Linguistica Applicata 2007–2008

Lezioni 8/9: Collocazioni e Keywords, Espressioni regolari, Preparazione della ricerca per la tesina

Tutor: Dott. Emiliano Guevara[1]

[1] Facoltà di Lingue e Letterature Straniere
Università degli studi di Bologna

27–29 Febbraio 2008

Scaletta

Riepilogo

Esplorazioni di base: le collocazioni

Esplorazioni di base: keywords

Le espressioni regolari

Esplorazioni di base:

- Abbiamo 4 strumenti di base per esplorare le proprietà del linguaggio in un corpus:
  - Liste di frequenza
  - Concordanze (KWIC)
  - Collocazioni (di un singolo termine)
  - Parole chiave – *Keywords* (di un intero corpus)
- Le prime due le possiamo fare con tutti i programmi *stand alone* che abbiamo
- Le collocazioni e le keywords le possiamo estrarre con **AntConc** e con **WsTools**
- Oggi ci occupiamo brevemente di collocazioni e keywords su **AntConc**

Esplorazioni di base: le collocazioni

- Il termine collocazione (En. *collocation*) è una di quelle parole che in Linguistica Applicata tutti usano, di cui tutti sanno qualcosa, di cui tutti discutono, ma che nessuno sa definire in modo chiaro.
  - WsTools ha tre o quattro modi diversi per arrivare alle collocazioni di una parola!
- Non è un concetto condiviso largamente né definito con precisione
- La definizione più vecchia che c'è (Firth 1957:181):
  - Collocations of a given word are statements of the habitual or customary places of that word
- Non è molto chiaro, ma lascia vedere è che la nozione di collocazione ha a che fare con il contesto-distribuzione di una parola.
- Questo è un concetto che non si trova *quasi mai* nella linguistica strutturale (Saussurre) né in quella generativa (Chomsky)

Esplorazioni di base: le collocazioni

- In linguistica dei corpora, una **collocazione** è definita come *due o più di parole che co-occorrono* entro una finestra variabile di testo con una *frequenza maggiore di quanto ci aspettiamo* (data l'ipotesi che siano legate casualmente) (Manning & Schütze 1999, cap. 5)
- Esiste una serie di fenomeni molti diversi che corrispondono a questa definizione:
  1. combinazioni tipiche di parole (*caffè forte* vs. ?*caffè potente*)
  2. espressioni idiomatiche variabili (*domani **tiro** le cuoia*, *aveva quasi **tirato** le cuoia*)
  3. espressioni idiomatiche fisse (*nella morsa del freddo*, *è severamente vietato*)
  4. nomi propri e titoli vari (*il presidente della repubblica*, *la farnesina*), ecc.
- Le prime sono quelle più interessanti: trasparenti, produttive. Sono anche quelle più difficili da trovare

Esplorazioni di base: le collocazioni

- Per trovare le collocazioni di una parola, serve prima estrarre la sua **concordanza** (tipicamente con un contesto 5L–5R o inferiore), e poi bisogna contare quante volte compare ogni termine diverso
- Si applicano poi delle formule statistiche per determinare **l'ordine di associazione** di ogni termine con la parola ricercata (coppia di parole **word–collocate**).
- Le tecniche più usate:
  - **Mutual Information** (*bias* verso le coppie poco frequenti)
  - **T-Score** (*bias* verso le coppie molto frequenti)
  - **Log-Likelihood** (evita il *bias* legato alla frequenza)
- AntConc cerca le collocazioni usando Mutual Information o T-Score
- Il nostro sito per i corpora le usa *tutte e tre*

Esplorazioni di base: le collocazioni

- Quindi, due tipi di collocazione:
  - n-grammi (contigui) che compaiono molto frequentemente ma, attenzione, non tutti i bigrammi/trigrammi frequenti sono collocazioni (cfr. Lenci etal. 2005: p.198-199)

    es. *pesce fresco*, *notte fonda*, *benzina verde*, *New York*, *Osama bin Laden*, ecc.

  - gruppi di parole che, pur non comparendo contigui, hanno forte tendenza a comparire *vicino*

Esplorazioni di base: keywords

- Le parole chiave sono invece i termini che:
  - compaiono con **frequenza maggiore alle attese**, caratterizzando fortemente un dato corpus
  - . . . ma anche in senso negativo, le parole che compaiono con **frequenza minore alle attese**
- Per estrarre le keywords di un corpus serve innanzitutto determinare le nostre attese:
  - si confronta un **corpus specialistico** (piccolo) con un **corpus di riferimento** (grande)
  - AntConc confronta i corpora *per intero*
  - WsTools confronta soltanto le *liste di frequenza* di ogni corpus

Esplorazioni di base: keywords

- Le statistiche più usate per computare le keywords:
  - **Log-Likelihood** (più affidabile)
  - **Chi-quadro** (**Chi-Squared**) (meno affidabile)
- AntConc e WsTools possono usare entrambe queste misure
- Il nostro sito ancora non lo fa...

