Domande e Risposte

Gabriele Pozzani

Corso di Laurea Magistrale in Editoria e Giornalismo, A.A. 2011/2012
Università degli Studi di Verona, Italia

D.1: come si risolve un esercizio che richiede di calcolare la lunghezza di ricerca attesa?

R.1: prendiamo in esame l'esercizio 4 proposto nel fax-simile d'esame pubblicato sulla pagina web del corso. Il testo dell'esercizio è il seguente.

Si calcoli la lunghezza di ricerca attesa supponendo che l'utente voglia 6 documenti rilevanti e che l'insieme dei documenti recuperati venga suddiviso nei seguenti 3 sottoinsiemi:

- S1 contiene 5 documenti di cui 2 rilevanti e 3 non rilevanti
- S2 contiene 4 documenti di cui 3 rilevanti e 1 non rilevante
- S3 contiene 4 documenti di cui 2 rilevanti e 2 non rilevanti

La soluzione si calcola nel seguente modo.

L'utente legge i doc in S1 e trova solo 2 doc ril sui 6 voluti, ma per farlo ha comunque dovuto esaminare anche gli altri 3 nell'insieme, quindi: 2 doc ril trovati e 5 doc letti finora.

A questo punto passa a S2, legge altri 4 documenti (che fanno 9 con i 5 letti in S1) e trova altri 3 doc rilevanti (e quindi finora ha 5 doc rilevanti).

Non avendo ancora trovato il num di doc ril desiderati, deve leggere anche S3. A questo punto però gli basta trovare un solo doc ril dei 2 disponibili in S3, quindi il numero di documenti che l'utente deve esaminare in S3 dipende dalla posizione del primo doc rilevante nella lista dei 4 doc in S3.

Non sapendo come il sist. di IR ordina i doc all'interno dei sottoinsiemi, dobbiamo assumere che l'ordinamento in S3 sia casuale, e qui entrano in gioco la teoria delle variabili casuali. Essendo l'ordinamento in S3 casuale tutte le possibili combinazioni/ordini di due doc ril e due non rilevanti hanno la stessa probabilità di essere fornite all'utente. Quindi si calcola il numero medio (o valore atteso) di doc da leggere per trovare il primo doc ril sui 4 facendo la media su tutti i possibili ordini dei documenti in S3. Tutti i possibili ordini sono:

R	R	NR	NR
R	NR	R	NR
R	NR	NR	R
NR	R	R	NR
NR	R	NR	R
NR	NR	R	R

All'utente serve un solo doc ril quindi dobbiamo osservare la posizione del primo doc ril in ognuno degli ordinamenti e quanti doc l'utente deve leggere per arrivare a tale posizione. come si vede:

- in 3 casi (i primi tre) su 6 l'utente legge 1 solo doc perché trova subito il doc ril;
- in 2 casi (il quarto e il quinto) su 6 l'utente legge 2 doc perché il doc ril è al secondo posto;
- in 1 caso (l'ultimo) su 6 l'utente legge 3 doc perché il doc ril si trova in terza posizione.

A questo punto, il valore atteso di doc che l'utente deve leggere per trovare il primo doc rilevante in S3 è la media del numero di doc da leggere nei tre casi ma pesato per il numero di combinazioni per ognuno dei casi, cioè:

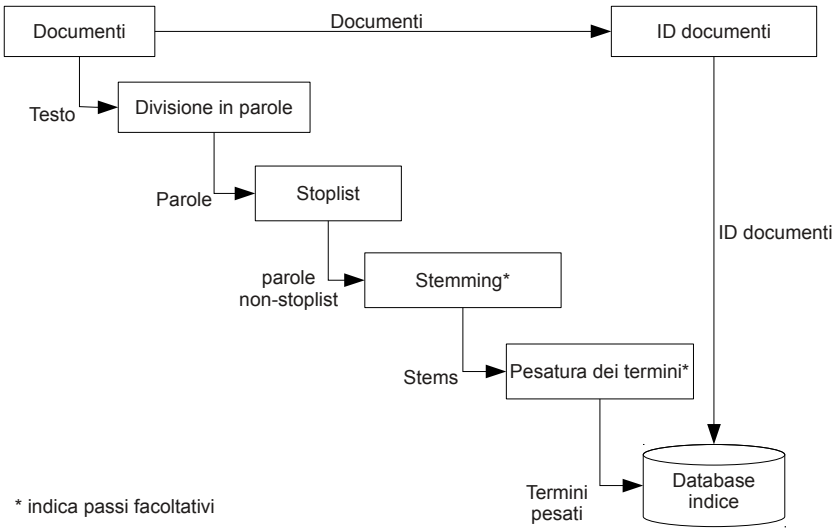
$$\frac{3}{6} \times 1 + \frac{2}{6} \times 2 + \frac{1}{6} \times 3 = \frac{10}{6} \approx 1,67$$

