Applicazioni della SVD

Gianna M. Del Corso

Dipartimento di Informatica, Università di Pisa, Italy

28 Marzo 2014

1. Le applicazioni presentate
2. Text Mining
3. Algoritmo di riconoscimento di volti

Le Applicazioni

- Text Mining: Modello dello spazio vettoriale, LSI
- Algoritmo Eigenfaces per il riconoscimento automatico di volti umani

Text Maning

Per Text Mining si intendono metodi per estrarre informazioni utili da grandi collezioni di documenti testuali.

Preprocessing dei documenti e delle query

- Eliminazione di *stop words*
- *Stemming*
- *Creazione di una lista di termini* $\longrightarrow$ *indicizzazione dei documenti*

Il modello dello spazio vettoriale

Si crea una matrice termini-documenti A, in cui ogni documento è rappresentato da un vettore.

$A_{i,j} \neq 0$ se il documento $j-$esimo contiene il termine $i-$esimo

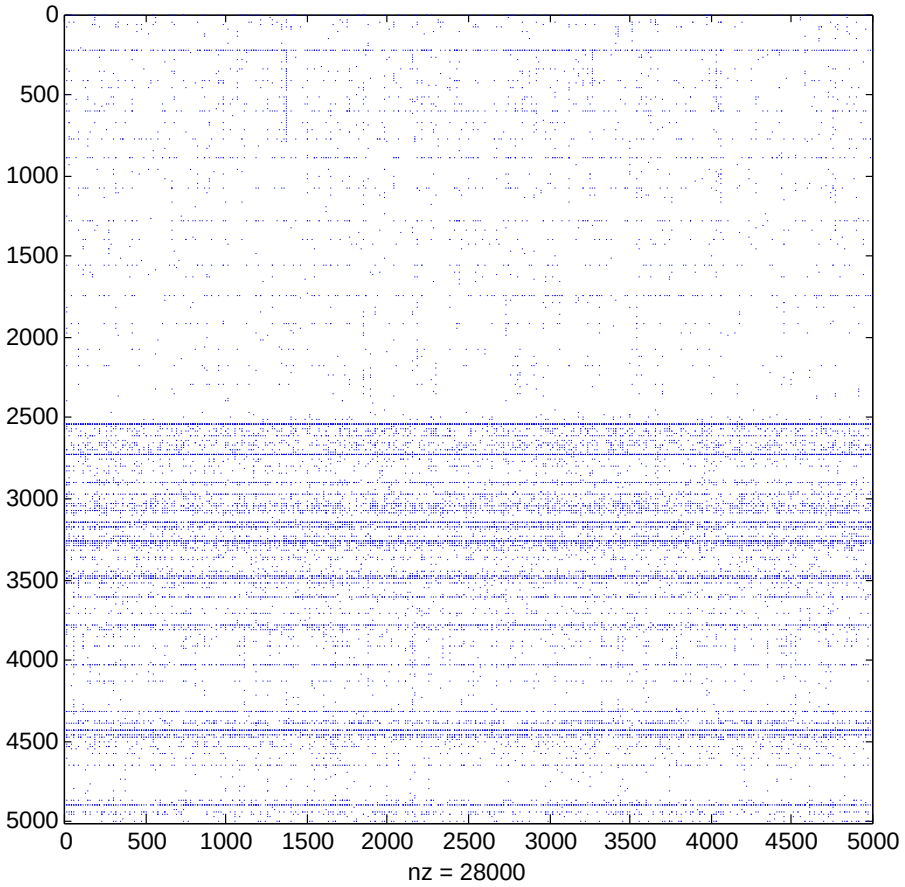
È comune applicare uno schema di "pesatura" dei termini ad esempio

$$A_{i,j} = f_{ij} \log(n/n_i)$$

dove f_{ij} è la frequenza del termine i nel doc j, e n_i è il numero di documenti in cui appare il termine i.
Se un termine è frequente solo in alcuni documenti, allora il termine discrimina bene tra differenti gruppi di documenti.

