\log f_{ik} & \text{si } f_{ik}\geq 1 \end{cases}$

- Idf. inverse document frequency para el término k. Idf logarítmico:

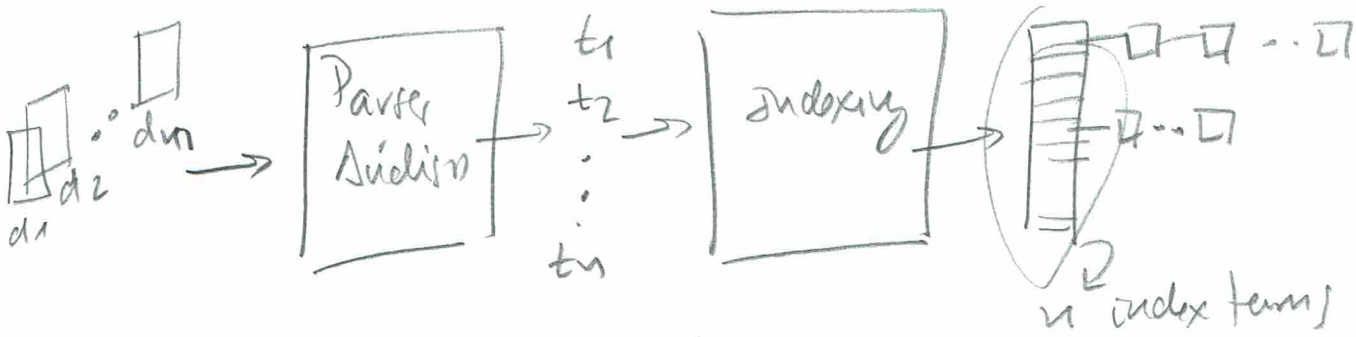
$$idf_k = \log \frac{N}{n_k}$$

N, num. docs de la colección
n_k, num docs en los que aparece el término k

Si $n_k = N \longrightarrow idf_k = 0$
Si $n_k = 1 \longrightarrow idf_k = \log N$

VSM Vector Space Model

m docs
n terms



Docs y Queris son vectores de dimension n

$d = \langle d_1, d_2 \text{ ——— } d_n \rangle$

$q = \langle q_1, q_2, \text{ ——— } q_n \rangle$

$$\cos(q,d) = \frac{\sum_{i=1}^{n} q_i d_i}{|q|\,|d|} = \frac{\sum_{i=1}^{n} q_i d_i}{\sqrt{\sum_i q_i^2}\ \sqrt{\sum_{i=1}^{n} d_i^2}}$$ rank

rank

$$\frac{\sum_{i=1}^{n} q_i d_i}{\sqrt{\sum_i d_i^2}} = \sum_i q_i \frac{d_i}{\sqrt{\sum_i d_i^2}}$$

componente vector normalizado

$$\frac{\sum_{i=1}^{n} q_i d_i}{\sqrt{\sum_i f_i^2}\ \sqrt{\sum_{i=1}^{n} d_i^2}}$$