Questo, infine, va sommato al numero di documenti giàà letti dall'utente in S1 e S2, cioé:

$$5 + 4 + 1,67 = 10,67$$

La lunghezza di ricerca attesa è quindi 10,67.

D.2: A lezione ci ha spiegato che a conclusione del processo di indicizzazione otteniamo, dopo l'eventuale pesatura dei termini, il database indice (o solamente indice). Questa informazione è riprodotta nello schema seguente:



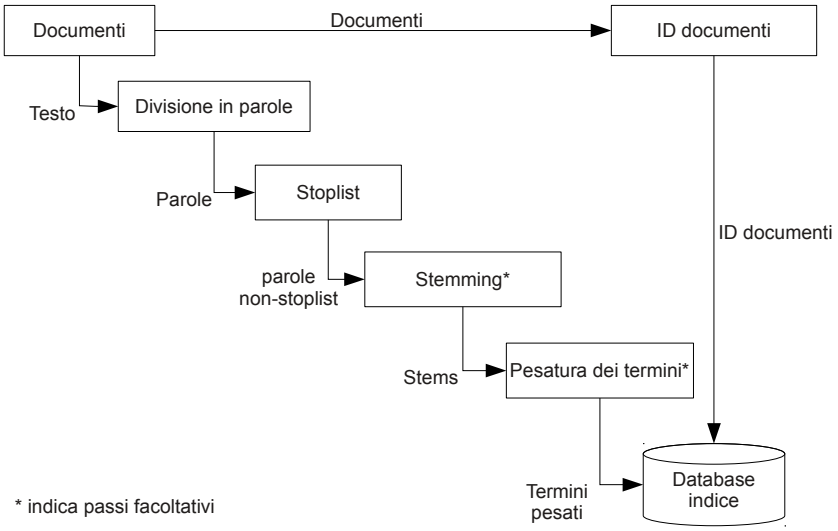
Fornendoci la definizione formale di thesauro ci ha spiegato che è il vocabolario (lista di termini) di un linguaggio di indicizzazione. Il thesauro è dunque l'indice di cui sopra?

R.2: **No, il thesauro non è l'indice**. **L'indice è una struttura in cui i termini sono associati con i documenti in cui appaiono**, quante volte appaiono e altre meta-informazioni aggiuntive che dipendono dal sistema di IR. **Il thesauro**, come abbiamo visto, invece non ha a che fare con i documenti, ma solo con i termini. **Mette in relazione (di vari tipi) termini con termini**.
Il thesauro può essere usato in fase di indicizzazione per decidere quali termini cercare e indicizzare nei documenti (ad esempio solo i Preferred Terms) in modo da ridurre il vocabolario e quindi la dimensione dell'indice. Questo si intende dicendo che costituisce il vocabolario del linguaggio di indicizzazione, nel senso che è come se il sistema conoscesse solo i termini nel thesauro e quindi ricerca e indicizza solo questi.
In realtà viene molto più spesso usato in fase di interrogazione (come fa google). Il sistema cerca e indicizza nei documenti tutti i termini (cioè qualunque parola/sequenza di caratteri, che siano parole di senso compiuto, errori di battitura o NPT o altro). Poi quando l'utente cerca un termine il thesauro viene usato per tornare (magari con una rilevanza minore) non solo i documenti che contengono quel termine, ma anche i documenti che contengono termini in relazione ad esso nel thesauro (ad esempio: sinonimi, antonimi, PT, NPT, ecc. . .).