Il modello dello spazio vettoriale

La matrice è tipicamente molto sparsa



Il modello dello spazio vettoriale

Documenti

D1	Pronto soccorso per *neonati* e *divezzi*
D2	La cameretta di *neonati* e *bimbi* (per la vostra *casa*)
D3	La *sicurezza* dei *bimbi* a *casa*
D4	La *salute* del vostro *bimbo* e la sua *sicurezza*: dal *neonato* al *divezzo*
D5	*Nozioni* base di *sicurezza* per il *neonato*
D6	*Guida* con *nozioni* di base per la rimozione in *sicurezza* della ruggine
D7	*Guida* all'uso *salutare* del *Bimby*

Il modello dello spazio vettoriale

Ecco un esempio:

T1	neonat(o,i,a,e)
T2	bimb(o,a,i,e, y)
T3	guida
T4	salut(e, are)
T5	casa
T6	sicurezza
T7	divezz(o,i)
T8	nozion(e,i)

Il modello dello spazio vettoriale

Abbiamo lo seguente matrice termini documenti

$$A = \begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

Il modello dello spazio vettoriale

Normalizzando per colonna- dividendo per $\|A(:,i)\|_2$ abbiamo

$$A = \begin{pmatrix} 0.7071 & 0.5774 & 0 & 0.4472 & 0.5774 & 0 & 0 \\ 0 & 0.5774 & 0.5774 & 0.4472 & 0 & 0 & 0.5774 \\ 0 & 0 & 0 & 0 & 0 & 0.5774 & 0.5774 \\ 0 & 0 & 0 & 0.4472 & 0 & 0 & 0.5774 \\ 0 & 0.5774 & 0.5774 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.5774 & 0.4472 & 0.5774 & 0.5774 & 0 \\ 0.7071 & 0 & 0 & 0.4472 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.5774 & 0.5774 & 0 \end{pmatrix}$$

Query matching

Ogni query può essere rappresentata con un vettore nello spazio vettoriale. In questo caso i vettori delle query sono lunghi 8 perchè abbiamo 8 termini.
Supponiamo la nostra query sia *La sicurezza del neonato in casa*, viene rappresentata come il vettore

$$q = (1, 0, 0, 0, 1, 1, 0, 0)^T$$

.
Quali sono i documenti più vicini alla query?

Query matching

Una delle tecniche più semplici è la misura del coseno dell'angolo tra la query e i vettori dei documenti. Si ricorda che il coseno dell'angolo compreso tra due vettori $\mathbf{x}$ e $\mathbf{y}$ è dato da

$$\cos(\widehat{\mathbf{xy}}) = \frac{\mathbf{x}^T\mathbf{y}}{\|\mathbf{x}\|_2\|\mathbf{y}\|_2}.$$

Caolcolando queste quantità vengono fuori i seguenti valori:

$$cos(\theta) = (0.4082, 0.6667, 0.6667, 0.5164, 0.6667, 0.3333, 0)^T$$

Documenti restituiti come rilevanti la query (cioè con $\cos(\theta)$ più alto) *La sicurezza del neonato in casa* sono il 2, 3 e 5, cioè...

Query matching

D1 Pronto soccorso per *neonati* e *divezzi*
D2 La cameretta di *neonati* e *bimbi* (per la vostra *casa*)
D3 La *sicurezza* dei *bimbi* a *casa*
D4 La *salute* del vostro *bimbo* e la sua *sicurezza*: dal *neonato* al *divezzo*
D5 *Nozioni* base di *sicurezza* per il *neonato*
D6 *Guida* con *nozioni* di base per la rimozione in *sicurezza* della ruggine
D7 *Guida* all'uso *salutare* del *Bimby*

Che sono rilevanti!

