Il *Natural Language Processing*

Chris Manning and Hinrich Schütze, *Foundations of Statistical Natural Language Processing*, MIT Press. Cambridge, MA: May 1999

NLP

NLP è alla base della maggior parte dei processi di TM

L'obiettivo principale di ricerca sul NLP è analizzare (grammaticalmente, identificandone le singole POS e il loro ruolo) e comprendere il linguaggio

Nasce all'interno dell'Intelligenza Artificiale, si sviluppa, negli anni '50 del Novecento, con le prime applicazioni di traduttori automatici e con scarsissimo successo.

Ad oggi, pur disponendo di analizzatori sintattici e di altri strumenti sofisticati (compresi efficienti *traduttori*), l'obiettivo è ancora lontano

Resta il fatto che molti strumenti sviluppati per l'NLP sono oggi diffusamente utilizzati nel TM

Strumenti di NLP entrati nel TM (1)

1. *Tokenization* (token=segno):

   suddivisione del testo in unità (token) *elementari (parole, numeri, date, segni di punteggiatura, ecc.), di regola delimitate da spazi*

2. *Stemming* (stem=ramo):

   estrazione della radice di una parola, rimovendo affissi e desinenze (es. ridendo, ridere, rideva *a* ride- *, così come* riso *e* risata*)*

3. *Lemmatization* (lemma)

   *identificazione della voce del vocabolario della lingua (*lemma*) a partire da una parola con desinenza*

N.B. *Stemming* e *Lemmatization* differiscono, perché quest'ultimo deve anche risolvere problemi di ambiguità poiché una forma flessa può provenire da più di un lemma:

canti — cantare/canto

botte ← botte/botta

Strumenti di NLP entrati nel TM (2)

4. *Finding Collocation (Term Extraction):*

   dove per collocation *si intende un'espressione che consiste di due o più parole, che corrispondono ad un uso linguistico convenzionale:* strumenti di distruzione di massa *La loro caratteristica è che il loro significato non può essere ricavato dal significato dei singoli termini che la compongono*

5. *Finding N-grams:*

   dove per n-gram *si intende una sequenza generica di n parole (bi-grammi; tri-grammi; tetra-grammi), non legati ad un particolare uso idiomatico*

6. *Anaphora resolution:*

   dove anafora *è la «relazione tra un'espressione linguistica ed un'altra che la precede»:*
   «E' venuto Luigi?» «Sì, lo ho incontrato al bar»; lo è un'anafora per Luigi. E' un compito molto difficile (anche per un essere umano) e anche i migliori software per il trattamento di testi non riescono il più delle volte a risolvere un'anafora

7. *Word Sense Disambiguation:*

   consiste nel determinare quale significato abbia una parola "ambigua" nel contesto dell'analisi. La risoluzione del problema può avvenire o avvalendosi di dizionari *o utilizzando metodi di apprendimento basati su testi-campione, o richiedendo l'intervento esterno del ricercatore*

Parole ambigue

Esistono diverse situazioni di ambiguità:

- *Polisemia:* un lemma cui corrispondono più significati
  - Farfalla: è un insetto, ma anche un elemento del motore che prende il nome dall'insetto, o una cravatta
  - Fine aggettivo = elegante; sottile

- *Omografia:* la stessa sequenza di caratteri è comune a due lemmi:
  - Fine sostantivo maschile = obiettivo
  - Fine sostantivo femminile = termine

  N.B. Quando i problemi non sono di natura semantica, ma sintattica ci si trova di fronte a problemi di *part-of-speech tagging* (es. canto)

L'*Information Retrieval*

Definizione

L'obiettivo principale di un sistema di *IR* è recuperare l'informazione che potrebbe essere utile o significativa, sulla base di una *query* definita dall'utente, su un *corpus* identificato di documenti. L'operazione che costituiscono questo processo, riguardano, quindi, rappresentazione, memorizzazione, organizzazione e accesso all'informazione

Alle origini, IR aveva una ristretta nicchia di interesse (bibliotecari ed esperti di informazioni, es. agenzie di stampa)

L'avvento del Web ha cambiato radicalmente le cose

- *Una sorgente di informazioni virtualmente illimitata*
- *Accesso universale ed a basso costo*
- *Non esiste un controllo editoriale centralizzato*
- *Molti nuovi problemi si pongono: IR è vista come una area chiave per identificare soluzioni appropriate*

Accesso a dati e Accesso a informazioni

- **Data retrieval**:

  trovare oggetti che soddisfino condizioni ***chiaramente specificate mediante una espressione regolare*** o di algebra relazionale.

