Relazione del Progetto

Modelli semantici distribuzionali

Corso di Elaborazione del Linguaggio Naturale del Prof. Fabio Tamburini

(22 giugno 2010)

Andrea Simonetto
Università di Bologna
simonett [at] cs [dot] unibo [dot] it
http://simonett.web.cs.unibo.it/

Indice

Introduzione

I modelli semantici distribuzionali[1] (in breve *DSM*) forniscono un interessante approccio al problema di trovare il significato delle parole, *senza* bisogno di imporlo dall'esterno specificando le relazioni che legano tra loro termini affini. Questi modelli – noti anche col nome di *semantiche basate su corpus*, *semantiche statistiche*, *modelli geometrici del significato* Widdows [2004], *semantiche vettoriali* o *modelli* WORDSPACE – concentrano l'attenzione sul significato delle parole. Ci sono vari livelli di *significato di una parola*, a seconda del punto di vista che adottiamo per definirlo; ad esempio, possiamo riferirci a:

- Significato nel mondo: ad esempio, il significato di *persona* è *l'insieme delle persone* nel mondo di riferimento (la sua estensione) oppure una funzione tra l'insieme dei *mondi possibili* e l'insieme delle persone in questi mondi (intensione, proprietà, ecc. . .). Parliamo in questo caso di semantica formale del linguaggio naturale, ed è l'approccio classico derivante dalla logica;

- Significato nella mente: il significato di *persona* è il concetto *PERSONA*, cioè la rappresentazione mentale della categoria delle persone: in questo caso parliamo invece di *psicologia cognitiva*;

[1] http://wordspace.collocations.de/

- Significato nel testo: il significato di *persona* è un'astrazione sul contesto linguistico in cui la parola *persona* compare: in questo terzo caso parliamo di semantiche distribuzionali.

L'*ipotesi distribuzionale* asserisce che:

> Perlomeno alcuni aspetti del significato delle espressioni lessicali dipendono dalle loro proprietà distribuzionali nel contesto linguistico. Il grado di somiglianza semantica tra due espressioni linguistiche A e B è funzione della somiglianza tra i contesti linguistici in cui A e B possono comparire.

La definizione di *contesto* è consistente con una nozione estesa di *contesti d'uso* di una parola, che include aspetti non linguistici (ad esempio aspetti dell'ambiente comunicativo). Di fatto il contesto è identificato col *contesto linguistico*, sia per ragioni pratiche (è più semplice collezionare ed elaborare contestici linguistici dai corpora), sia per questioni teoriche (è interessante investigare il ruolo delle distribuzioni linguistiche per modellare il significato delle parole).

Esistono alcune varianti all'ipotesi distribuzionale (in seguito *DH*), tra cui principalmente possiamo individuare la *DH debole*, che rappresenta un metodo quantitativo per effettuare analisi semantiche e induzione di risorse lessicali, che si può riassumere in:

> Il significato delle parole è riflesso nella distribuzione linguistica. Ispezionando un numero considerevole di contesti distribuzionali è possibile identificare quegli aspetti del significato che sono condivisi dalle parole che hanno distribuzioni contestuali simili.

Nella variante forte di *DH* si fa un'ipotesi cognitiva sulla formazione e sull'origine delle rappresentazioni semantiche:

> La distribuzione delle parole nel contesto ha uno specifico ruolo causale nella formazione della rappresentazione semantica di quella parola. Le proprietà distribuzionali delle parole nei contesti linguistici spiegano il comportamento semantico umano (ad esempio un giudizio di somiglianza semantica).

È interessante osservare che anche nelle scienze cognitive, l'uso di coordinate e dimensioni (cioè di strumenti algebrici, come vedremo fondamentali anche per i nostri scopi) per modellare processi mentali è la pietra angolare degli *spazi concettuali* (Gärdenfors [2004]).

Chiaramente noi siamo interessati alla variante debole di *DH*, le cui applicazioni coinvolgono:

- elaborazione del linguaggio naturale;
- e-language modeling;
- lessicografia;
- disambiguazione del significato delle parole;
- apprendimento di ontologie e thesauri;
- estrapolazione di relazioni;
- question answering;
- ...

