Modelli di IR

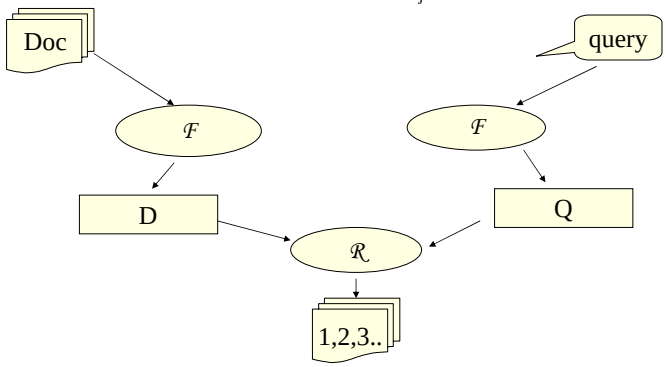
Modello di IR: definizione generale

- Un modello di IR caratterizza formalmente il processo di IR:
- *Def:* Un modello di IR è una quadrupla $[\mathbf{D},\mathbf{Q},\mathcal{F},\mathcal{R}(q_i,d_j)]$ dove:
  - **D** è un insieme di viste logiche, o *rappresentazioni* dei documenti nella collezione
  - **Q** è un insieme di viste logiche, o *rappresentazioni* dei bisogni informativi dell'utente, dette query
  - $\mathcal{F}$ è uno *schema* per modellare le rappresentazioni dei documenti, le query, e le inter-relazioni fra query e documenti
  - $\mathcal{R}(q_i,d_j)$ è una funzione di rilevanza, o *ranking,* che definisce un ordine fra i documenti, in relazione alla query q_i:

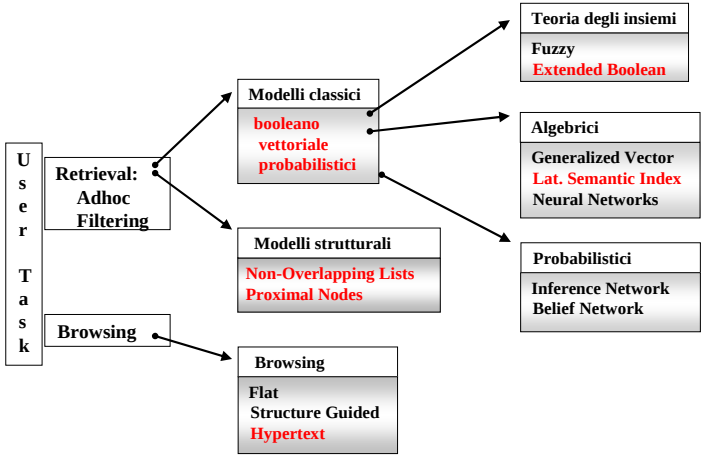
$$\mathcal{R}: \mathbf{Q}\times\mathbf{D}\rightarrow\Re$$

Modello di IR: definizione generale

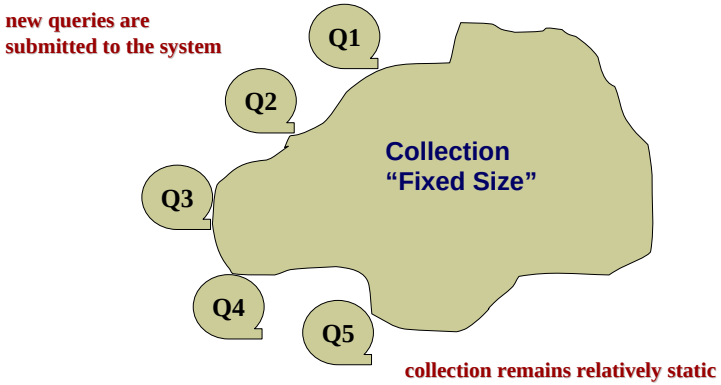
$[\mathbf{D},\mathbf{Q},\mathcal{F},\mathcal{R}(q_i,d_j)]$