  Es. FIND persona WHERE eta>40 AND laurea=1 OR diploma=1

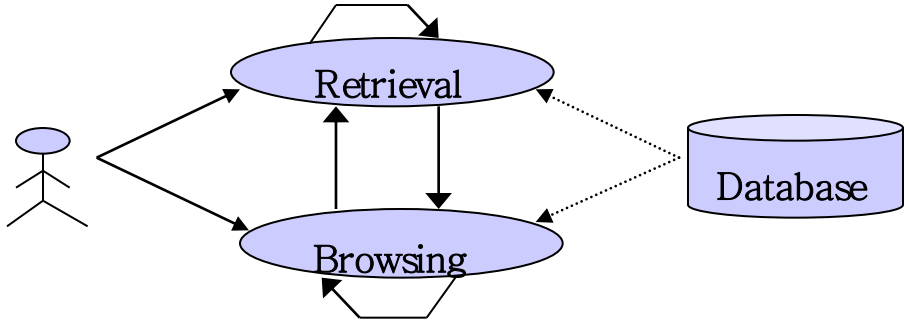
  N.B. Un errore è segnale di totale fallimento: la risposta non esiste

- **Information retrieval**:

  richiede una *interpretazione* della richiesta dell'utente. I documenti recuperati non possono essere classificati come "buoni" o "cattivi", ma vanno associati ad una misura di *rilevanza* rispetto alla richiesta dell'utente (o meglio: all'interpretazione della richiesta)

La nozione di rilevanza è centrale in IR

Le azioni dell'utente



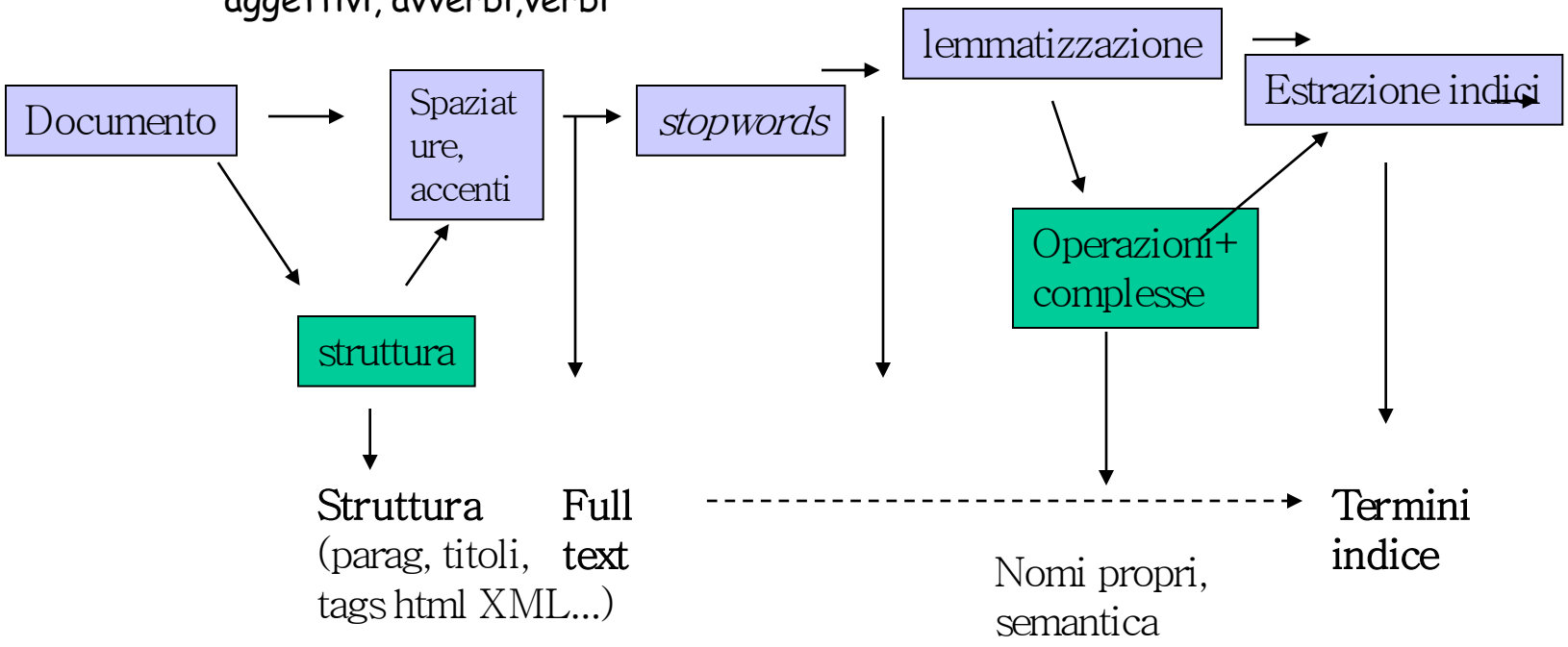
- Una sessione di **Retrieval** comporta la specifica degli interessi dell'utente e la sua trasformazione in una query (usualmente, un insieme di parole chiave o *keywords*)
- Se l'interesse dell'utente è mal specificato, o è molto vasto, l'utente può utilizzare una interfaccia interattiva (es. finestre a scelta multipla), visualizzare alcuni documenti proposti, seguire *hyperlinks* a partire da documenti che più lo interessano, o dettagliare meglio la sua *query*. Si parla allora di una sessione di **Browsing**.

