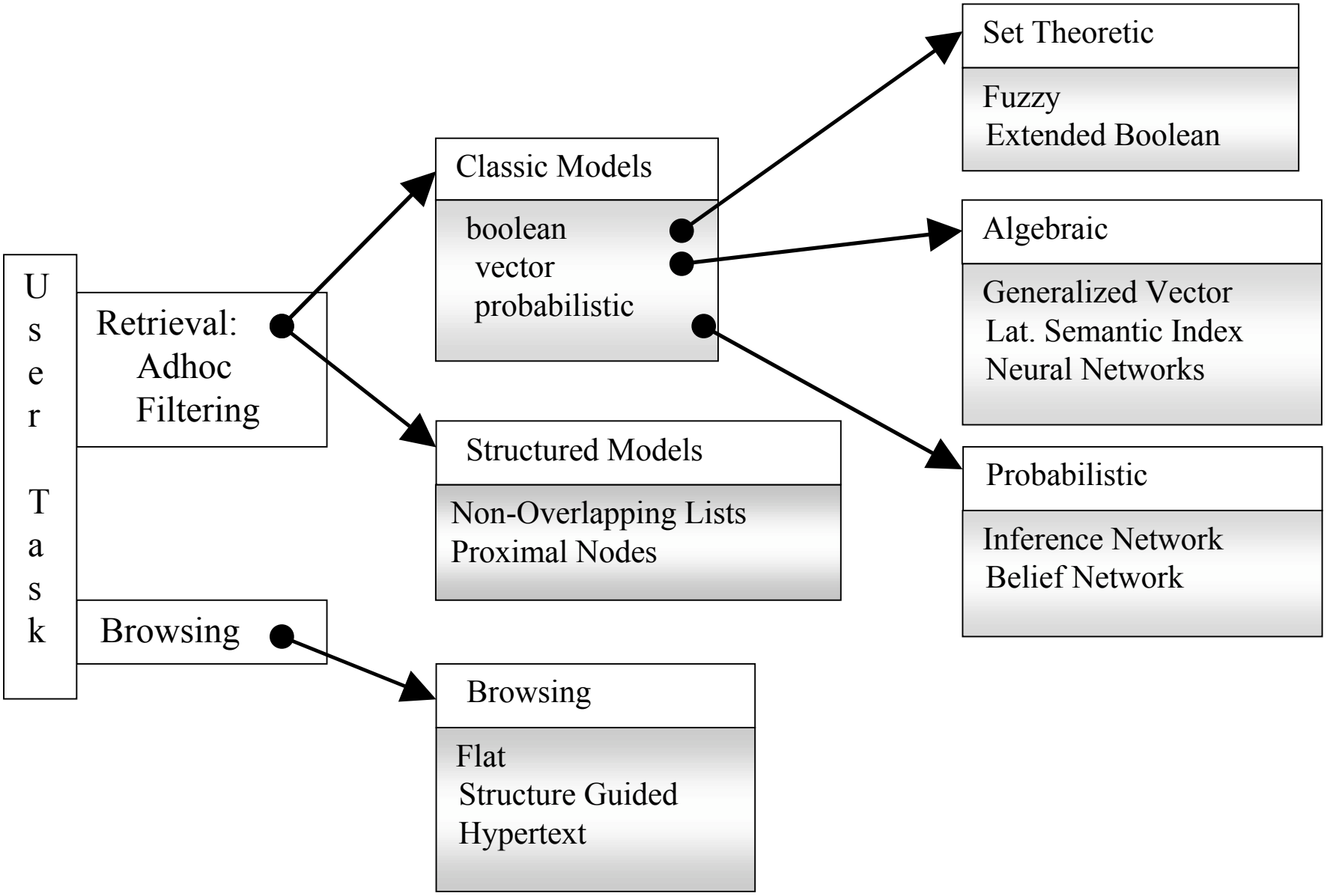
# Modelos Clásicos de Recuperación

M. Andrea Rodríguez Tastets
DIIC - Universidad de Concepción
http://www.inf.udec.cl/~andrea

# Introducción

- Un *ranking* es un orden de documentos recuperados que "ojalá" refleje la relevancia respecto a una consulta.
- Un ranking está basado en premisas fundamentales respecto a la noción de relevancia, tales como:
  - conjunto común de términos
  - compartir términos con pesos
  - similitud de relevancia
- Cada premisa en particular lleva a un modelo de Recuperación de Información