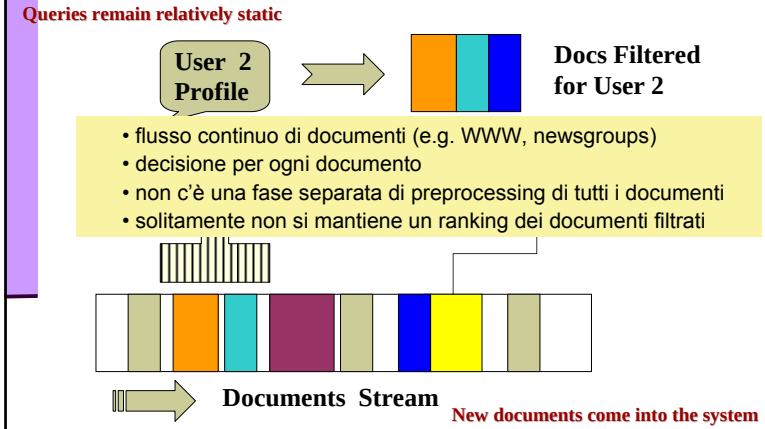
Una tassonomia dei modelli di IR



Ad Hoc Retrieval



Filtering



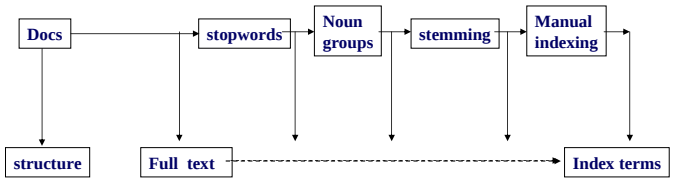
Filtering: Profilo utente

- Profilo utente
  - Rappresenta le preferenze dell’utente
  - Viene confrontato con i nuovi documenti per stabilire quali sono rilevanti per l’utente
- Due approcci
  - **Static user profile**
    - Insieme di keywords fornite dall’utente
    - Poco efficace
    - Non sempre efficiente
  - **Dynamic user profile**
    - Modificato continuamente
    - Si inizializza il profilo con keywords fornite dall’utente
    - Il profilo viene aggiornato in base a relevance feedback

Modelli Classici (non strutturali)

- Sia i documenti che le interrogazioni sono rappresentati in base a termini indice in essi contenuti
  - Un termine indice è costituito da una keyword (parola chiave) o da un gruppo di keyword, e rappresenta un concetto che caratterizza il contenuto informativo del documento
- Vantaggi
  - Semplice
  - E’ naturale esprimere la semanticha dei documenti e dei bisogni informativi rispetto ad un insieme di termini indoce
- Pesatura dei termini indice:
  - L’importanza di un indice è rappresentata da un peso ad esso associato
  - Non tutti i termini che compaiono in un documento sono egualmente rappresentativi del suo contenuto informativo
    - di solito i termini troppo frequenti non sono buoni candidati

Operazioni tipiche utilizzate per estrarre i termini indice



… le vedremo in dettaglio successivamente

Modelli classici di IR: Collezione di documenti

- Una collezione di *n* documenti è rappresentata come una matrice documenti-termini
  - Una cella della matrice corrisponde al peso del termine nel documento

$$\begin{array}{c|cccc} & T_1 & T_2 & \ldots & T_t \\ D_1 & w_{11} & w_{21} & \ldots & w_{t1} \\ D_2 & w_{12} & w_{22} & \ldots & w_{t2} \\ \vdots & \vdots & \vdots & & \vdots \\ \vdots & \vdots & \vdots & & \vdots \\ D_n & w_{1n} & w_{2n} & \ldots & w_{tn} \end{array}$$

- Celle con valore 0
  - il termine non è significativo per il documento o, semplicemente, che non è contenuto nel documento

Modelli Classici: notazione

- Siano
  - k_i il termine indice i-esimo
  - d_j il documento j-esimo
- w_{ij}: il peso associato a k_i nel documento d_j
  - quantifica l'importanza dell'indice k_i per descrivere il contenuto informativo del documento d_j
  - $w_{ij} = 0$ indica che k_i non compare in d_j
- $vec(d_j) = (w_{1j}, w_{2j}, \ldots, w_{tj})$ -- t è il numero di termini indice
  - vettore di pesi associati al documento d_j
- $g_i(d_j)=w_{ij}$: il peso del termine k_i per il documento d_j