3 fasi fondamentali

- Operazioni sui testi (query e documenti)
- Generazione di indici (strutture di puntamento)
- Ricerca e ranking

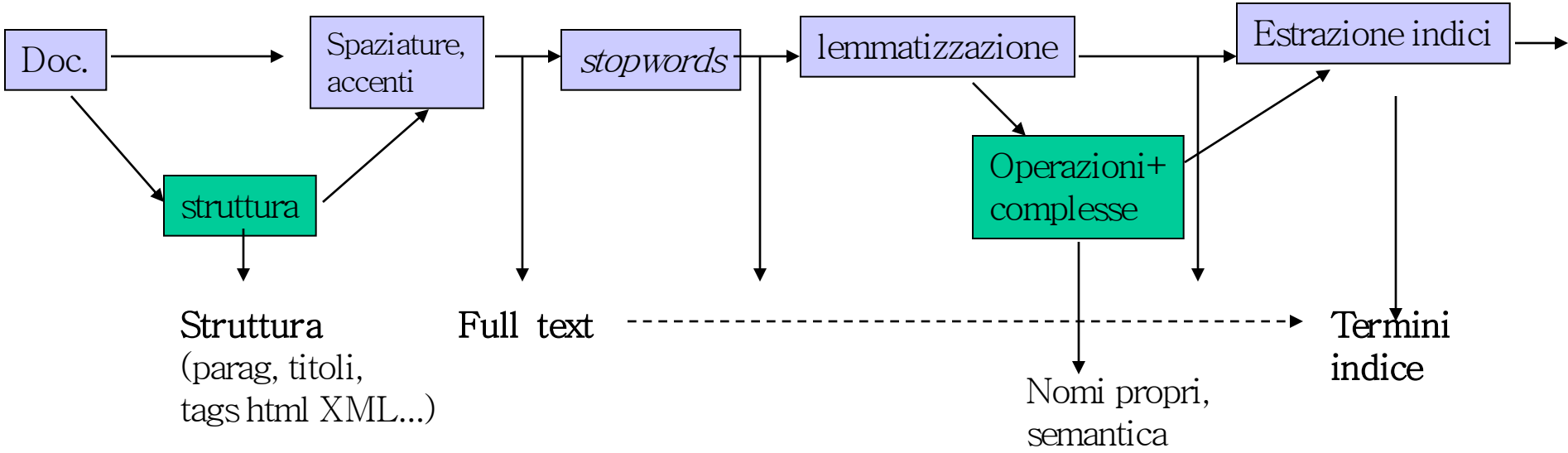
Operazioni sui testi

- generazione di una rappresentazione astratta dei documenti e delle richieste degli utenti (query) , detta *logical view*:
  - *full text logical view*
  - *Index term view*: riduzione a *parole chiave*: eliminazione delle *stopwords*, di aggettivi, avverbi,verbi



- Costruzione dell'*index* del testo

Operazioni sui testi



Indicizzazione dei documenti

- Ogni documento viene rappresentato mediante un insieme di parole-chiave o *termini indice*
- Un termine-indice è una parola ritenuta utile per rappresentare il ***contenuto*** del documento
- In genere, nei sistemi di IR "classici", gli indici sono nomi, perché maggiormente indicativi del contenuto
- Tuttavia, nei motori di ricerca vengono considerati tutti i termini (full text representation)
- Gli indici vengono utilizzati per generare ***strutture di puntamento*** ai documenti della collezione, facilitandone il recupero a fronte di una query.

Ranking (ordinamento)

- Un *ranking* è un ordinamento dei documenti recuperati che dovrebbe riflettere gli interessi dell'utente
- E'basato su :
  - Identificazione di gruppi di termini comuni
  - Condivisione di termini pesati
  - Probabilità di rilevanza
- La classificazione dei modelli di IR è basata si diversi criteri di ranking

Costruzione del vocabolario

- Il vocabolario viene generato scandendo i testi del *repository*
- Vengono effettuate operazioni preliminari sui testi, di cui abbiamo parlato, al fine di limitare la taglia del vocabolario
- Ad ogni parola del vocabolario, quale che sia la struttura dati utilizzata, viene associata la lista delle occorrenze nei documenti
- Ogni parola incontrata in un testo viene prima cercata nel vocabolario: se non viene trovata, viene aggiunta al vocabolario, con una lista inizialmente vuota di occorrenze.