→ componentes de $\vec{q}$ y $\vec{d}$ normalizados

Esquema pesado

q_i, d_i → binario 1/0

→ tf raw tf del termino en → d → q

→ tf normalizado en el doc o en la query

$$\frac{tf_i}{\sum_{k1}^{J} tf_k}$$

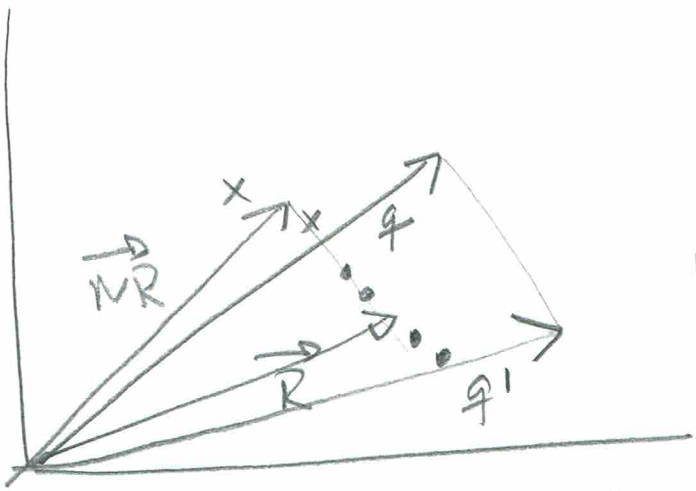
→ tf log $(1 + \log tf)$

→ $idf = \log \frac{N}{n_i}$

$\boxed{tf \cdot idf}$ → para cada q_i, d_i

Relevance Feedback

• docs relevants
× docs no relevants



$$\vec{q}' = \vec{q} + \vec{R} - \vec{NR}$$

$\vec{R}$ = centroide de los docs relevantes

$\vec{NR}$ = centroide de los docs no relevantes

Con parámetros α, β, γ

$$\vec{q}' = \alpha \vec{q} + \beta \vec{R} - \gamma \vec{NR}$$

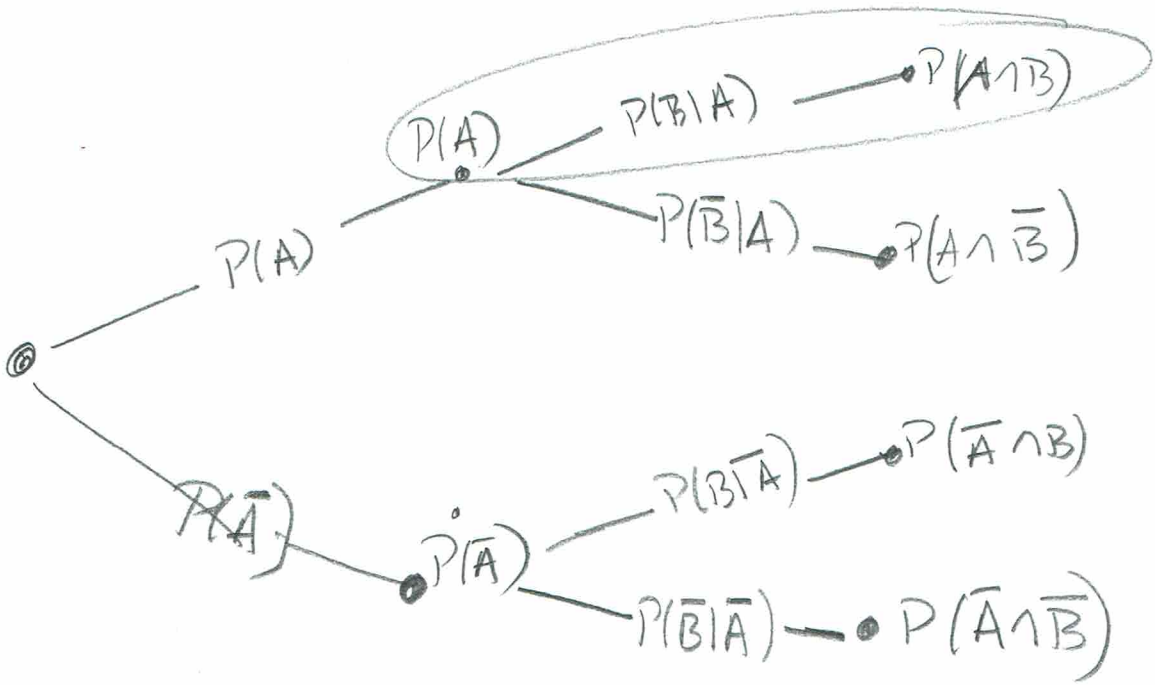
En la práctica puede llevar a:

1. Componentes negativas en $\vec{q}'$
2. $\vec{q}'$ con muchos términos no nulos

soluciones

1. Ignorarlos
2. Seleccionar terminos con mayor $tf \cdot idf$

Prob. condicional



$$P(A) \cdot P(B|A) = P(A \wedge B)$$

$$P(B|A) = \frac{P(A \wedge B)}{P(A)}$$

Queremos estimar $P(d|q)$, para todos los docs d dada una query q, o sea, queremos estimar

$P(d_1|q)$, $P(d_2|q)$, $P(d_3|q)$, etc

y producir ranking por orden decreciente de $P(d|q)$

$$P(d|q) = \frac{P(d \cap q)}{P(q)}$$

$$P(q|d) = \frac{P(q \cap d)}{P(d)}$$

$$\rightarrow \boxed{P(d|q) = \frac{P(q|d) \cdot P(d)}{P(q)}} \overset{\text{rank}}{=}$$

$$\overset{\text{rank}}{=} P(q|d) \cdot P(d)$$

→ QUERY LIKELIHOOD

Si q tiene q_i terms, suponiendo los q_i|D son events independientes, modelo multinomial

$$\boxed{P(q|d) = \prod_{i=1}^{n} P(q_i|D)}$$

q tiene n query terms

Si a y b son independientes

$$P(a \wedge b) \equiv P(a,b) = P(a) \cdot P(b)$$

$$P(a|b) = \frac{P(a \wedge b)}{P(b)} = \frac{P(a) \cdot P(b)}{P(b)} \quad \text{(si son independientes)}$$

$$P(a|b) = P(a) \quad , \text{ y a la } \quad P(b|a) = P(b)$$

lo mismo para probs condicionales. Si a y b son condicionalmente independientes de C:

$$P(a,b|C) = P(a|C) \cdot P(b|C)$$

Esto fue lo que supusimos al estimar el QL

$$P(q|d) = P(q_1, q_2, \ldots, q_n|d) = \prod_{i=1}^{n} P(q_i|d)$$

q tiene n términos. Estos pueden ser t términos distintos en q, que cada uno aparece $tf_{t,q}$ veces

$$P(q|d) = \prod_{i=1}^{n} P(q_i|d) = \prod_{\substack{\text{terms} \\ \text{distintos} \\ t \text{ en } q}} P(t|d)^{tf_{t,q}}$$

QUERY LIKELIHOOD: MODELO MULTINOMIAL (omitiendo factor cte multinomial).

Distribución binomial con prob. p, params n, K

Una variable aleatoria X se rige por esta distribución si

$$P(K; n, p) = P_r(X = K) = \binom{n}{K} p^K (1-p)^{n-K}$$

Representa la probabilidad de que ocurran K éxitos en n intentos (n sucesos de Bernoulli, i.e., sucesos binarios independientes con prob. p)

Para la probabilidad de éxito en un suceso Bernoulli, por tanto:

K éxitos ocurren con prob. p^K

$n-K$ fallos " " $(1-p)^{n-K}$

Además los K éxitos pueden ocurrir en cualquier lugar de los n intentos, lo que implica que hay $\binom{n}{K} = \frac{n!}{n!\,(n-K)!}$ formas distintas de distribuir los K éxitos.

Ej: Prob. de sacar 3 unos en 10 intentos con dado de 6 caras

$$P\left(3; 10, \frac{1}{6}\right) = \binom{10}{3}\left(\frac{1}{6}\right)^3\left(\frac{5}{6}\right)^7 = 0{,}155$$

La distribución multinomial es una generalización de la binomial, donde cada intento lleva a K distintas categorías de éxito

Distribución multinomial

Se extraen n bolas de una urna con bolas de K diferentes colores, reemplazando la bola extraída después de cada extracción

$X_i \rightarrow$ var. para el número de bolas extraídas del color i $(i = 1, \dots K)$