Modelli classici di IR: assunzione

- Si assume che i pesi dei termini indice siano indipendenti
  - Questa assunzione è una semplificazione perché esistono delle correlazioni tra i termini di un documento
  - Es: computer → network
- Efficace nella pratica

Modelli classici di IR: categorie

- Teoria degli insiemi
  - Boolean
  - Extended Boolean
- Algebrici
  - Vettoriale (Vector Space)
  - Vettoriale generalizzato (generalized VS)
  - Latent Semantic Indexing (LSI)
- Probabilistici

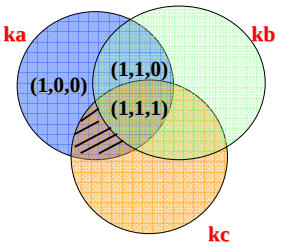
Modello Booleano

Modello Booleano

- Modello molto semplice basato sulla teoria degli insiemi
- Un documento è modellato come insieme di termini
- I pesi dei termini rispetto ai documenti hanno valori binari:
  - $w_{ij} \in \{0,1\}$
  - 1 indica che il termine compare nel documento, mentre 0 indica che il termine non compare nel documento
- Le query sono rappresentate come espressioni booleane
  - $q = ka \wedge (kb \vee \neg kc)$
  - q = (automobili $\wedge$(vendita $\vee\neg$fabbricazione)
- Similarità fra query e documenti
  - Solo 2 casi estremi: il documento soddisfa o soddisfa la query
  - La semantica è precisa, ma non permette ranking

Modello Booleano: valutazione delle query

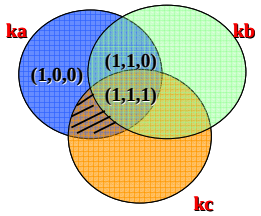
- Si supponga di avere la seguente query:
  - $q = ka \wedge (kb \vee \neg kc)$



- La query può essere riformulata in forma normale disgiuntiva (DNF)
  - $q_{dnf} = (ka \wedge kb) \vee (ka \wedge \neg kc) =$
    $(ka \wedge kb \wedge kc) \vee (ka \wedge kb \wedge \neg kc) \vee (ka \wedge \neg kb \wedge \neg kc)$

Modello Booleano: valutazione delle query

- Data la query in forma DNF
  - $q_{dnf} = (ka \wedge kb \wedge kc) \vee (ka \wedge kb \wedge \neg kc) \vee (ka \wedge \neg kb \wedge \neg kc)$
- La query viene rappresentata come insieme di vettori binari del tipo (ka, kb, kc)
  - $vec(q_{dnf}) = \{ (1,1,1), (1,1,0), (1,0,0) \}$
  - i vettori sono detti componenti congiuntive della query: $vec(q_{cc})$



Modello Booleano: valutazione delle query

- Criterio di similarità fra query e documenti:

$$sim(q,d_j) = \begin{cases} 1 \text{ se } \exists \text{ vec}(q_{cc}), \text{ tale che vec}(d_i) = \text{vec}(q_{cc}), \\ 0 \text{ altrimenti} \end{cases}$$

  - un documento è restituito solo se la sua similitudine con l'interrogazione è pari ad uno
- Esempio:
  - $q = vec(q_{dnf}) = \{ (1,1,1), (1,1,0), (1,0,0) \}$
  - $vec(d_j) = (0,1,0)$ non è rilevante per q anche se contiene kb
  - $vec(d_i) = (1,1,0)$ è rilevante per q: combacia con la seconda componente congiuntiva

Limitazioni del modello Booleano

- Il retrieval è basato su criteri di decisione binari
  - non esiste la nozione di corrispondenza parziale
  - non c'è un ordinamento parziale dei risultati (ranking)
- Gli utenti trovano difficile trasformare le loro richieste informative in una espressione booleana
  - Gli utenti formulano spesso query booleane troppo semplicistiche (congiunzioni di termini)
  - Di conseguenza, le query booleane restituiscono troppo pochi o troppi documenti
- Non supporta il feedback di rilevanza
  - Come modificare la query se un documento è giudicato rilevante/irrilevante dall'utente?