Il nostro oggetto di indagine sono i *DSM*, modelli computazionali che costruiscono rappresentazioni semantiche contestuali dai dati di un corpus, ed il cui contenuto semantico è rappresentato da vettori ottenuti a partire da analisi statistiche dei contesti linguistici di ciascuna parola. Seminale per questo approccio è il lavoro di Hinrich Schütze (Schütze [1997, 1998]) che per primo ha indirizzato il problema dell'apprendimento ambiguo verso una soluzione basata sull'algebra vettoriale. Per un introduzione alle principali tematiche dell'elaborazione del linguaggio naturale, un testo di riferimento è Manning and Schütze [1999].

In questo progetto ci concentreremo particolarmente sul lavoro emerso dagli studi delle connessioni tra geometria e significato compiuti da Dominic Widdows (Widdows [2004]) e nello specifico su *Infomap*[2], un software che implementa una semplice *DSM*, testandolo su un piccolo corpus di italiano.

[2] http://infomap-nlp.sourceforge.net/

1 Modelli semantici distribuzionali

I *DSM* sono basati sull'assunzione che il significato di una parola può essere inferito dalla sua distribuzione nel testo. Inoltre questi modelli costruiscono dinamicamente una rappresentazione semantica – nella forma di spazi vettoriali a più dimensioni – attraverso un'analisi statistica dei contesti in cui ogni parola compare.

Formalmente i modelli semantici distribuzionali sono tuple $< T, C, R, W, M, d, S >$, dove:

- T sono gli elementi *target*, cioè le parole per cui i *DSM* forniscono una rappresentazione contestuale;
- C sono i contesti, in cui ogni $t \in T$ cooccorre;
- R è una relazione tra T e i contesti C;
- W schema di pesatura dei contesti;
- M matrice distribuzionale, $T \times C$;
- d funzione di riduzione dimensionale, $d : M \to M$;
- S metrica, tra i vettori di M.

Uno schema di elaborazione che a partire da un corpus produce un *DSM* può essere suddiviso in due fasi:

a) Fase "linguistica":

   1. Preprocessing del corpus (ad esempio *POS-tagging*, *lemmatizzazione*, ...);
   2. Selezione dei target e dei contesti;

b) Fase "matematica":

   1. Conteggio delle cooccorrenze target/contesto;
   2. Pesatura dei contesti (opzionale, ma raccomandata);
   3. Costruzione della matrice distribuzionale;
   4. Riduzione delle dimensioni della matrice (opzionale);
   5. Calcolo delle distanze tra i vettori nella matrice (ridotta), i.e. interrogazione del modello.

Nel seguito ci concentreremo su *Infomap*, un programma che implementa (alcune di) queste fasi, sviluppato nel 2004 presso la Stanford University da Stefan Kaufmann, Dominic Widdows, Beate Dorow e Scott Cederberg.

2 Fase linguistica

Il nostro corpus di prova è costruito prendendo l'archivio storico (ultimi 10 anni) di alcuni quotidiani disponibili gratuitamente online; sebbene appartenenti esclusivamente all'ambito giornalistico, questi testi costituiscono una solida base per un corpus di dimensioni notevoli e facile da costruire. Inoltre il testo giornalistico tende ad essere più concreto nello stile, ad esempio, di quello letterario, che contiene molte forme poetiche e retoriche di cui è difficile (spesso anche per gli esseri umani) individuare una semantica precisa. I quotidiani usati (tutti appartenenti al gruppo de *"Il sole 24 ore"*) sono:

- Il Resto del Carlino (http://ilrestodelcarlino.ilsole24ore.com/);
- Il Giorno (http://ilgiorno.ilsole24ore.com/);
- La Nazione (http://lanazione.ilsole24ore.com/);
- Quotidiano.net (http://quotidianonet.ilsole24ore.com/);
- Cavallo Magazine (http://cavallomagazine.quotidianonet.ilsole24ore.com/);
- Ubitennis (http://ubitennis.ilsole24ore.com/).

Il corpus così costruito consta di poco più di 100 *milioni* di parole (in valore assoluto), con un dizionario di oltre 750˙000 *termini* (chiaramente comprendenti parole straniere, sigle e altre forme non propriamente appartenenti all'italiano).

2.1 Preprocessing

Il corpus deve essere perlomeno tokenizzato: per quanto questa possa sembrare una banalità, nasconde degli aspetti spinosi come l'uso o meno degli accenti e altri aspetti specifici di ogni lingua. A titolo di esempio, nella lingua inglese l'apostrofo è considerato parte della parola (come in *don't*, *you're*, ecc...): per far funzionare la tokenizzazione di *Infomap* – che è basata su questa regola dell'inglese – su un corpus in italiano, bisogna modificare i sorgenti del programma.