# Modelos


User Task
Retrieval: Adhoc Filtering
Browsing
Classic Models
boolean
vector
probabilistic
Structured Models
Non-Overlapping Lists
Proximal Nodes
Browsing
Flat
Structure Guided
Hypertext
Set Theoretic
Fuzzy
Extended Boolean
Algebraic
Generalized Vector
Lat. Semantic Index
Neural Networks
Probabilistic
Inference Network
Belief Network

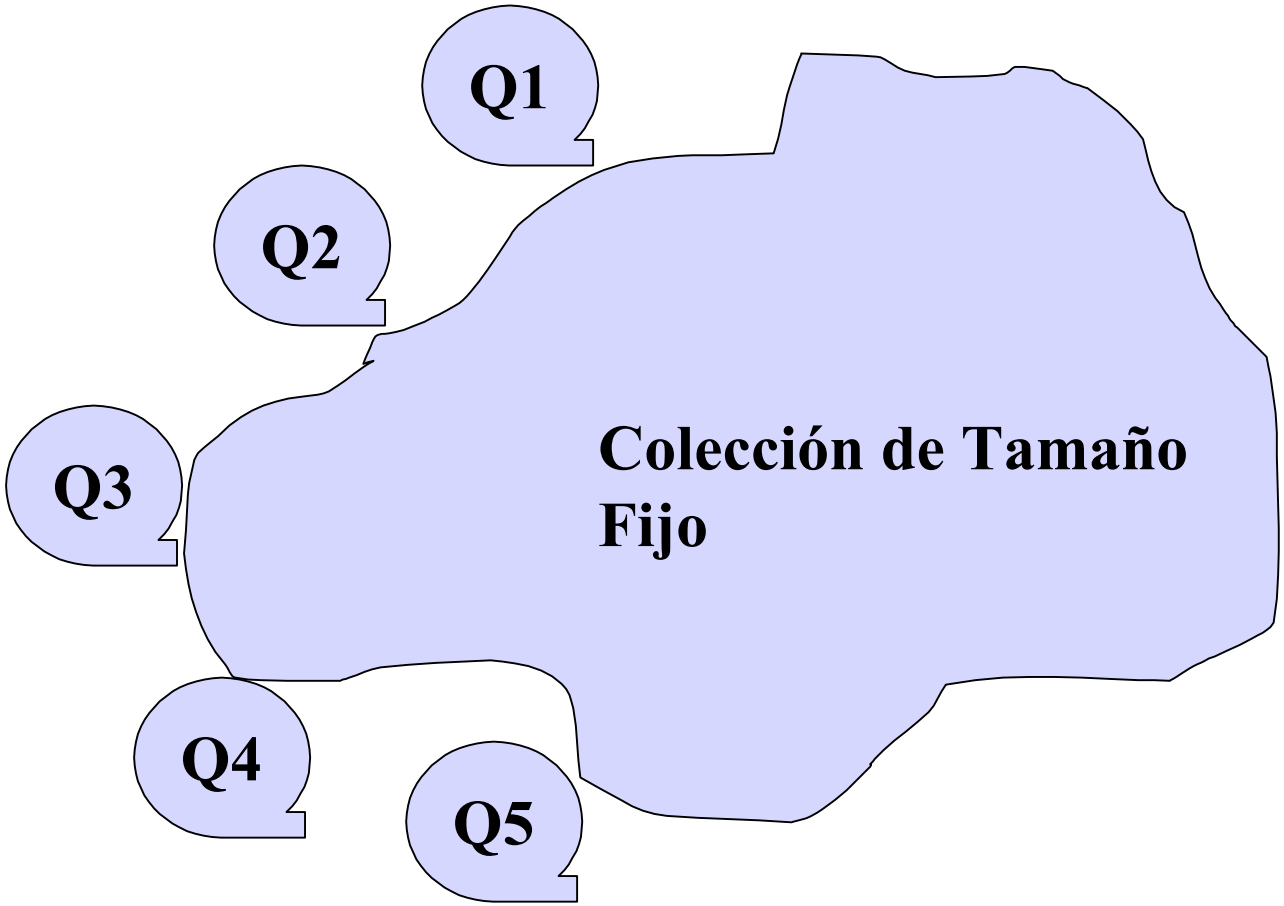
# Modelos IR

- Vista lógica versus tarea de recuperación

| | **Index Terms** | **Full Text** | **Full Text + Structure** |
|---|---|---|---|
| **Retrieval** | Classic<br>Set Theoretic<br>Algebraic<br>Probabilistic | Classic<br>Set Theoretic<br>Algebraic<br>Probabilistic | Structured |
| **Browsing** | Flat | Flat<br>Hypertext | Structure Guided<br>Hypertext |