Modello Vettoriale

Modello Vettoriale:

Rappresentazione di documenti e query

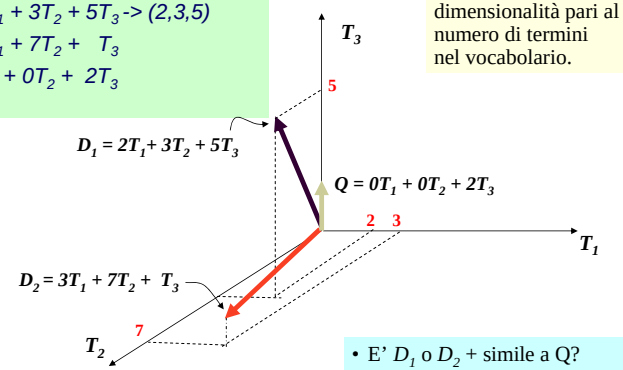
- Query e documenti sono rappresentati come vettori:
  - con una componente per ogni termine
  - il valore di una componente rappresenta il peso del termine corrispondente
- Documento d_j:
  - $w_{ij} > 0$ quando $k_i \in d_j$
  - $vec(d_j) = (w_{1j}, w_{2j}, ..., w_{tj})$
  - Un documento e' visto come multi-insieme di termini (un termine può occorrere piu' volte)
  - i pesi dei termini dipendono dalla loro frequenza di occorrenza nel documenti
- Query q:
  - $w_{iq} > 0$ quando $k_i \in q$
  - $vec(q) = (w_{1q}, w_{2q}, ..., w_{tq})$
  - Non ammette operatori booleani

Modello Vettoriale:

Rappresentazione nello spazio vettoriale

Esempio:
$D_1 = 2T_1 + 3T_2 + 5T_3$ -> (2,3,5)
$D_2 = 3T_1 + 7T_2 + T_3$
$Q = 0T_1 + 0T_2 + 2T_3$

Lo spazio ha una dimensionalità pari al numero di termini nel vocabolario.



- E' D_1 o D_2 + simile a Q?
- Come misurare il grado di similarità?

Modello Vettoriale:

Valutazione delle query

- Retrieval basato sulla similarità fra query e documenti
  - La similarità si basa sulle frequenze di occorrenza dei termini (nei documenti e nella query)
  - I documenti recuperati possono essere ordinati in base alla similarità con la query (ranking)
- Relevance feedback automatico:
  - Documenti rilevanti possono essere "aggiunti" alla query
  - Documenti irrilevanti possono essere "sottratti" alla query

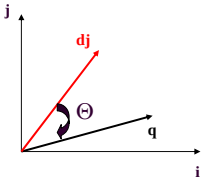
Modello Vettoriale

Similarità: misura del coseno

- Similarità fra documento e query = coseno dell'angolo fra due vettori
  - Si misura quanto i due vettori hanno un'inclinazione simile

$$sim(q,d_j) = \cos(\Theta) = [vec(d_j) \bullet vec(q)] \; / \; |d_j| * |q| =$$

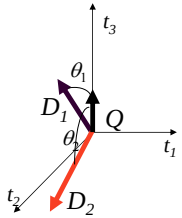
$$\frac{\left(\sum_i W_{ij} * W_{iq}\right)}{\sqrt{\sum_i W_{ij}^2} * \sqrt{\sum_i W_{iq}^2}}$$



  - Il prodotto scalare (al numeratore) è normalizzato con la lunghezza dei vettori
    - permette di astrarre dalla lunghezza del documento