Essendo costruito *ex novo*, il corpus non è stato lemmatizzato, cosa che (in retrospettiva) avrebbe certamente migliorato di molto la qualità dei risultati delle interrogazioni. Inoltre anche la marcatura delle parti del discorso sarebbe stata un ottimo strumento per ottenere risultati più mirati. Tuttavia ambedue queste elaborazioni preliminari avrebbero richiesto un enorme lavoro di taglio linguistico che esula dagli scopi di questo progetto.

Chiaramente c'è un trade-off tra un'analisi linguistica approfondita e la necessità di risorse specifiche per il linguaggio che si sta considerando: innanzitutto gli errori introdotti in questa fase di analisi linguistica possono

ripercuotersi drasticamente sulle fasi successive; inoltre ci sono più parametri da considerare e da mettere a punto.

Ad ogni modo la strategia di preprocessing influisce sulla successiva selezione dei target e dei contesti: il nostro caso è molto semplice perché volendo essere questo un mero esercizio accademico, ci siamo potuti limitare a considerare il corpus nella sua forma naturale riducendo al minimo la fase di preprocessing – è comunque stato un notevole lavoro finalizzato a rendere almeno omogeneo e tokenizzabile il corpus.

2.2 Target e contesti

Esistono vari modi di definire i contesti: analizziamo i principali in una breve carrellata:

- documenti come contesti: questo è l'approccio dei *document retrieval systems*. Per quanto non adatto nei *DSM*, questo approccio è spesso citato come antenato dei metodi che seguono;
- parole come contesti: approccio adottato in *Infomap*, ha diversi parametri, inerenti alla cosiddetta "finestra", cioè alle parole circostanti a quella considerata:
  - dimensione della finestra: stabilisce quello che in *Infomap* è chiamata *vicinanza testuale*, ossia quanta distanza può al più intercorrere tra la parola considerata e quella più lontana ancora appartenente al contesto.
  - forma della finestra: principalmente si considerano finestre rettangolari (come in *Infomap*, dove tutte le parole nella finestra hanno lo stesso peso) e triangolari (dove le parole più vicine a quella considerata hanno un peso maggiore);
- relazioni sintagmatiche come filtri sul contesto: qui solo le parole che sono collegate a quella considerata da una certa relazione sono selezionate come appartenenti al contesto;
- relazioni sintagmatiche come funzioni di tipo: i contesti sono selezionati in base al loro tipo.

È curioso sottolineare che per quanto sia un fattore critico (nonché certamente portatore di errori di approssimazione) gli autori di *Infomap* non hanno esplicitamente formulato congetture riguardanti la dimensione della finestra, limitandosi ad impostarla a 200 parole. È possibile che all'epoca della stesura del programma, la ricerca fosse ancora in uno stato embrionale e alcune delle questioni affrontate qui siano emerse solo in seguito.

I *target* sono le parole usate come "assi" nello spazio vettoriale. In *Infomap* considera come *target* le 1000 parole più frequenti del corpus, una

volta rimosse quelle menzionate in una lista (denominata *stoplist*) contente le parole frequenti ma "povere di significato" (come le parole grammaticali *mentre*, *perché*, *come*, *sebbene*, *che*, *quando*, ...), quelle eccessivamente ambigue (come i nomi propri), e le forme ripetute (*amato*, *amata*, ... ricordiamo che il corpus non è stato lemmatizzato).

3 Fase matematica: Infomap

Una WORDSPACE è uno spazio vettoriale di parole; questo termine è stato introdotto da Hinrich Schütze (Schütze [1997]), il cui lavoro presso la Stanford University ha spianato la strada verso molti dei successivi sviluppi. Il lavoro sviluppato in *Infomap* è una conseguenza diretta degli studi di Schütze.

Infomap costruisce, a partire da un corpus, una WORDSPACE in cui le parole sono rappresentate da vettori; le componenti di questi vettori codificano alcune informazioni sulla distribuzione delle relative parole nel corpus. Dati sperimentali indicano che le parole i cui vettori associati sono simili – più precisamente *vicini* in termini geometrici – spesso hanno significati simili o correlati: pertanto la WORDSPACE costruita da *Infomap* può essere usata per modellare similitudini tra parole diverse attraverso il confronto dei rispettivi significati.

L'algoritmo consiste nel costruire delle *matrici di cooccorrenza*, concentrare l'informazione *riducendo il numero di dimensioni*, *confrontare i vettori* usando la *somiglianza al coseno* (o *cosine similarity*), il tutto con l'aggiunta di *operazioni logiche* che estendono il linguaggio di interrogazione.