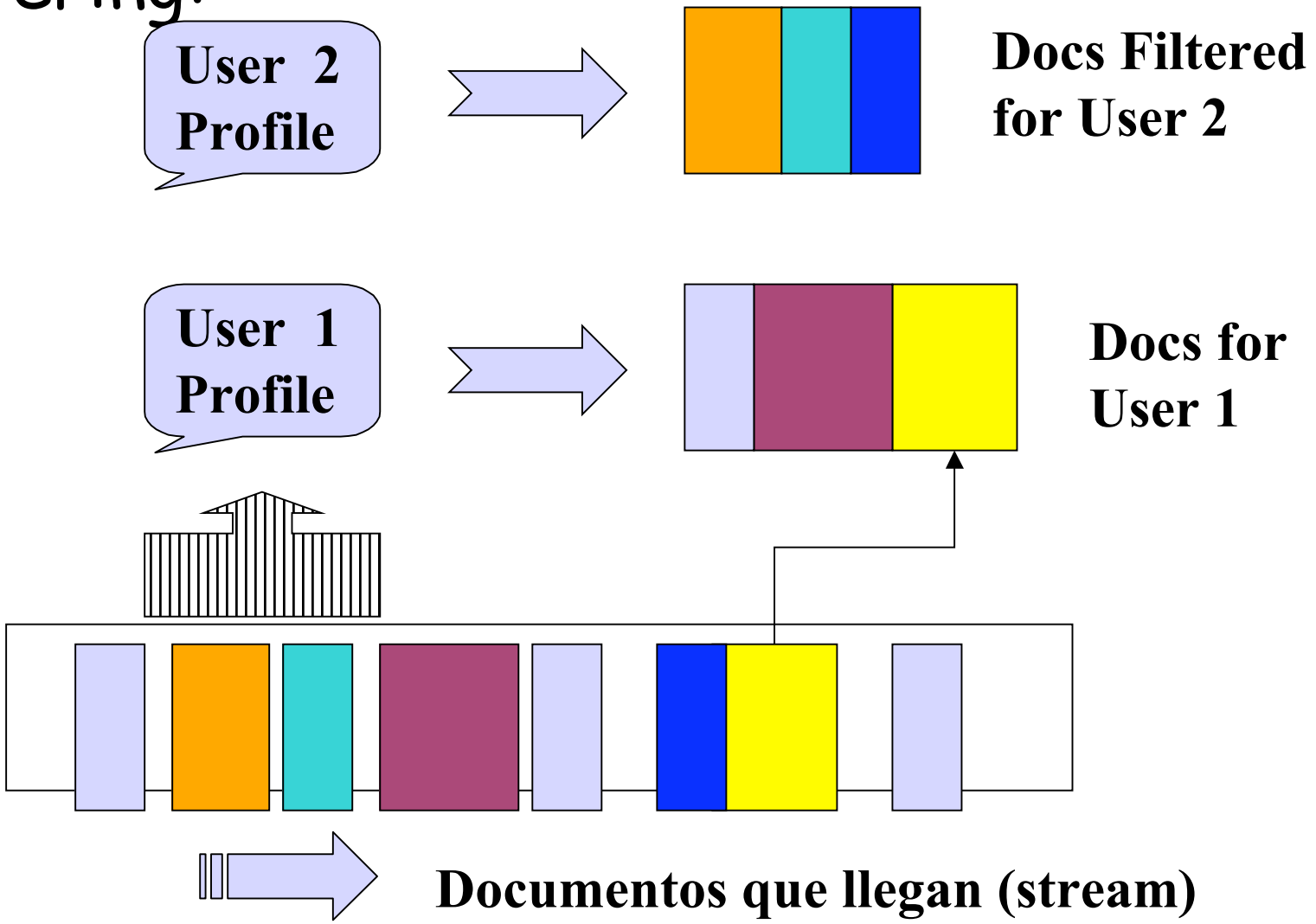
# Recuperación: Ad Hoc x Filtering

- Ad-hoc



# Recuperación: Ad Hoc x Filtering

- Filtering:



# Modelos Clásicos: Conceptos

- Cada documento es representado por un conjunto de palabras representativas o keywords.
- Un índice es una palabra en un documento que es útil para identificar el contenido del documento.
- Usualmente estos índices son sustantivos porque ellos tienen significado por sí solos.
- Sin embargo, máquinas de búsqueda asumen que todas las palabras son términos índices (representación de full text)

# Modelos Clásicos: Conceptos

- No todos los términos son igualmente útiles para representar un documento: términos menos frecuentes permiten identificar un conjunto más selecto de documentos.