Modello Vettoriale

Misura del coseno: esempio



$$sim(dj,q) = \frac{\vec{dj} \bullet \vec{q}}{|\vec{dj}| \times |\vec{q}|} = \frac{\sum_{i=1}^{t} w_{i,j} \times w_{i,q}}{\sqrt{\sum_{i=1}^{t} (w_{i,j})^2 \times \sum_{i=1}^{t} (w_{i,q})^2}} = \cos\Theta$$

Q = (0,0,2)

D_1 = (2,3,5) $sim(D_1,Q)=(0\times2+0\times3+2\times5)/\sqrt{(4+9+25)(0+0+4)} = 0.81$

D_2 = (3,7,1) $sim(D_2,Q)=(0\times3+0\times7+2\times1)/\sqrt{(9+49+1)(0+0+4)} = 0.13$

D_1 è 6 volte migliore di D_2 usando la misura del coseno.

Modello Vettoriale

Misura del coseno: esempio

- Docs:
  - Austen's *Sense and Sensibility* (SAS)
  - Austen's *Pride and Prejudice* (PAP)
  - Bronte's *Wuthering Heights* (WH)

	SaS	PaP	WH
affection	115	58	20
jealous	10	7	11
gossip	2	0	6

	SaS	PaP	WH
affection	0.996	0.993	0.847
jealous	0.087	0.120	0.466
gossip	0.017	0.000	0.254

- Similarity scores:
  - cos(SAS, PAP) = .996 x .993 + .087 x .120 + .017 x 0.0 = 0.999
  - cos(SAS, WH) = .996 x .847 + .087 x .466 + .017 x .254 = 0.929

Vantaggi del modello vettoriale

- La formula del coseno dell'angolo consente di ordinare i documenti rispetto al grado di similarità con la query
  - Poiché possono verificarsi matching parziali fra documenti e query, è possibile ottenere risposte che approssimino le richieste dell'utente
- Il peso dei termini influenza la qualità delle risposte

Modello Probabilistico

- Data una query *q* e un documento *d*, si stima la probabilità che l'utente consideri il documento *d* rilevante
  - Il modello assume che tale probabilità dipenda solo dalla query e dal modo in cui il documento è rappresentato
  - I documenti trovati possono essere ordinati in ordine descrescente rispetto alla probabilità di rilevanza
- La specifica della query consiste nel definire le caratteristiche della risposta ideale
  - Data una query esiste sempre un insieme di documenti che costituiscono la risposta ideale
  - Le caratteristiche riguardano la distribuzione dei termini
  - Il problema è capire quali sono tali caratteristiche
    - All'inizio viene effettuata un'ipotesi su quali queste caratteristiche possono essere
    - Tale ipotesi viene poi raffinata durante un processo di iterazione

Modello Probabilistico

- Pesatura dei termini
  - nel modello probabilistico classico i pesi sono binari (0/1)
  - il modello è detto anche “Binary Independence Retrieval”
- Criterio di ranking
  - Si basa sulla probabilità che i documenti siano rilevanti rispetto alla query
  - Due approcci per stimare la probabilità di rilevanza:
    - La probabilità di rilevanza è basata solo sulla presenza dei termini cercati nei documenti
    - La probabilità di rilevanza è basata sulla presenza e sull’assenza dei termini cercati nei documenti

Modello probabilistico

Calcolo del ranking

- Data una query q e un documento d_j
  - R = insieme dei documenti relevanti per q
  - $\underline{R}$ = insieme dei documenti non relevanti per q
  - Probabilità che d_j sia rilevante: $P(R|d_j)$
  - Probabilità che d_j sia irrelevante: $P(\underline{R}|d_j)$
  - w_{ij}= peso del termine t_i nel documento d_j
  - w_{iq}= peso del termine t_i nella query q
- Ranking di d_j rispetto a q: $P(R|d_j) / P(\underline{R}|d_j)$

$$Sim(d_j,q) = P(d_j|R) / P(d_j|\underline{R}) =$$

$$\sum_i w_{ij}\, w_{iq} \left(\log(\mathbf{P(t_i|R)} / (1-\mathbf{P(t_i|R)}) + \log((1-\mathbf{P(t_i|\underline{R})}) / P(t_i|\underline{R})) \right)$$