Costruzione del vocabolario (2)

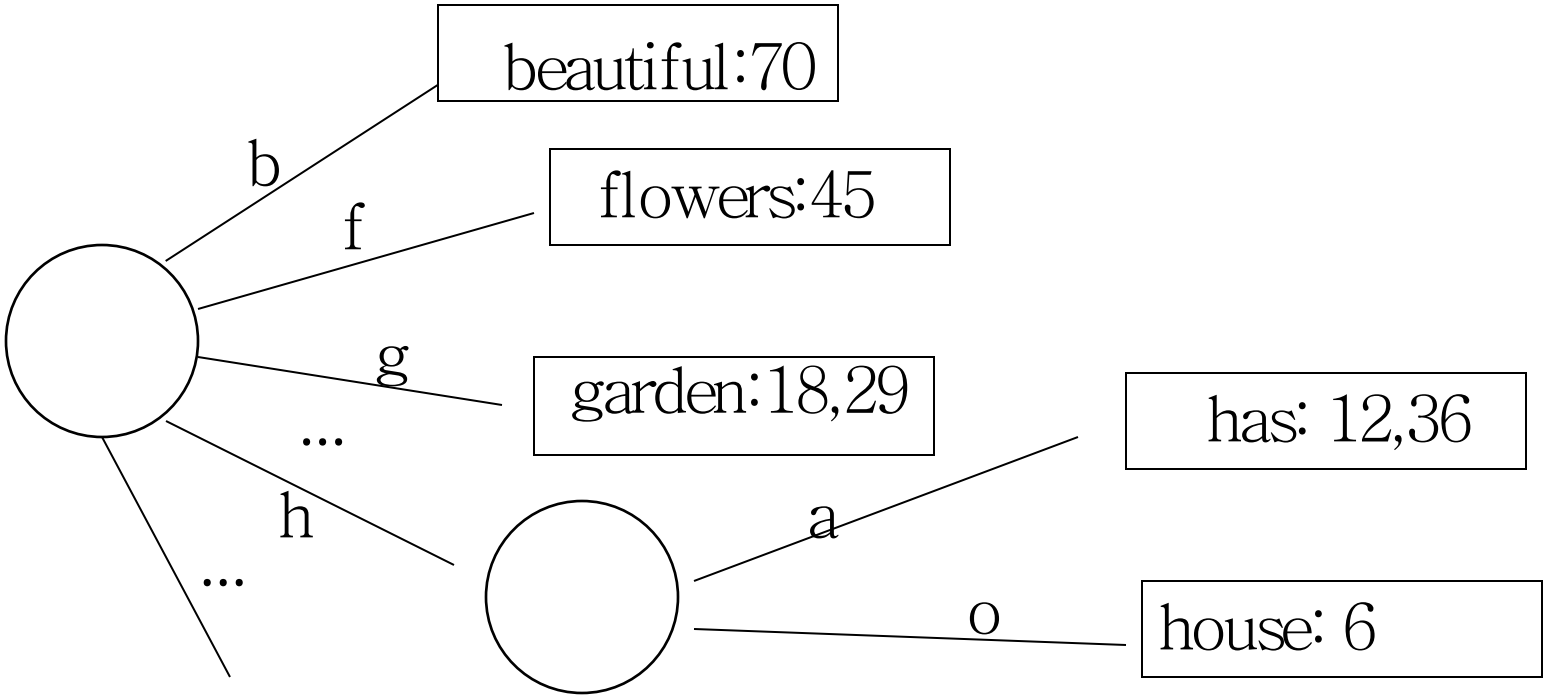
- Una volta che si siano esaminati tutti i testi, il vocabolario viene memorizzato sul disco con la lista delle occorrenze. Vengono generati due files:
  - Nel primo file, vengono memorizzate in locazioni contigue le liste delle occorrenze
  - Nel secondo file, il vocabolario è memorizzato in ordine lessicografico (alfabetico), e viene generato per ogni parola un puntatore alla sua lista di occorrenze nel secondo file. Questo consente, durante la ricerca, di mantenere il vocabolario in memoria.
- L'intero processo ha un costo *O(n)* nel caso peggiore. La ricerca (binaria) ha un costo O(logn)