- La *importancia* de los términos índices es representada por pesos asociados a ellos. Sean
  - *ki* un término índice
  - *dj* un documento
  - *wij* es un peso asociado con *(ki,dj)*
- El peso *wij* cuantifica la importancia del término índice para describir el contenido de documentos

# Modelos Clásicos: Conceptos

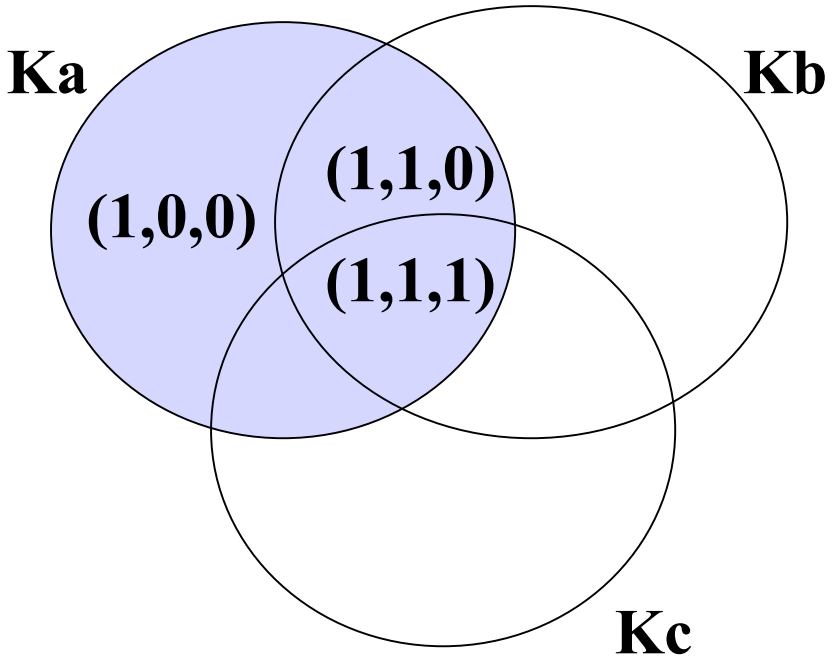
- *Ki* es un término índice
- *dj* es un documento
- *t* es el total de términos índices
- *K = (k1, k2, …, kt)* es el conjunto de términos índices
- *wij >= 0* es un peso asociado con *(ki,dj)*
- *wij = 0* indica que el término *i* no pertenece a un documento *j*
- *vec(dj) = (w1j, w2j, …, wtj)* es un vestor de pesos asociado a documento *dj*
- *gi(vec(dj)) = wij* es una función la cual retorna el peso asociado al par *(ki,dj)*