D.3: a cosa serve la rappresentazione matriciale e come è collegato all'indicizzazione dei documenti?

R.3: le matrici illustrano il tipo di informazioni, e un modo per rappresentarle, che costituiscono l'indice di un sistema di IR e che viene ottenuto come risultato dell'indicizzazione dei documenti.
In altre parole, il famoso indice a cui il sistema di IR accede per ricercare i documenti in risposta ad una query dell'utente contiene l'associazione tra i termini e i documenti in cui appaiono (eventualmente con il numero di occorrenze) (matrice termine-documento), l'associazione tra termini che appaiono nello stesso documento (matrice termine-termine) e/o l'associazione tra documenti che contengono uno stesso termine (matrice documento-documento). Le matrici quindi sono possibili modi per rappresentare queste associazioni nell'indice. Poi in realtà un sistema di IR memorizza anche altri tipi di associazioni e informazioni su termini e/o documenti. Inoltre, come abbiamo visto, un sistema di IR usa altre rappresentazioni, diverse da quella maticiale, in modo da essere più efficiente (le matrici sono "costose" perché occupano molto spazio e richiedono molto tempo per essere lette/processate/analizzate).

D.4: come avviene l'indicizzazione e quale è il suo schema funzionale?
R.4: lo schema funzionale del processo di indicizzazione è il seguente:



Il processo parte dall'insieme dei documenti (o dei loro surrogati) da indicizzare e li legge e analizza dividendoli in parole (=termini). Si ottiene così la lista di tutti i termini che occorrono in tutti i documenti.
Da questa lista di termini vengono eliminati quelli troppo frequenti (ad esempio: articoli, preposizioni, ecc. . .) applicando la cosiddetta stoplist (cioè la lista dei termini da non considerare perché poco influenti). Si nota che la stoplist dipende dalla lingua di riferimento e che o viene calcolata osservando la frequenza dei termini e togliendo quelli troppo o troppo poco frequenti o è conosciuta a priori dal sistema di IR.
Opzionalmente la lista dei termini fin qui ottenuta può essere ulteriormente raffinata applicando ulteriori due passi. Lo stemming mantiene per ogni termine in lista solo la sua radice (se più termini hanno la stessa radice questa viene considerata una sola volta), ad esempio i termini "associazione" "associazioni", "associare", "associo", . . . hanno tutti la stessa radice "assoc". Lo stemming è usato in alcuni casi per ridurre il linguaggio di indicizzazione e in altri casi (durante un'interrogazione) per recuperare documenti (magari con una rilevanza minore) che contengono termini "simili" a quello inserito dall'utente, come avviene anche con l'uso dei thesauri (se cerco "accociare", lo stemming permette di recuperare anche i documenti che contengono, ad esempio, "associato"). In questo secondo caso la lista delle radici (stems, al plurale, in inglese) non è usata al posto della lista di termini ottenuta dopo la stoplist, ma si "affianca" ad essa e viene usata insieme ad essa dal sistema di IR per rispondere alle interrogazioni.
Il secondo passo opzionale è la pesatura dei termini che permette di associare ai termini (o alle radici) un peso (un valore numerico) che ne rappresenti una proprietà. I pesi possono rappresentare quanto un termine è frequente o la sua sua IDF (Inverse Document Frequency) o altre proprietà simili. Anche in questo caso i pesi possono essere mantenuti a parte della lista dei termini vera e propria ed essere usati o meno a seconda del tipo di interrogazione e/o del sistema di IR.
Alla fine di tutti questi passi il processo ha prodotto una lista di termini che costituisce il linguaggio o vocabolario di indicizzazione, cioè la lista di parole di cui tenere traccia.
L'indice a questo punto non è altro che un database o una struttura (come ad esempio delle matrici, che abbiamo visto) che associa i termini, e/o le radici, ai documenti in cui appaiono e, eventualmente, altre informazioni di contorno, come ad esempio la posizione in cui appaiono. L'indice è usato dal sistema di IR per rispondere alle query senza dover leggere ogni volta i documenti. Quindi anche se l'indicizzazione può richiedere diverso tempo (ma si tratta comunque di secondi o minuti anche per migliaia di documenti), essa è eseguita una sola volta e permette di rispondere alle query in tempi rapidissimi (nell'ordine di decimi di secondo anche con query complesse o su migliaia di documenti).