3.1 Matrici di cooccorrenza

Molti algoritmi per il recupero delle informazioni iniziano costruendo una *matrice termine/documento*, che hanno una colonna per ogni documento e in cui le righe corrispondono ai termini: ogni cella contiene il numero di occorrenze di un determinato termine in un documento. In questo modo ad ogni termine corrisponde un vettore; tali vettori possono essere visti come le coordinate di un termine in uno spazio vettoriale.

Nel caso di *Infomap*, diversamente da quanto accade negli *information retrieval systems*, si vogliono studiare proprietà tra parole che risiedono nello stesso documento. Pertanto una matrice termine/documento (o parola/documento) non è del tutto adatta allo scopo; ad esempio, nello stesso articolo che tratta di musica si potranno avere diverse occorrenze di parole come "brano", "canzone", "pezzo", ... che dovrebbero risultare in qualche modo associate (o associabili) pur provenendo tutte dalla stessa fonte. A tal fine *Infomap* procede scegliendo un certo numero di parole "cardinali", "ricche di significato" o *target*, e assegnando in seguito le coordinate di tutte le altre parole in funzione di quelle cardinali; in un certo senso questa scelta

(*estremamente* critica) di "parole riferimento" ha il significato geometrico di scegliere gli assi dello spazio vettoriale.

Il numero di dimensioni può essere un altro fattore critico: dati sperimentali hanno mostrato buoni comportamenti prendendo 1000 come numero di dimensioni dello spazio vettoriale. In questo modo *Infomap* è in grado di costruire una grande *matrice di cooccorrenze* in cui le colonne rappresentano le parole maggiormente "ricche di significato", mentre ad ogni riga corrisponde una parola del corpus con associato il numero di volte in cui la parola è testualmente vicina a ciascuna di quelle cardinali.

Gli autori non considerano esplicitamente il problema della pesatura dei contesti, limitandosi a prendere il *calcolo della frequenza relativa* come funzione di peso. Metodi alternativi prevedono di prendere il logaritmo della frequenza, per smussare le differenze delle alte frequenze, o ancora di assegnare maggior peso ai contesti che sono più significativamente associati alle *parole target*. Altri metodi sono basati sulla teoria dell'informazione, come la *Mutual Information* e la *Log-Likelihood Ratio* (vedi Manning and Schütze [1999]).

3.2 Ridurre le dimensioni

Il numero di dimensioni dello spazio vettoriale così ottenuto è spesso troppo elevato: il problema sta nell'avere troppo spazio a disposizione in cui muoversi, il che porta ad avere una rappresentazione troppo sparsa dell'informazione. Ci sono vari modi di ridurre il numero di dimensioni: la tecnica usata in *Infomap* è nota come *decomposizione dei valori singolari* (Trefethen and Bau [1997]), o anche *analisi della semantica latente*.

Dall'algebra lineare sappiamo come trovare una matrice ridotta con colonne ortogonali tra loro tale da preservare la maggior parte della varianza tra i vettori della matrice originale. L'idea è di cercare una nuova variabile latente in grado di riassumere il significato di due (o più) parole correlata. Ad ogni parola è assegnata una nuova coordinata relativa all'asse latente così trovato, proiettando una perpendicolare tra la sua posizione originale e il nuovo asse, cioè *proiettando* la parola sul nuovo asse (vedi figura 1).

Il numero di dimensioni a cui si può ridurre è un altro parametro dell'algoritmo che può essere alterato: alcuni ricercatori hanno ottenuto buoni risultati con 100 dimensioni, altri sostengono che valori tra 200 e 300 funzionino meglio. Probabilmente è ragionevole assumere che il risultato migliore sia (almeno in parte) determinato dal task da eseguire, e che la domanda "quante dimensioni servono per rappresentare il significato?" possa avere risposte diverse in situazioni diverse.