$p_i \rightarrow$ prob. de la bola extraída sea del color i

Prob. de extraer x_i bolas de cada color i para los K diferentes colores en n extracciones con reemplazo:

$$f(x_1, \dots, x_K; n, p_1, \dots p_K) = P(X_1 = x_1, X_2 = x_2, \text{and}, X_K = x_K)$$

$$= \begin{cases} \dfrac{n!}{x_1! x_2! \cdots x_K!} p_1^{x_1} \cdot p_2^{x_2} \cdots p_K^{x_K} & \text{si } \sum_{i=1}^{K} x_i = n \\ 0 \text{ otherwise} \end{cases}$$

$\rightarrow$ factor de la distribución multinomial

Analogía con QL

urna $\rightarrow$ documento

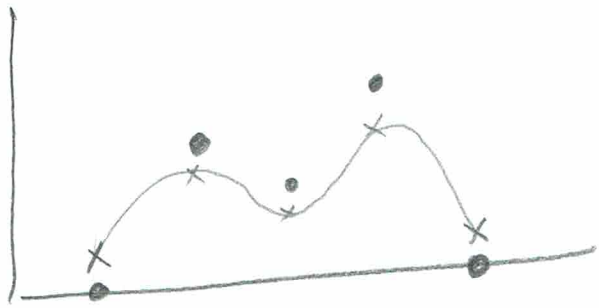
n, num bolas extraídas $\rightarrow$ n query terms totales

K diferentes colores $\rightarrow$ K diferentes palabras, (K palabras únicas)

$$p_i = \frac{\text{nº bolas color } i}{\text{nº bolas totales urna}} \longrightarrow p(w|d) = \frac{\text{nº ocurrencias } w \text{ en } d}{\text{nº ocurrencias total de palabras en } d}$$

Smoothing

• Probs discretas originales
× Probs suavizadas



$$P(q_i|C) = \frac{\text{nº ocurrencias palabra } q_i \text{ en } C}{\text{nº ocurrencias totales en } C}$$

$P(Q|D) \rightsquigarrow \log P(Q|D)$

log es transformación monótona

Si $A > B \Rightarrow \log(A) > \log(B) \Rightarrow$

$\Rightarrow$ preserva el ranking

Sede where is tf.idf weight?

$i = 1, \dots, n$ todos los query terms

$i : f_{q_i,D} > 0$ terminos de la query que ocurren en el doc.

$i : f_{q_i,D} = 0$ " " " no " " "

De la 2ª a 3ª linea, hacemos $+ \sum_{i:f_{q_i,D}>0} \log\left(\frac{\lambda c_{q_i}}{|C|}\right)$

De la 3ª a 4ª linea, $\left(\sum_{i=1}^{n} \log\left(\frac{\lambda c_{q_i}}{|C|}\right)\right)$ no afecta al ranking, igual para todos los docs.

Se llega a algo parecido a tf x idf pero con una derivación formal

Dirichlet Smoothing

$$\alpha_D = \frac{\mu}{|D| + \mu}$$

$$P(q_i|D) = (1-\alpha_D)\frac{f_{q_i,D}}{|D|} + \alpha_D \frac{c_{q_i}}{|C|} =$$

$$= \frac{|D|}{|D|+\mu} \cdot \frac{f_{q_i,D}}{|D|} + \frac{\mu}{|D|+\mu} \cdot \frac{c_{q_i}}{|C|} =$$

$$= \frac{|C| f_{q_i,D} + \mu c_{q_i}}{(|D|+\mu)\cdot|C|} = \frac{f_{q_i,D} + \mu \frac{c_{q_i}}{|C|}}{|D|+\mu}$$

$$P(Q|D) = \prod_{i=1}^{n} P(q_i|D) \quad \leftrightsquigarrow \quad QL$$

$$\log P(Q|D) = \sum_{i=1}^{n} \log \frac{f_{q_i,D} + \mu \frac{c_{q_i}}{|C|}}{|D| + \mu}$$

QUERY LIKELIHOOD MODEL

$$P(Q|D) = \prod_{qi} P(qi|D)$$ qi query term

DOCUMENT LIKELIHOOD MODEL

$$P(D|Q) = \prod_{di} P(di|Q)$$ di document words

- dificultad de comparar Docs de muy distinta longitud
- $|\{di\}| \gg |\{qi\}|$

aunque muchos di no están en Q, si muestras hay que corregirlo igual $P(di|Q)$

RELEVANCE MODEL

$P(w|R)$

KL-DIVERGENCE (Kullback-Leibler)

Mide distancias entre distribuciones de prob.