Riassumendo

Nel modello completo dello spazio vettoriale si procede nel seguente modo:

- Si costruisce la matrice A termini-documenti, nella quale $a_{ij} \neq 0$ se il termine i-esimo compare nel documento j-esimo.
- Si costruisce il vettore della query in modo simile $q_i \neq 0$ se il termine i compare nella query
- Si calcola il $\cos(\widehat{\mathbf{qa}_j})$ per $j = 1, \ldots, m$, m numero di documenti, a_j $j$esima colonna di A.
- Fissato un threshold t, restituisci come retrieved tutti i documenti j tali che $\cos(\widehat{\mathbf{qa}_j}) > t$.

Mettendo un threshold più basso sul coseno abbiamo che un numero maggiore di documenti sarà retrieved in seguito alla nostra query. Non è però detto che tutti i documenti restituiti siano anche rilevanti, ad esempio....

Query matching

Se la query fosse stata Guida al bimbo, avremmo avuto

$$\cos(\theta) = (0, 0.4082, 0.4082, 0.3162, 0, 0.4082, 0.8165)^T$$

che restituiva come rilevante il documento 7: *Guida* all'uso *salutare* del *Bimby*!! Ci saremmo invece aspettati i documenti 1, 3, 4, 5....

Precision e Recall

Due parametri ci permettono di capire la bontà del nostro algoritmo... Vorremmo

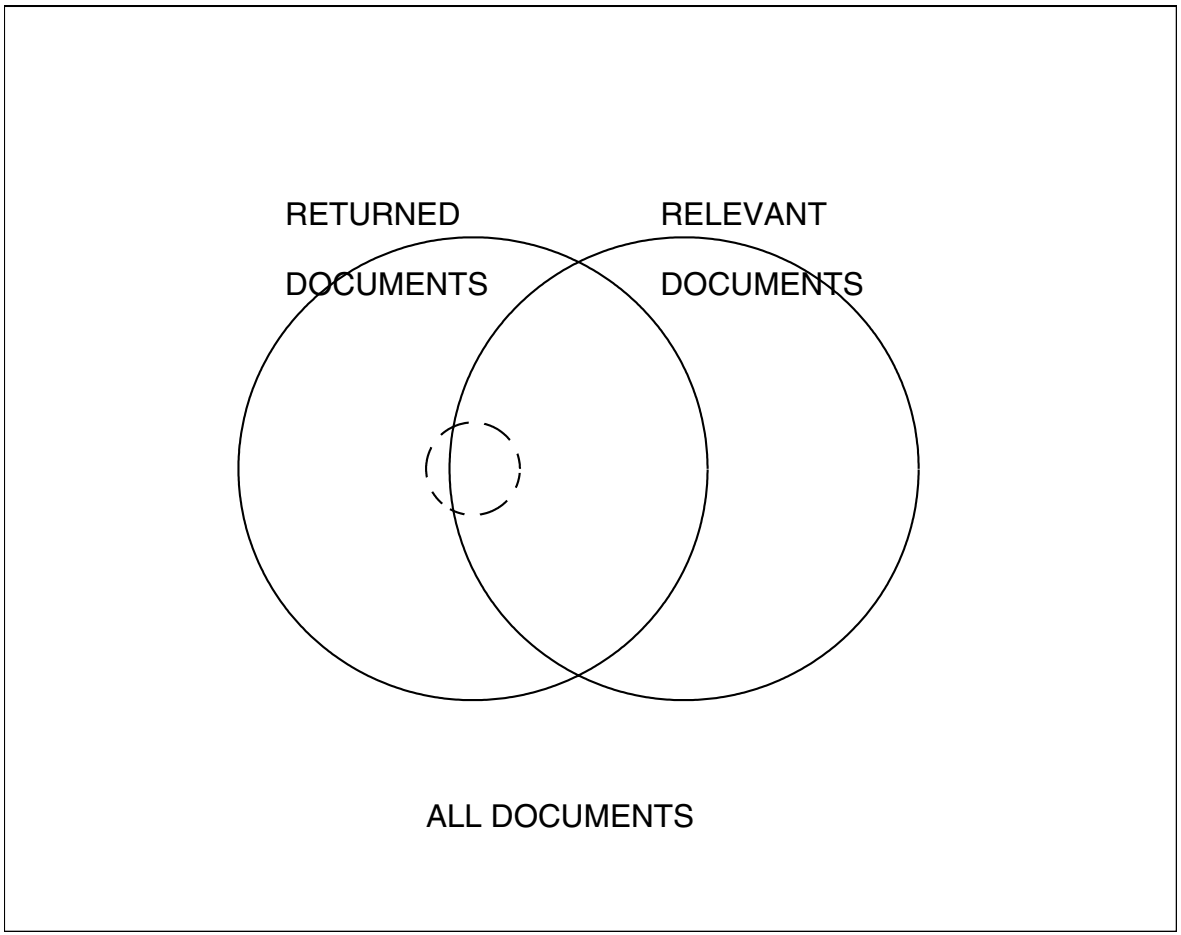
- Che i documenti retrieved fossero anche rilevanti
- Che i documenti rilevanti fossero anche retrieved