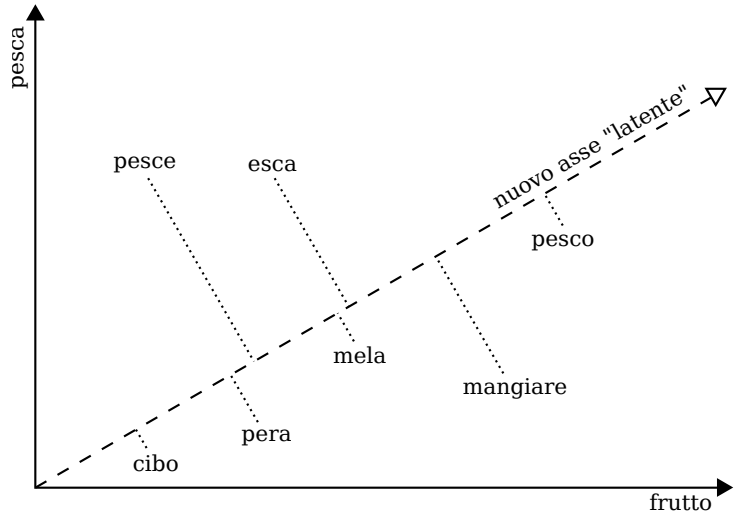

Figura 1: Proiezione delle parole inerenti *pesca* e *frutto* su un nuovo asse che combina queste due variabili

3.3 Confrontare i vettori

Ora che ogni parola è rappresentata con un vettore di una dimensione adeguata, è possibile iniziare a confrontare le parole tra loro per capire se sono simili o differenti. Un metodo standard per trovare il grado di similitudine tra due vettori è la misura di somiglianza al coseno. Definiamo il prodotto scalare tra due vettori $a = (a_1, \ldots, a_n)$ e $b = (b_1, \ldots, b_n)$ come segue:

$$a \cdot b = a_1 b_1 + \cdots + a_n b_n$$

La *norma* di un vettore a, indicata con $\|a\|$ è la distanza euclidea del vettore dall'origine dello spazio vettoriale, cioè:

$$\|a\| = \sqrt{a_1^2 \cdots a_n^2}$$

ora, dividendo il prodotto scalare $a \cdot b$ per il prodotto tra le norme $\|a\|$ e $\|b\|$ otteniamo il coseno dell'angolo compreso tra i due vettori, chiamato appunto somiglianza al coseno:

$$\cos(a, b) = \frac{a \cdot b}{\|a\|\|b\|}$$

o, in alternativa, considerando i vettori normalizzati $\hat{a} = a/\|a\|$ e $\hat{b} = b/\|a\|$ otteniamo $\cos(a, b) = \hat{a} \cdot \hat{b}$.

Con questo semplice calcolo possiamo trovare i "vicini più prossimi" di una parola data (quelli con somiglianza al coseno più alta) usando il programma `associate` del pacchetto *Infomap*.

Uno dei maggiori benefici del formalismo vettoriale è che permette di comporre più parole in frasi semplicemente sommandone i corrispettivi vettori. Di conseguenza per trovare il vicino più prossimo a *due* parole a e b in un corpus C è sufficiente calcolare:

$$\text{argmax}_{c \in C} \cos(a + b, c)$$

3.4 Connettivi quantistici

La WORDSPACE che abbiamo fin qui descritto offre strumenti di interrogazione poco soddisfacenti: le sole operazioni sui vettori che abbiamo introdotto sono l'addizione ed il prodotto scalare, ambedue commutative (mentre nel linguaggio naturale le frasi "Maria cerca Antonio" e "Antonio cerca Maria" hanno significati completamente differenti). A parte questo problema, il fatto è che abbiamo solo un modo per formare interrogazioni composte, e cioè giustapponendo le parole; sarebbe interessante introdurre degli operatori aggiuntivi.

La prima cosa potrebbe venirci in mente quando ci poniamo questo problema, è di usare l'algebra booleana: ad esempio, una query del tipo "a OR b" si potrebbe interpretare come "trova tutti le parole con significati simili ad a, poi quelle con significati simili a b e restituisci l'unione dei due"; o ancora, l'operatore "NOT a" potrebbe restituire solo le parole che hanno *minor somiglianza semantica* con a.

Tuttavia questo non è l'approccio seguito in *Infomap*, in cui gli autori hanno preferito porsi in un contesto differente; il problema sorge con l'uso di parole ambigue: come fare per discriminare tra significati diversi di parole ambigue? Come si può *rimuovere* da una interrogazione una possibile interpretazione semantica, preservando le altre? A questa domanda ha almeno due possibili risposte: quella booleana (nello spirito delle osservazioni di cui sopra) è introdurre un operatore "a NOT b" che prende tutti i vicini di a che non sono anche vicini di b; ma esiste anche un'alternativa, derivante dalla *quantum logic*.