Si $Q(x)=0$ y $P(x)\neq 0$, no está definido, por lo que si $Q(x)=0$, la contribución a KL-DIV es 0

$$KL(P||Q) = \sum_x P(x) \log \left(\frac{P(x)}{Q(x)} \right)$$

Siempre no negativa

$P(w|R) \rightsquigarrow P$

$P(w|D) \rightsquigarrow Q$, una aproximación de P

valores pequeños de KL → menor distancia → mejor

similitud → $-KL$ nos da directamente el ranking por similitud

$$-KL(P(w|R)||P(w|D)) = -\sum_{w\in V} P(w|R) \log \frac{P(w|R)}{P(w|D)} =$$

$$= \sum_{w\in V} P(w|R) \log P(w|D) - \sum_{w\in V} P(w|R) \log P(w|R)$$

igual para todos los docs ⟷ no afecta ranking

Relación KL-Div y QL

QL $P(Q|D) = \prod_{i=1}^{n} P(q_i|D) \overset{\text{rank}}{=} \sum_{i=1}^{n} \log P(q_i|D)$

Si estimo $P(w|R)$ con $\frac{f_{w,Q}}{|Q|}$ (MLE)

$$KL\,Div = \sum_{w \in V} P(w|R) \log P(w|D) =$$

$$= \sum_{w \in V} \left(\frac{f_{w,Q}}{|Q|} \log P(w|D) \right)$$

$\rightarrow$ 0 para las w que no están en la query $\rightarrow$ producen el mismo ranking

Estimación de RM

Definimos $P(w|R) \simeq P(w|q_1, q_2, \dots q_n)$

Por definición de probabilidad condicional:

$$P(w|R) \simeq \frac{P(w, q_1, q_2, \dots q_n)}{P(q_1, q_2, \dots q_n)}$$

Estimación de RM

$\{D \in C\}$ partición espacio eventos

ley prob. total

• Ley Prob. Total.

$$P(w, q_1 \cdots q_n) = \sum_{D \in C} P(D)\, P(w, q_1 \cdots q_n \mid D)$$

• Ahora suponemos que w y las query words q_i son condicionalmente independientes de D, i.e, una vez elegido D, w y los q_i 'are i.i.d' (identically and independently) sampled'

$$P(w, q_1 \cdots q_n \mid D) = P(w \mid D) \cdot \prod_{i=1}^{n} P(q_i \mid D)$$

$$P(w, q_1, \cdots q_n) = \sum_{D \in C} P(D)\, P(w \mid D) \prod_{i=1}^{n} P(q_i \mid D)$$

$$P(w \mid R) \triangleq \frac{P(w, q_1 \cdots q_n)}{P(q_1 \cdots q_n)} = \frac{P(w, q_1 \cdots q_n)}{\sum_{w \in V} P(w, q_1 \cdots q_n)}$$

↙ ↙

cte de normalización

Pseudo Feedback Alg

En 4. A efectos prácticos seleccionamos las palabras con $P(w|R)$ más alto. y el ranking que produce 4 es el mismo que lanzando una query con QL con estos top w query terms

RM3 $P'(w|R) = \lambda P(w|R) + (1-\lambda) P(w|Q)$

↳ Interpola el RM con la query original.

Eval de PRF Algs, además de MAP, NDCG, ... evalúa también Robustness Index

$$R_I = \frac{n_+ - n_-}{|Q|}$$

n_+ nº queries mejoradas por PRF Alg

n_- " dañadas por " "

Lime PRF with Linear Regression Methods
ACM SAC 2018

Lime > Relevance Models

Dar referencia