Si definiscono due quantità: la Precision $\mathcal{P}$ e la Recall $\mathcal{R}$ date da:

$$\mathcal{P} = \frac{D_r}{D_t}, \quad \mathcal{R} = \frac{D_r}{N_r},$$

dove D_r è il numero di documenti rilevanti che sono anche retrieved, D_t sono i documenti totali che sono retrieved, N_r è il numero totale di documenti rilevanti nel database.

Precision e Recall



Precision e Recall

Diminuendo il valore del threshold sulla misura del coseno, abbiamo che aumenta il numero di documenti retrieved ovvero D_t, e probabilmente anche D_r cioè i documenti nell'intersezione, cioè che sono sia rilevanti che retrieved, ma verranno generalmente restituiti anche documenti che non sono rilevanti!

Possibili miglioramenti

- criteri di pesatura dei termini
- approssimazioni di rango basse della matrice termini-documenti

DA NOTARE che per calcolare queste quantità abbiamo bisogno di conoscere i documenti rilevanti una query che devono essere quindi forniti all'utente.

Approssimazioni di rango basso:LSI

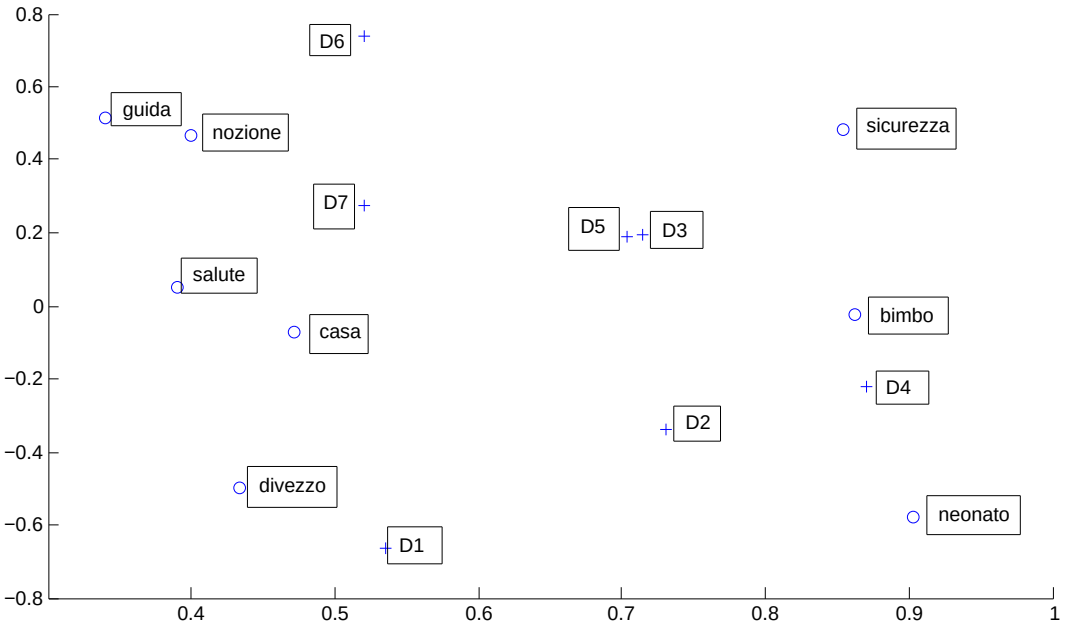
Parte del risultati negativi sembrano dovuti

- Scelta delle parole utilizzate
- differenza tra l'uso dei termini di diversi autori
- possibili errori

Sembra esserci una struttura sottostante latente nei dati che è distrutta dalla varietà delle parole usate nel testo.