Memorizzazione del vocabolario in un trie

n. ordine dei caratteri

1 6 12 16 18 25 29 36 40 45 54 58 66 70

That house has a garden. The garden has many flowers. The flowers are beautiful



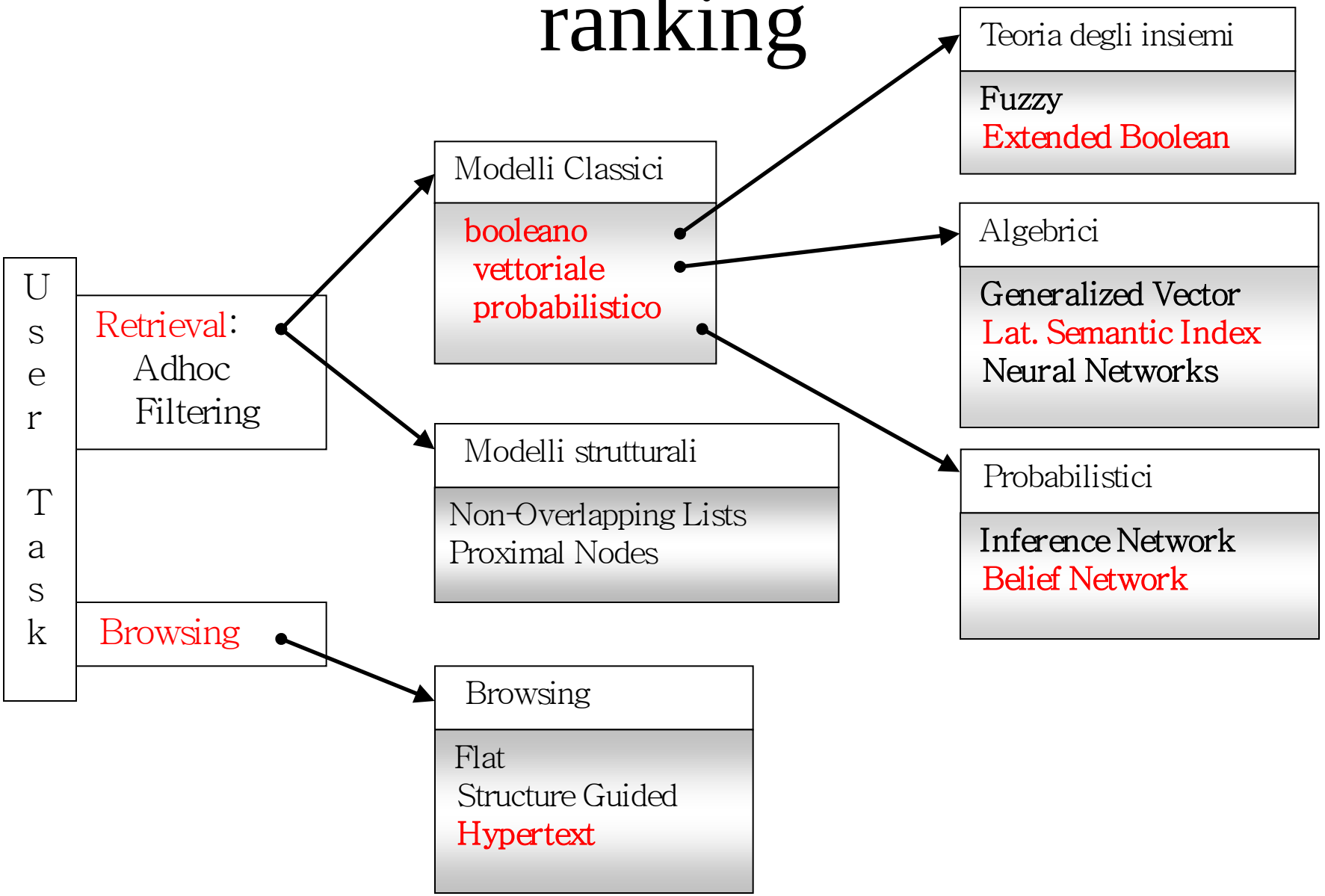
3. Metodi di Ranking:

come ordinare i documenti per “rilevanza” rispetto ad una query

Modelli di IR

- Supponiamo di avere a disposizione una visione logica, e questa sia data da un insieme di parole-chiave o *keywords*.
- Un banale *matching* di parole chiave fra documenti e query spesso fornisce risultati modesti, gli utenti sono insoddisfatti.
- Inoltre, spesso gli utenti non sono in grado di esprimere i loro interessi mediante un elenco appropriato di *keywords*
- Il problema è resto più grave se siamo nell'ambito Web IR
- Un *ranking* appropriato dei documenti ha un effetto notevole sulle prestazioni (vedi i motori di ricerca)

Una tassonomia dei modelli di ranking



Modelli “classici”: concetti base (1)

- Non tutti i termini sono ugualmente importanti per la rappresentazione di un testo (o query). Nella selezione di termini indice è importante assegnare un peso di rilevanza alle varie parole.
- L’importanza di un termine indice è rappresentata da un valore che ad esso viene associato
- Sia
  - *ki* un termine indice
  - *dj* un documento della collezione
  - *wij* un peso associato a *(ki,dj)*
- Il peso *wij* quantifica l’importanza dell’indice *ki* per descrivere il contenuto del documento

Modelli “classici”: **concetti base** (2)