Dove w_{ij} e w_{iq} sono numeri binari (0/1)

Modello probabilistico

Stima delle probabilità $P(t_i|\boldsymbol{R})$, $P(t_i|\underline{\boldsymbol{R}})$

- Passo 0
  - n_i = numero di documenti contenenti k_i
  - $P_0(t_i|\boldsymbol{R}) = 0.5$
  - $P_0(t_i|\underline{\boldsymbol{R}}) = n_i / N$
- Passo 1
  - N^r = # di documenti con ranking > r (soglia)
  - n_i^r = # di documenti con ranking > r contenenti t_i
  - $P_1(t_i|R)=n_i^r/N^r \quad \rightarrow \quad \mathbf{P_1(t_i|R) = (n_i^r + n_i/N) / (N^r + 1)}$
  - $P_1(t_i|\underline{R})=(n_i-n_i^r)/(N-N^r) \rightarrow \mathbf{P_1(t_i|\underline{R})=(n_i- n_i^r+n_i/N)/(N-N^r+1)}$
- Si continua ad iterare

Modello probabilistico

Vantaggi e svantaggi

- Vantaggi
  - I documenti sono ordinati rispetto alla probabilità di rilevanza
- Svantaggi:
  - È necessario indovinare buone stime iniziali per $P(k_j | R)$
  - Usa pesi binari (ignora la frequenza/importanza di un termine rispetto ad un documento)
  - Independence assumption

Confronto fra modelli classici di IR

- Il modello Booleano è il meno potente in quanto non consente il matching parziale
- Risultati sperimentali indicano che il modello vettoriale ha prestazioni migliori del modello probabilistico

Altri modelli di IR

- Modello booleano esteso
- Modelli strutturali

Modello Booleano Esteso

- Il modello booleano è semplice ed elegante ma non consente ranking
- Modello Booleano Esteso
  - Estende il modello booleano mediante la nozione di corrispondenza parziale e di pesatura dei termini
  - Combina le caratteristiche del modello vettoriale con le proprietà dell'algebra booleana
  - introdotto nel 1983 da Salton, Fox, e Wu

Modello Booleano Esteso

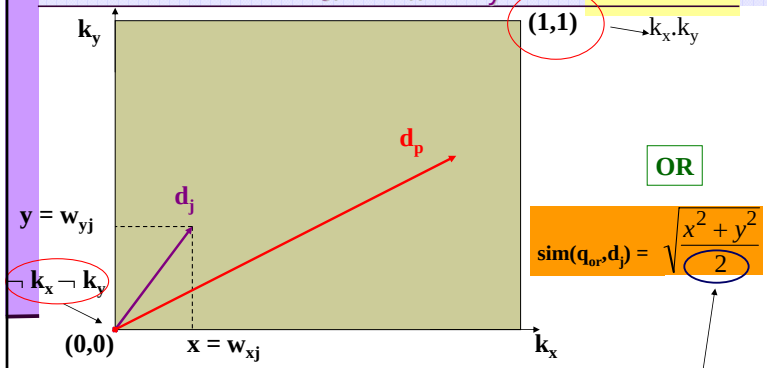
- Il peso w_{ij} associato a $[k_i,d_j]$ è normalizzato: $0 \leq w_{ij} \leq 1$

$$w_{ij} = tf(i,j) \times \frac{idf(t_i)}{\max_k(idf(t_k))}$$

- Si considerino query del tipo:
  - $q_{OR} = k_x \vee k_y$
  - $q_{AND} = k_x \wedge k_y$