Le espressioni regolari: REGEX (regular expressions)

- Spesso dobbiamo eseguire ricerche sul testo che non corrispondono esattamente ad una singola parola, p. es.:
  - trova tutte le occorrenze di tutte le voci del verbo *mangiare* (mangio, mangi, mangerebbero, mangiare, ecc.)
  - trova tutte le parole che finiscono per *–ione*
  - trova tutte le parole che iniziano per *intro–*
  - trova tutte le occorrenze di *Casa delle Libertà* e sostituisci con *Popolo delle Libertà*
- Quello che facciamo normalmente è inserire nelle nostre stringhe di ricerca dei **caratteri jolly** (o **wildcards**)
  - `/cas.*/ /mang.*/ /bambin[aeoi]/ /inizio?/ /H(ä|ae?)ndel/`
- Le espressioni regolari o *regex* sono una notazione algebrica per definire in modo formale e rigoroso questi pattern di stringhe
- Le regex sono state create dal logico S. Kleene nel 1956 (ed è per quello che il simbolo * si chiama "Kleene star")

REGEX

- Esistono molti dialetti diversi di espressioni regolari, ma quasi tutti si assomigliano per la sintassi di base
  - Ogni programma usa un tipo in particolare
  - Anche M$ Word supporta un uso limitato delle regex
  - Il dialetto che useremo di più è quello derivato dal linguaggio di programmazione Perl: PCRE – Perl Compatible Regular Expressions
  - Le regex servono a fare moltissime cose, ad esempio la *tokenizzazione* di un testo
- Tipicamente, una regex opera su una singola linea (riga) di testo, delimitata dal carattere(i) di nuova linea LF, CR, LN/CR

REGEX: caratteri

- Ogni carattere *matches* sè stesso:
  - `a`, `z`, `5`, `o|i`, `abbastanza`, ecc.
- Ci sono però i **caratteri jolly** (o **wildcards**) che non matchano sé stessi, ma che hanno la funzione di essere *molteplicatori*:
  - `?` 0 ò 1 occorrenza del carattere precedente
  - `*` 0 ò più occorrenze del carattere precedente
  - `+` 1 ò più occorrenze del carattere precedente
  - `obb?iett*iv(o|i)` matcha *obiettivo*, *obiettivi*, *obbiettivo*, *obbiettivi*, *obietttivo*, *obbiettttttttttivi*, ecc.
- Classi di caratteri:
  - `[la]` *l* ò *a*
  - `[a-z]` una lettera minuscola
  - `[A-Z]` una lettera maiuscola
  - `[a-zA-Z0-9]` una lettera minuscola, maiuscola, oppure una cifra numerica
  - `[^a-zA-Z0-9]` qualsiasi carattere tranne quelli nella classe

REGEX: caratteri speciali

- Altri caratteri con significato speciale:
  - `.` matcha qualsiasi carattere (tranne la nuova linea)
  - `^` indica l'inizio della riga
  - `$` indica la fine della riga
  - `[^...]` in una classe, indica la negazione della stessa
- Se si vuole usare uno dei caratteri speciali in modo letterale, bisogna "scapparli" con un backslash \:
  - `\?` il segno di domanda
  - `\.` il punto
  - `\+` il segno +
  - `\*` l'asterisco *
  - `\[ \] \( \) \{ \} ecc.`

REGEX: caratteri speciali

- Caratteri speciali combinati con backslash \:
  - `\d` qualsiasi cifra numerica
  - `\D` il complemento di `\d`, caratteri non numerici
  - `\w` caratteri alfanumerici ò underscore _
  - `\W` il complemento di `\w`
  - `\s` qualsiasi carattere di spaziatura (spazio, tab, linea)
  - `\S` il complemento di `\s`

  - `\b` confine di token/parola
  - `\B` il complemento di `\b`

  - `\n` nuova linea (LF)
  - `\r` ritorno di carrello (CR)
  - `\t` tabulazione

Esempi: che cosa matchano?

- `\d+`
- `\d+\.\d+`
- `b.*s`
- `[A-Z][a-z]+\s+\d\d\d\d`
- `c[aeiou]n(e|i)`
- `^\w+\t \w+ \t \d+$`
- `[a-zA-Z0-9_-\.]+@[a-zA-Z0-9-]+\.[a-zA-Z]{0,4}`