- k_i è un termine indice
- d_j è un documento
- t è il numero totale di documenti in **D**
- $K = (k1, k2, ..., kt)$ è l’insieme dei termini indice
- $w_{ij} >= 0$ è un peso associato con (ki,dj)
- $w_{ij} = 0$ indica che un termine non appartiene al documento
- $vec(d_j) = \mathbf{dj} = \vec{dj} = (w1j, w2j, ..., wtj)$ è un vettore pesato associato *a dj*
- $gi(\mathbf{dj}) = wij$ è una funzione che restituisce il peso associato alla coppia (ki,dj)

Modello Booleano

- Un modello molto semplice basato sulla *teoria degli insiemi*
- Le *query* vengono rappresentate mediante espressioni booleane
  $q = k_a \wedge (k_b \vee \neg k_c)$
  Es: (automobili $\wedge$ *(vendita* $\vee$ $\neg$ *fabbricazione)*
- I termini sono presenti o assenti. Dunque, $wij \in \{0,1\}$

PROBLEMI

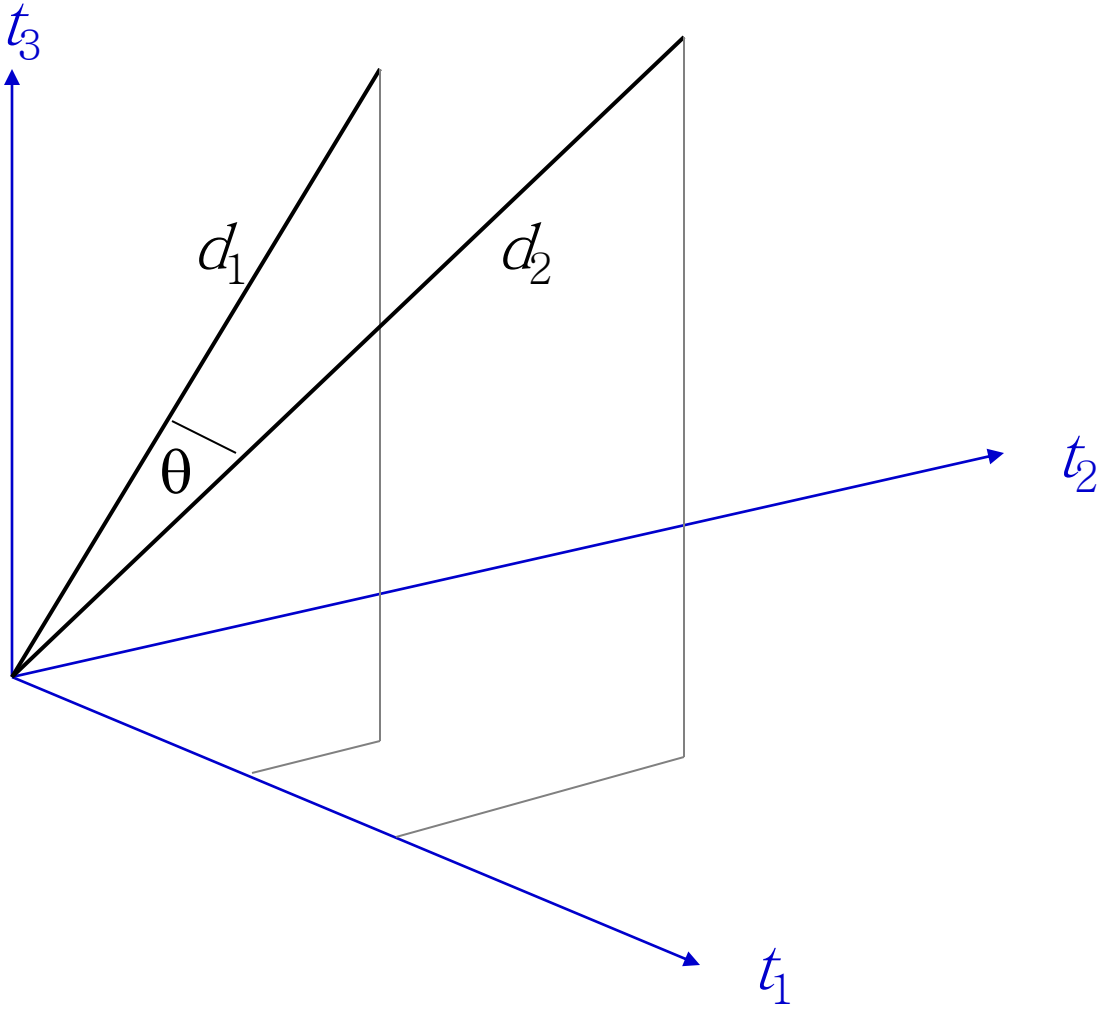
- Il retrieval è basato su criteri di decisione binari, non esiste la nozione di corrispondenza parziale
- Non viene fornito un ordinamento parziale dei documenti (non c'è ranking!!)
- Gli utenti trovano difficile trasformare le loro richieste informative in una espressione booleana
- Gli utenti formulano spesso query booleane troppo semplicistiche (congiunzioni di termini)
- Di conseguenza, le query booleane restituiscono o troppo pochi o troppi documenti