# Modelo Boolean

- Modelo Simple basado en teoría de conjunto
- Consultas expresadas como expresiones booleanas
  - precisa especificación
  - formalismo claro
  - *q = ka (kb ∀ kc)*
- Términos están o no presentes. Así, *wij {0,1}*
- Considere
  - *q = ka (kb ∀ kc)*
  - *vec($q_{dnf}$) = (1,1,1) ∀ (1,1,0) ∀ (1,0,0)*
  - *vec($q_{cc}$) = (1,1,0)* es un componente conjuntivo

# Modelo Boolean

- *q = ka (kb ∀ kc)*




- *sim(q,dj) = 1 if ∃ vec($q_{cc}$) | (vec($q_{cc}$) vec($q_{dnf}$))*
  *( ki, gi(vec(dj)) = gi(vec($q_{cc}$)))*

  *0 otherwise*

# Desventajas Modelo Boolean

- Recuperación basada en un criterio binario con ninguna noción de correspondencia parcial
- No ranking es producido
- La consulta debe ser traducidas a una expresión Boolean la cual no es cómoda para usarios
- Las consultas son muchas veces demasiado simples
- Consecuentemente, el modelo puede retornan muchos o pocos documentos en respuesta a una consulta del usuario.

# Desventajas Modelo Boolean

- Recuperación basada en un criterio binario con ninguna noción de correspondencia parcial
- No ranking es producido
- La consulta debe ser traducidas a una expresión Boolean la cual no es cómoda para usarios
- Las consultas son muchas veces demasiado simples
- Consecuentemente, el modelo puede retornan muchos o pocos documentos en respuesta a una consulta del usuario.

# Ejemplo

| | k1 | k2 | k3 | k4 | k5 | k6 |
|---|---|---|---|---|---|---|
| d1 | 5 | 0 | 2 | 1 | 2 | 1 |
| d2 | 2 | 5 | 2 | 0 | 0 | 3 |
| d3 | 0 | 2 | 1 | 5 | 0 | 0 |
| d4 | 3 | 0 | 1 | 5 | 1 | 0 |
| d5 | 0 | 1 | 1 | 0 | 2 | 0 |

$q$: k1 ( k3 $\forall$ ¬k6)

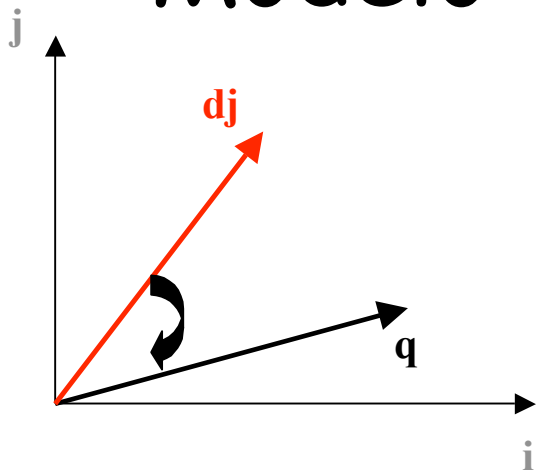
# Modelo Vectorial

- El uso de pesos binarios es muy limitado
- Considera correspondencia parcial
- El peso de los términos es usado para calcular *el grado de similitud* entre consulta y cada documento.
- Conjunto ordenado (ranked) de documentos en base a mejor correspondencia con la consulta

# Modelo Vectorial

- Define:
  - *wij > 0* cuando *ki dj*
  - *wiq >= 0* asociado con el par *(ki,q)*
  - *vec(dj) = (w1j, w2j, ..., wtj)*
    *vec(q) = (w1q, w2q, ..., wtq)*
  - Por cada término *ki* se asocia un vector unitario *vec(i)*
  - Los vectores unitarios *vec(i)* y *vec(j)* son asumidos ortogonales (i.e., los índices son considerados independientes en un documento)
- El vector t-unario *vec(i)* forma una base ortogonal para el espacio t-dimensional
- En este espacio, consultas y documentos son representados como vectores de pesos

# Modelo Vectorial




- *Sim(q,dj) = cos( )*
  *= [vec(dj) vec(q)] / |vec(dj)| |vec(q)|*
  *= [ wij wiq] / $wij^2$ $wiq^2$*
- Ya que *wij >= 0* y *wiq >= 0, 0<= sim(q,dj) <=1*
- Un documento es recuperado aunque corresponda parcialmente a una consulta.