Questa struttura può essere scoperta proiettando i dati in uno spazio di dimensione minore. Questo è il Latent semantic indexing

Approssimazioni di rango basso



Approssimazioni di rango basso

Usando la SVD possiamo proiettare sia i termini che i documenti in spazi di dimensione ridotta.
Sia A la matrice termini-documenti. $A = U\Sigma V^T$, approssimandola con una matrice di rango k,

$$A \approx U_k \Sigma_k V_k^T,$$

$U_k \approx$ lo spazio dei documenti, $V_k \approx$ lo spazio dei termini.

Query matching

In questo caso vogliamo confrontare una query $\mathbf{q}$ con una matrice di rango ridotta A_k

$$\cos(\theta_j) = \frac{(A_k \mathbf{e}_j)^T \mathbf{q}}{\|A_k \mathbf{e}_j\| \, \|\mathbf{q}\|} = \frac{(U_k \Sigma_k V_k^T \mathbf{e}_j)^T \mathbf{q}}{\|U_k \Sigma_k V_k^T \mathbf{e}_j\| \, \|\mathbf{q}\|}$$
$$= \frac{\mathbf{e}_j^T V_k \Sigma_k (U_k^T \mathbf{q})}{\|\Sigma_k V_k^T \mathbf{e}_j\| \, \|\mathbf{q}\|}$$

Query matching

Detto $\mathbf{s}_j = \Sigma_k V_k^T \mathbf{e}_j$, cioeè le colonne di A_k nella base di U_k, abbiamo

$$\cos(\theta_j) = \frac{\mathbf{s}_j^T (U_k^T \mathbf{q})}{\|\mathbf{s}_j\| \, \|\mathbf{q}\|}.$$

Non si deve esplicitamente calcolare A_k!
I vettori $\mathbf{s}_k$ possono essere calcolati una volta e usati per più query.

Query matching

Con la query La salute del bambino, utilizzando tutto lo spazio abbiamo

$$\cos(\theta) = (0, 0.4082, 0.4082, 0.6325, 0, 0, 0.8165),$$

proiettando sullo spazio di dimensione 2 otteniamo

$$\cos(\theta) = (0.4236, 0.6314, 0.6901, 0.6801, 0.6906, 0.4283, 0.6385)$$

D3 La *sicurezza* dei *bimbi* a *casa*
D5 *Nozioni* base di *sicurezza* per il *neonato*

Riassumendo

Nel modello approssimato dello spazio vettoriale si procede nel seguente modo:

- Si costruisce la matrice A termini-documenti, nella quale $a_{ij} \neq 0$ se il termine i-esimo compare nel documento j-esimo.
- Si calcola la SVD troncata al k-esimo termine di A, $A_k = U_k \Sigma_k V_k^T$.
- Si calcola il $\widehat{\cos(\mathbf{q}(A_k \mathbf{e}_j))}$ per $j = 1, \ldots, m$, m numero di documenti, cioè il coseno dell'angolo compreso tra il vettore query $\mathbf{q}$ e ogni colonna dell'approssimazione di rango k di A cioè di A_k. Può essere impiegata la formula vista in una slide precedente.
- Fissato un threshold t, restituisci come retrieved tutti i documenti j tali che $\widehat{\cos(\mathbf{q}(A_k \mathbf{e}_j))} > t$.

Riassumendo

Si verifica che in generale già con un valore piccolo di k abbiamo un miglioramento di Precision e Recall.