Modello vettoriale

- Definisci:
  - Ad ogni termine *ki è associato un vettore unitario*
  - I vettori unitari si assumono essere ortonormali (si assume che i termini appaionao indipendentemente l'uno dall'altro nei documenti)
- I *t* vettori unitari formano una base ortonormale in uno spazio t-dimensionale
- In questo spazio, query e documenti sono rappresentati mediante vettori pesati

Lo spazio vettoriale

La dimensione dello spazio è pari al numero di termini nel vocabolario



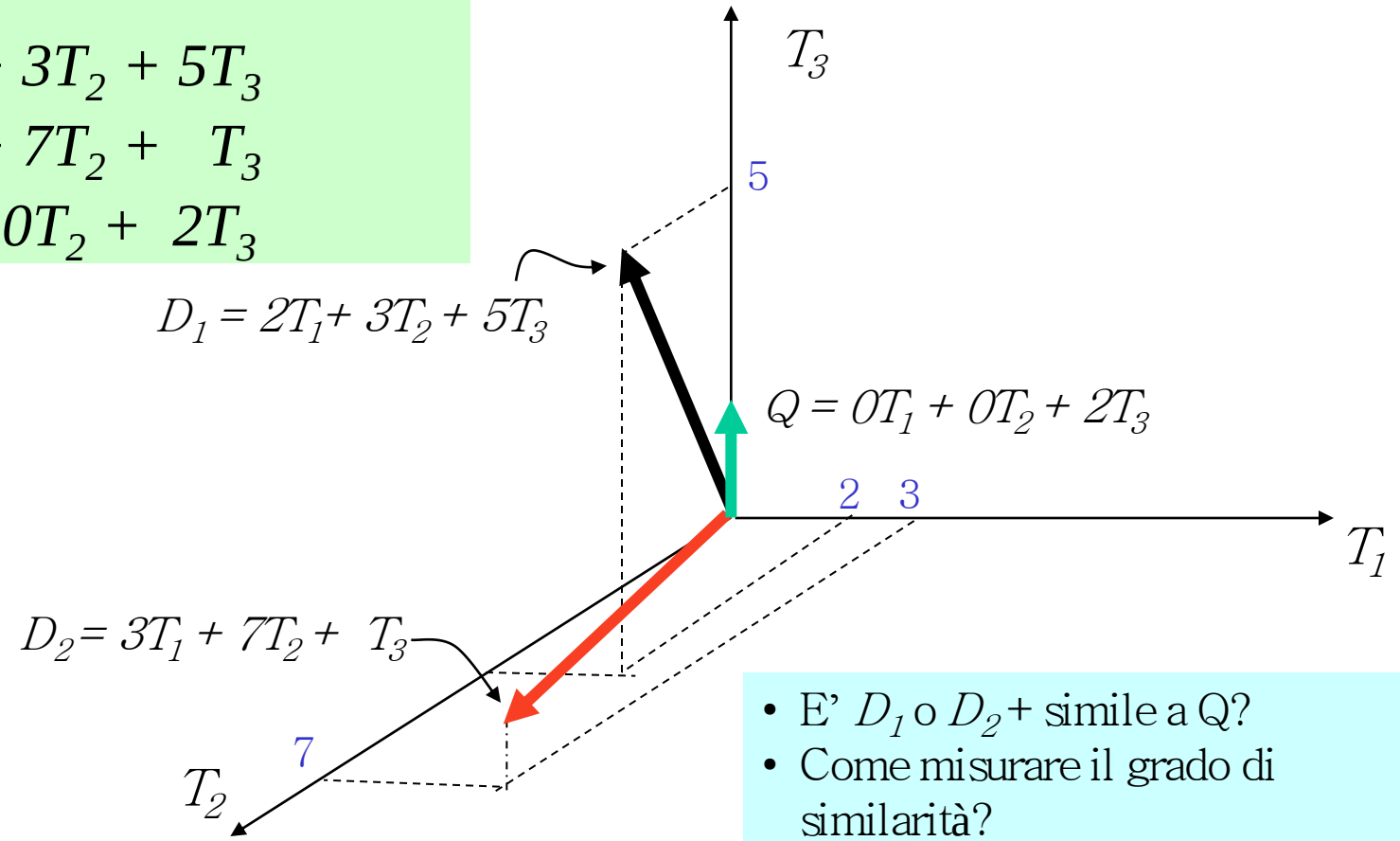
Rappresentazione grafica

Esempio:

$D_1 = 2T_1 + 3T_2 + 5T_3$

$D_2 = 3T_1 + 7T_2 + T_3$

$Q = 0T_1 + 0T_2 + 2T_3$

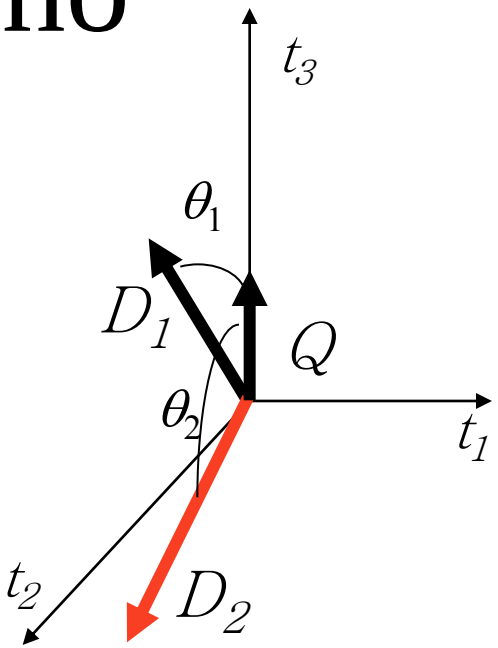


- E' D_1 o D_2 + simile a Q?
- Come misurare il grado di similarità?

Misura del coseno

- Si misura il coseno dell'angolo fra due vettori.
- Il prodotto scalare (al numeratore) è normalizzato con la lunghezza dei vettori.

$$\text{CosSim}(\vec{d_j}, \vec{q}) = \frac{\vec{d_j} \bullet \vec{q}}{|\vec{d_j}| \cdot |\vec{q}|} = \frac{\sum_{i=1}^{t} (w_{ij} \cdot w_{iq})}{\sqrt{\sum_{i=1}^{t} w_{ij}^2 \cdot \sum_{i=1}^{t} w_{iq}^2}}$$



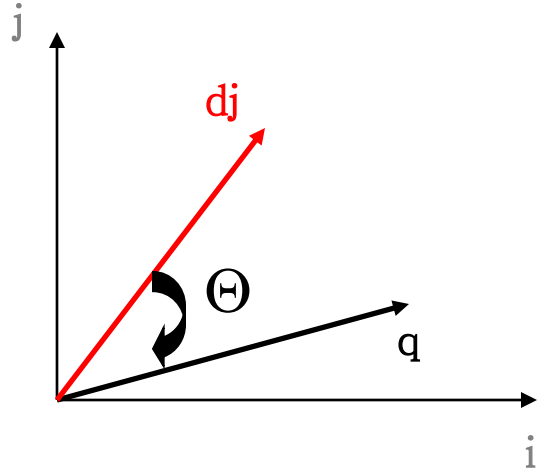
$D_1 = 2T_1 + 3T_2 + 5T_3$ $\quad \text{CosSim}(D_1, Q) = 10 / \sqrt{(4+9+25)(0+0+4)} = 0{,}81$

$D_2 = 3T_1 + 7T_2 + 1T_3$ $\quad \text{CosSim}(D_2, Q) = 2 / \sqrt{(9+49+1)(0+0+4)} = 0{,}13$

$Q = 0T_1 + 0T_2 + 2T_3$

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

Similarità nel modello vettoriale



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

Pesatura dei termini nel modello vettoriale

- Una misura del peso di un termine in un documento deve tenere conto di due fattori :
  - Quantificazione del peso che il termine ha nel documento:
    - *Fattore tf (term frequency)* $tf(i,j) = \frac{freq_{i,j}}{\max_k(freq_{k,j})}$
  - Quanto il termine aiuta a discriminare il documento dj dagli altri documenti in D
    - *Fattore idf (inverse document frequency)*

    Dove: N numero di doc. nella collezione, n_i numero dei documenti in cui il termine ki appare $idf_i = \log \frac{N}{n_i}$
  - *wij = tf(i,j) * idf(i)*

Peso dei termini nella query

- Salton e Buckley suggeriscono:

$$w_{i,q} = \left(0,5 + \frac{0,5\, freq_{i,q}}{\max\limits_{k}(freq_{k,q})} \right) \times log \frac{N}{n_i}$$

Metodi per il Text retrieval

- VSM (Vector Space Model)

  I documenti sono rappresentati da punti in uno spazio multidimensionale. La similarità tra i documenti è definita in termini di distanza tra i punti o di angolo tra i vettori. La ricerca è effettuata calcolando la distanza tra il vettore query e i vettori documento (full search)

- LSI (Latent Semantic Indexing)

  La matrice documenti X occorrenze è scomposta tramite la SVD. La riduzione di dimensionalità si ottiene conservando solo le prime componenti ed effettuando una ricerca full search nel sottospazio ottenuto