# Modelo Vectorial

*Sim(q,dj) = [vec(dj) vec(q)] / |vec(dj)| |vec(q)|*

- Como calcular los pesos *wij* y *wiq*?
- Un buen peso debe tomar en cuenta dos efectos:
  - cuantificación de intra-documentos (similitud)

    *factor tf*, la frecuencia del término en un documento
  - cuantificación inter-documentos (disi-militud)

    *factor idf*, la frecuencia inversa en los documentos

    *wij = tf(i,j) * idf(i)*

# Modelo Vectorial

- Sean,
  - $N$ el número total de documentos
  - $ni$ el número de documentos que contienen $ki$
  - $freq(i,j)$ frecuencia de $ki$ en $dj$
- Un factor normalizado de $tf$ está dado por
  - $f(i,j) = freq(i,j) / max(freq(l,j))$
  - donde el máximo es computado sobre todos los términos que ocurren en el documento $dj$
- El factor $idf$ es calculado por
  - $idf(i) = log (N/ni)$
  - el $log$ es usado para hacer los valores de $tf$ y $idf$ comparables. Puede ser interpretado como la cantidad de información asociada con el término $ki$.

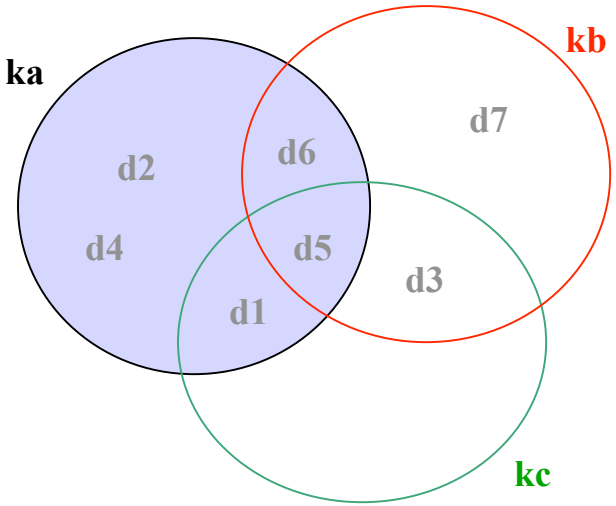
# Modelo Vectorial

- El esquema best term-weighting usa los pesos definidos por
  - *wij = f(i,j) * log(N/ni)*
  - esta estrategia es conocida como esquema de pesos *tf-idf*
- Para los términos de la consulta, se sugiere:
  - *wiq = (0.5 + [0.5 * freq(i,q) / max(freq(l,q)]) * log(N/ni)*
- El modelo vectorial con pesos *tf-idf* es una buena estrategia de ranking para una colección general
- El modelo vectorial es usualmente tan bueno como las alternativas de ranking conocidas. Es también simple y rápido de calcular.

# Modelo Vectorial

- Vectajas:
  - Peso de términos mejora lacalidad del conjunto respuesta
  - Correspondencia parcial permite la recuperación de documentos que aproxima las condiciones de la consulta.
  - Formula del ranking cosine ordena documentos deacuerdo al grado de similitud de la consulta
- Desventajas:
  - Asume independencia de los índices (algo aún incierto)

# Modelo Vectorial: Ejemplo I




$q = ka \quad (kb \ \forall \quad kc)$

# Modelo Probabilístico

- Objetivo: capturar el problema de IR usando un enfoque probabilístico
- Dada una consulta de un usuario, hay un conjunto respuesta ideal
- Consulta como especificación de las propiedades de este conjunto ideal de respuesta (clustering)
- Pero, cuáles son estas propiedades?
- Estima al comienzo cuáles pueden ser (e.i., una estimación inicial del conjunto respuesta ideal)
- Mejora por iteración

# Modelo Probabilístico

- Un conjunto inicial de documentos es recuperado de alguna manera
- El usuario inspecciona estos documentos buscando por los relevantes (en la práctica, sólo los top top 10-20 necesitan ser inspeccionados)
- Un sistema IR usa esta información para refinar la descripción del conjunto ideal
- Por repetición, se espera que el conjunto respuesta ideal vaya mejorando
- Siempre se debe pensar en que al comienzo se hace una estimación del conjunto ideal de respuesta
- La descripción del conjunto ideal es modelada en términos probabilísticos