Modello Booleano Esteso

Query OR: $q_{or} = k_x \vee k_y$



OR

$sim(q_{or},d_j) = \sqrt{\frac{x^2 + y^2}{2}}$

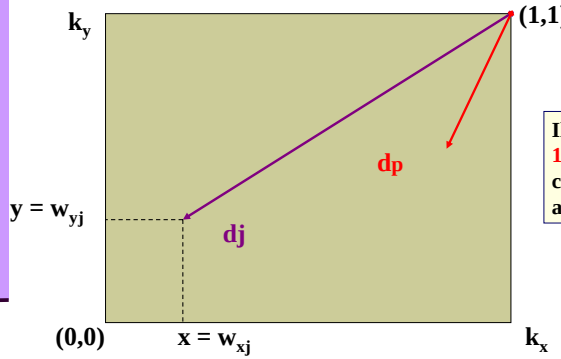
distanza normalizzata

x e y sono i pesi normalizzati di k_x e k_y in dj

dj è tanto più simile a q quanto più il punto di coordinate (x,y) è distante da (0,0)

Modello Booleano Esteso

Query AND: $q_{and} = k_x \wedge k_y$



Il punto ottimo è 1,1 che corrisponde a $k_x \wedge k_y$

AND

$sim(q_{and}, d_j) = 1 - \sqrt{\frac{(1-x)^2 + (1-y)^2}{2}}$

Modello Booleano Esteso

Query più complesse

- Se esistono t termini indice, il modello può essere esteso ad uno spazio t-dimensionale
- Espressioni contenenti sia AND sia OR
  - esempio: $(k1 \wedge k2) \vee k3$

$$sim(q,d) = \sqrt{\frac{\left(1 - \left(\sqrt{\frac{(1-x_1)^2 + (1-x_2)^2}{2}}\right)\right)^2 + (x_3)^2}{2}}$$

Modelli Strutturali

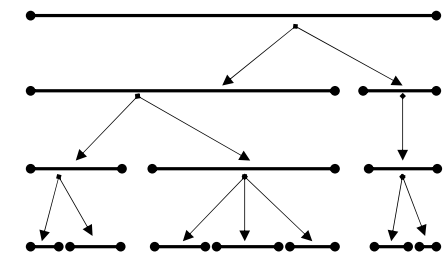
Modelli Strutturali

- I modelli classici) considerano una struttura piatta per i documenti, basata su parole-chiave
  - Se cerco “consiglio di amministrazione” potrei trovare documenti in cui queste parole compaiono ma non sono correlate
- Inoltre, il peso di una parla è lo stesso sia che la parola compaia nel testo che, ad es. nel titolo

Proximal Nodes

- Il testo è rappresentato con strutture di indicizzazione gerarchiche
  - Strutturate in capitoli, sezioni, sottosezioni, paragrafi, linee
  - Ognuna di queste componenti è un nodo (della gerarchia)
- Ad ogni nodo è associata una regione di testo
- Definizioni:
  - Regione: una porzione contigua di testo
  - Nodo: componente strutturale del testo
  - Match point: la posizione del testo in cui occorre una parola o sequenza di parole

Proximal Nodes

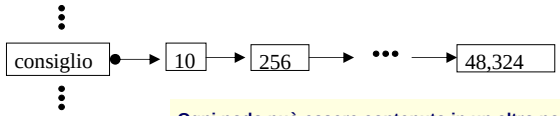


Capitolo

Sezione

Sottosezione

Paragrafo



Ogni nodo può essere contenuto in un altro nodo
Due nodi sullo stesso livello non si possono sovrapporre

Proximal Nodes: interrogazioni

- Le query sono espressioni regolari, è possibile cercare stringhe e far riferimento a componenti strutturali
- E’ possibile fare query del tipo:

$$[(*section)with((consiglio)and(amministrazione))]$$

Proximal nodes: risposte alle query

$$[(*section)with((consiglio)and(amministrazione))]$$

1. Per ogni termine, trova la lista delle componenti strutturali che contengono un'occorrenza del termine
   - Scandisci la lista delle occorrenze del termine 'consiglio' e per ogni occorrenza, trova nell'indice gerarchico le componenti strutturali (in questo caso solo le sezioni) che includono la posizione del termine
   - Analogamente, per ogni occorrenza del termine 'amministrazione', trova le componenti strutturali (sezioni) che contengono tale occorrenza
2. Combina le liste ottenute al passo precedente per I vari termini
   - Nella query considerata, poiché è presente l'operatore AND, è necessario calcolare gli elementi comuni alle due liste (intersezione)