L'interpretazione quantistica è uno strumento utile per descrivere il modo in cui una particella può essere rappresentata da una combinazione di possibili stati puri, così come una parola ambigua può essere rappresentata da una combinazione di "significati puri". In questo modo è possibile introdurre una forma di negazione "a NOT b" definita direttamente sui vettori come segue:

$$a \text{ NOT } b = a - (a \cdot b)b$$

è facile provare che il vettore a NOT b ha somiglianza al coseno pari a zero col vettore b (cioè per definizione è *ortogonale* al vettore b), infatti:

$$\begin{aligned} cos(a \text{ NOT } b, b) &= cos(a - (a \cdot b)b, b) \\ &= (\hat{a} - (\hat{a} \cdot \hat{b})\hat{b}) \cdot \hat{b} \\ &= \hat{a} \cdot \hat{b} - (\hat{a} \cdot \hat{b})(\hat{b} \cdot \hat{b}) \end{aligned}$$

che è uguale a zero, poichè per ogni vettore normalizzato si ha $\hat{b} \cdot \hat{b} = 1$. Pertanto la regione della WORDSPACE corrispondente all'espressione a NOT b è il sottospazio dei punti ortogonali a b, ossia i punti che non hanno significati in comune con b.

Quest'approccio costituisce una variante interessante alla tradizionale negazione booleana, principalmente per il fatto che lavorando direttamente sui vettori (cioè sulla rappresentazione del significato) è possibile rimuovere più sinonimi della parola non voluta; dati sperimentali mostrano che la negazione vettoriale restituisce il 20% in meno di sinonimi della parola non voluta rispetto alla versione booleana.

In maniera analoga, è possibile prendere il piano esteso dai vettori a e b che corrisponde ad un'espressione "a OR B" e che è consistente con l'idea di negazione via ortogonalità. *Infomap* attualmente implementa gli operatori di negazione vettoriale e di disgiunzione negata:

$$a \text{ NOT } (b_1 \text{ OR } b_2 \text{ OR } \cdots \text{ OR } b_n)$$

che per ragioni computazionali è molto più trattabile della disgiunzione positiva (questa query si scrive in *Infomap* proprio nella forma "a NOT b_1 OR $\ldots$ OR b_n").

Questi operatori logici furono originariamente introdotti da Birkhoff e von Neumann negli anni '30 per descrivere la logica di un sistema di meccanica quantistica, ed ecco perché sono chiamati *connettivi quantistici* (l'intero sistema prende il nome di *logica quantistica*).

4 Evoluzione dei DSM

Concludiamo con un sintetico riassunto dell'evoluzione dei *DSM* a partire dai *document retrieval systems* di metà anni '90:

Latent Semantic Analysis (Landauer & Dumais 1996)	
contesti	documenti
matrice	parole × documenti
W	logaritmo della frequenza e entropia delle parole nel corpus
d	decomposizione dei valori singolari (SVD)
S	somiglianza al coseno

Hyperspace Analogue to Language (Lund & Burgess 1996)	
contesti	window-based, triangolare con posizioni come funzioni di tipo per i contesti
matrice	parole × parole
W	frequenza
d	dimensioni con la maggior varianza
S	Metrica di Minkowski

Random Indexing (Karlgren & Salhgren 2001)	
contesti	window-based, rettangolare
matrice	parole × parole
W	vari
d	Information Retrieval
S	vari

Infomap NLP (Widdows 2004)	
contesti	window-based, rettangolare
matrice	parole × parole
W	frequenza
d	SVD
S	somiglianza al coseno

Dependency Vectors (Pado & Lapata 2007)	
contesti	dependency-based, con dipendenze come filtri per i contesti
matrice	parole × parole
W	rapporto log-Likelihood
d	nessuna
S	misura di similitudine da teoria dell'informazione in Lin (1998)

Distributional Memory (Baroni & Lenci 2009)	
contesti	dependency-based, con dipendenze come funzioni di tipo per i contesti
matrici	varie
W	Mutual Information locale
d	nessuna
S	somiglianza al coseno

Riferimenti bibliografici

Gärdenfors, P. 2004. *Conceptual spaces: The geometry of thought*. The MIT Press.

Manning, C. D. and Schütze, H. 1999. *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge, MA.

Schütze, H. 1997. *Ambiguity Resolution in Language Learning*.

Schütze, H. 1998. Automatic word sense discrimination. *Computational Linguistics*, 24(1):97–123.

Trefethen, L. N. and Bau, D. 1997. *Numerical Linear Algebra*. SIAM: Society for Industrial and Applied Mathematics.

Widdows, D. 2004. *Geometry and Meaning*. Center for the Study of Language and Information/SRI.