# Principio del Ranking Probabilístico

- Dada una consulta de usuario *q* y un documento *dj*, el modelo probabilístico trata de estimar la probabilidad que el usuario encontrará el documento *dj* interesante (i.e., relevante). Este modelo asume que esta probabilidad de relevancia depende sólo de la consulta y del documento. El conjunto de respuesta ideal es denotado por *R* y debe maximizar la probabilidad de relevancia. Documentos en el conjunto *R* se dicen ser relevantes.
- Pero,
  - como determinar las probabilidades?
  - cuál es el espacio de muestreo?

# El Ranking

- El ranking probabilístico es calculado como:
  - *sim(q,dj) = P(R | vec(dj)) / P( R | vec(dj))*
- Definición
  - *wij {0,1}*
  - *P(R | vec(dj))* : probabilidad de que un documento dado es relevante
  - P( R | vec(dj)) : probabilidad de un documento no es relevante

# El Ranking

- sim(dj,q) = P(R | vec(dj)) / P( R | vec(dj))

  = $\frac{[P(vec(dj) \mid R) * P(R)]}{[P(vec(dj) \mid R) * P(R)]}$ , Por Bayes's

  ~ $\frac{P(vec(dj) \mid R)}{P(vec(dj) \mid R)}$ , P(R ), P( R)] para todos los docs.

- P(vec(dj) | R) : probabilidad de aleatorimente seleccionar el documento dj desde el conjunto R de documentos relevantes

# El Ranking

- $sim(dj,q) \sim \frac{P(vec(dj) \mid R)}{P(vec(dj) \mid \overline{R})}$

$$\sim \frac{[P(ki \mid R)][P(\overline{ki} \mid R)]}{[P(ki \mid \overline{R})][P(\overline{ki} \mid \overline{R})]}$$

- P(ki | R) : probabilidad que el término índice ki este presente en un documento aleatoriamente seleccionado desde el conjunto R de documentos relevantes

# El Ranking

- $$sim(dj,q) \sim \log \frac{[P(ki \mid R)]\,[P(kj \mid R)]}{[P(ki \mid \bar{R})]\,[P(kj \mid \bar{R})]}$$

$$\sim \sum wiq\ wij \left(\log \frac{P(ki \mid R)}{P(ki \mid R)} + \log \frac{P(ki \mid \bar{R})}{P(ki \mid \bar{R})}\right)$$

donde $P(\bar{ki} \mid R) = 1 - P(ki \mid R)$

$P(\bar{ki} \mid \bar{R}) = 1 - P(ki \mid \bar{R})$

# El Ranking Incial

- sim(dj,q) ~

$$\sim \quad wiq * wij * \left(\log \frac{P(ki \mid R)}{P(\overline{ki} \mid R)} + \log \frac{P(ki \mid \overline{R})}{P(\overline{ki} \mid \overline{R})}\right)$$

- Probabilidades $P(ki \mid R)$ y $P(ki \mid \overline{R})$ ?
- Estimaciones basedas en presunciones:
  - $P(ki \mid R) = 0.5$
  - $P(ki \mid \overline{R}) = \frac{ni}{N}$

    donde ni es el número de documentos que contienen ki
  - Use esta estimación inicial para recuperar un ranking inicial
  - Mejoras sobre el ranking inicial

# Mejorando el Ranking Inicial

- $sim(dj,q) \sim$
  $\sim wiq * wij * \left(\log \frac{P(ki \mid R)}{P(\overline{ki} \mid R)} + \log \frac{P(ki \mid \overline{R})}{P(\overline{ki} \mid \overline{R})}\right)$
- Sean
  - V : conjunto de documentos inicialmente reecuperados
  - Vi : subconjunto de documentos recuperados que contienen ki
- Re-evaluar estimaciones :
  - $P(ki \mid R) = \frac{Vi}{V}$
  - $P(ki \mid \overline{R}) = \frac{ni - Vi}{N - V}$
- Repetir recursivamente