D.5: Come funziona il matching probabilistico?

R.5: Si basa sull'idea che sia possibile calcolare la probabilità che un doc sia rilevante per la query.
La probabilità che un doc scelto a caso sia rilevante sull'insieme di doc totali su cui il sist esegue la query, è la proporzione fra il num di doc rilevanti (n) e il num tot di doc (N), cioè $P(RIL) = \frac{n}{M}$. (Che non sia ril è invece $P(\neg RIL) = \frac{(N-n)}{N}$). Ma nessun sistema di IR sceglie i doc a caso, esso effettua la sua ricerca su un doc rilevante per la query e calcola la probabilità che un doc sia selezionato (cioè recuperato/ritornato) quando è rilevante: $P(SEL|RIL)$. Questa formula si chiama (sempre) probabilità condizionata, condizionata appunto perché ha una condizione, in questo caso è che sia ril. Ma come calcoliamo tale formula? Trasformandola in un'altra formula equivalente applicando invece il teorema della probabilità composta e il teorema di Bayes.
Teorema della prob. composta: $P(A \cap B) = P(B)P(B|A)$
dove A e B rappresentano due eventi (come, per esempio, "essere rilevante" o "essere recuperato").
Quando gli eventi sono indipendenti il teorema si semplifica a: $P(A \cap B) = P(A)P(B)$.
Dal teorema della prob. composta si ottiene il teorema di Bayes: $P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{P(A)P(B|A)}{P(B)}$.
Ma come verificare se i doc selezionati come ril in risposta a una query siano effettivamente quelli da ritornare? Si suppone che un doc sia selezionato se ha più probabilità di essere ril invece che non ril, cioè se $P(RIL|SEL) > P(\neg RIL|SEL)$.
Da questa idea si ottiene la funzione discriminante. Per decidere se ritornare o no un doc, il sist di IR utilizza la funzione discriminante, che restituisce un numero sulla base del quale viene deciso se ritornare o meno quel doc. Secondo tale funzione, un doc è rit se la proporzione fra la prob che sia rilevante quando selezionato (cioè $P(RIL|SEL)$) e la prob che non sia rilevante quando selezionato (cioè $P(\neg RIL|SEL)$) è maggiore di 1 (che è come dire che $P(RIL|SEL) > P(\neg RIL|SEL)$, ma in questo modo abbiamo un solo numero su cui eseguire la nostra decisione).

$$dis(sel) = \frac{P(RIL|SEL)}{P(\neg RIL|SEL)}$$

Questa proporzione equivale, per l'applicazione del teorema di Bayes, a

$$\frac{P(SEL|RIL)P(RIL)}{P(SEL|\neg RIL)P(\neg RIL)}$$

All'inizio abbiamo detto cosa sono $P(RIL)$ e $P(\neg RIL)$, quindi non ci rimane che dire come calcolare $P(SEL|RIL)$ e $P(SEL|\neg RIL)$. Un documento è selezionato in relazione ai termini che compaiono in esso (che lo rappresentano), quindi **se** conosciamo il valore di ril dei singoli termini (t_1, t_2, ..., t_n), sarà sufficiente moltiplicare la probabilità dei singoli termini perché il sist stabilisca se ritornare o meno il doc che li contiene, cioè:

$$P(SEL|RIL) = P(t_1|RIL)P(t_2|RIL)\dots P(t_n|RIL)$$

e

$$P(SEL|\neg RIL) = P(t_1|\neg RIL)P(t_2|\neg RIL)\dots P(t_n|\neg RIL)$$

quindi il discriminante è:

$$dis(sel) = \frac{P(t_1|RIL)P(t_2|RIL)\dots P(t_n|RIL)P(RIL)}{P(t_1|\neg RIL)P(t_2|\neg RIL)\dots P(t_n|\neg RIL)P(\neg RIL)}$$