

**25. Indici.***Per la consultazione del volume.*

25.0 **PREMESSA.** In un'opera delle dimensioni e respiro che la presente ha assunto è indispensabile che la dotazione di indici sia generosa, anche a costo di aumentarne la già esorbitante mole, per permettere non solo al lettore di orizzontarvisi senza troppo sforzo, ma anche perché egli lo possa usare come comodo (?) strumento di riferimento.

Alla prima funzione assolve soprattutto l'indice analitico (§ 25.5) ed alla seconda l'indice dettagliato (§ 25.4); a questi due più fondamentali indici sono fatti precedere quello delle script impiegate nell'elaborazione del CT (§ 25.1), quello degli autori citati (§ 25.2) e dei passi esemplificati (§ 25.3).

A qualcosa, per ragioni di spazio, si è dovuto però rinunciare: così ad un pianificato indice delle query, e ad un indice delle abbreviazioni, che speriamo il lettore trovi abbastanza autoevidenti, essendoci contenuti al realmente necessario (ed il volume è per sua natura privo delle mille sigle di dialettonimi e toponimi che ne sono la primaria fonte), evitando di abbandonarci alla frenesia di sigle tecniche cui certa linguistica, con compiacimento tra il masturbatorio e l'illusionistico, spesso per compensare l'assenza di effettivi contenuti referenziali, ama indulgere.

25.1 **INDICE DEGLI SCRIPT.** [MT] Si danno di ogni script creato per la fabbrica del CT una descrizione sintetica, completa di indicazioni di che materiali richieda in ingresso, dell'autore (le sigle sono le consuete. cfr. § 1.3, cui si aggiunga CeO 'Cesare Oitana'), e di eventuale rimando al paragrafo dove ve ne sia una più significativa trattazione.

`Aufbereitung_new.perl` Esegue il primo passaggio per la codifica in CQP del corpus disambiguato. *Input:* corpus disambiguato. Autore: AF. Cfr. § 20.1.

`check=dirtydos.awk` Aiuta a individuare caratteri anomali all'interno del testo DOS. *Input:* testo taggato. Autore: CeO.

`checkline.awk` Verifica la congruenza dei dati nel corpus segnalando le eventuali irregolarità. *Input:* testo disambiguato. Autore: MT. Cfr. § 7.5.1.5.

`checksum.awk` Computa il numero complessivo di forme. *Input:* testo taggato o disambiguato. Autore: MT. Cfr. § 7.5.1.4.

`conto.awk` Computa le occorrenze del carattere specificato nel programma. *Input:* testo taggato. Autore: CeO.

`count_Toktype.awk` Computa i dati ricavati da `estra_Toktype`. *Input:* file di uscita di `estra_Toktype`. Autore: MT. Cfr. § 7.5.1.7.

`dis_end.awk` Formatta il testo disambiguato secondo le caratteristiche del testo di origine. *Input:* file di uscita di `modulo6`. Autore: MT. Cfr. § 9.1.1.

`dis_prep.awk` Prepara il testo per le operazioni di disambiguazione. *Input:* file di uscita di `spalmaF2`. Autore: MT. Cfr. § 9.1.1.

`estra_Toktype.awk` Estrae i lemmi riferiti ai type presenti nel corpus. *Input:* corpus disambiguato. Autore: MT. Cfr. § 7.5.1.7.

`estraF.awk` Estrae le frequenze d'uso delle forme presenti nel corpus. *Input:* corpus disambiguato. Autore: CeO. Cfr. § 7.5.1.2.

`estraL.awk` Estrae i lemmi dal testo disambiguato. *Input:* corpus disambiguato. Autore: CeO.

`fix_uncinate` Verifica e corregge eventuali anomalie nelle parentesi uncinate. *Input:* file di uscita di `Aufbereitung_new`. Autore: MT. Cfr. § 20.2.2.

- `horiz.awk` Riporta un testo convertito da `vertic` nel formato originario. *Input*: file di uscita di `vertic`. Autore: MT. Cfr. § 7.6.2.
- `lineaMW.awk` Formatta il formario-MW consentendo di ricavare il lemmario-MW. *Input*: file di multiword. Autore: CeO.
- `modulo1.awk` Primo modulo di disambiguazione. *Input*: file di uscita di `dis_prep`. Autore: MT. Cfr. § 9.1.
- `modulo2.awk` Secondo modulo di disambiguazione. *Input*: file di uscita di `modulo1`. Autore: MT. Cfr. § 9.1.
- `modulo3.awk` Terzo modulo di disambiguazione. *Input*: file di uscita di `modulo2`. Autore: MT. Cfr. § 9.1.
- `modulo4.awk` Quarto modulo di disambiguazione. *Input*: file di uscita di `modulo3`. Autore: MT. Cfr. § 9.1.
- `modulo5.awk` Quinto modulo di disambiguazione. *Input*: file di uscita di `modulo4`. Autore: MT. Cfr. § 9.1.
- `modulo6.awk` Sesto modulo di disambiguazione. *Input*: file di uscita di `modulo5`. Autore: MT. Cfr. § 9.1.
- `mwl-hinzu.perl` Permette una efficace codifica in CQP delle multiword. *Input*: file di multiword e file di uscita di `Fix_uncinate`. Autore: AF. Cfr. § 20.1.
- `ovi-ct_tm.awk` Automatizza la catena di sostituzioni necessaria per il passaggio dal formato Gatto al formato *Corpus Taurinense*. Autore: MT. *Input*: testo in formato Gatto. Cfr. § 8.9.1.1.
- `pesca50.awk` Estrae le congiunzioni coordinate (50). *Input*: testo taggato. Autore: CeO.
- `pescaCL.awk` Estrae le forme che contengono il carattere di *backslash* `<\>`. *Input*: testo in formato Gatto. Autore: CeO. Cfr. § 8.8.
- `pescaSi.awk` Estrae le forme pronominali. *Input*: testo taggato. Autore: CeO.
- `pex_d'.awk` Simula le regole di disambiguazione per la forma *d'*. *Input*: file di uscita di `modulo6`. Autore: MT. Cfr. § 9.5.1.
- `pex_questi.awk` Simula le regole di disambiguazione per la forma *questi*. *Input*: file di uscita di `modulo2`. Autore: MT. Cfr. § 9.5.1.
- `sdoppia.awk` Segnala le righe doppie consecutive in un testo ordinato alfabeticamente. Autore: CeO. *Input*: file di uscita di `vertic`.
- `separa.awk` Si occupa di separare i grafoclitici di un testo. Richiede il file di riferimento `conencl`. *Input*: testo in formato Gatto. Autore: CeO. Cfr. § 8.8.
- `smista.awk` Segnala l'assenza del campo lemma in F2. Autore: CeO. *Input*: F2.
- `smista_.awk` Segnala in F2 (*effedue*) il campo lemma presente, ma non ancora valorizzato. *Input*: F2. Autore: CeO.
- `somma.awk` Calcola la somma delle occorrenze del formario. *Input*: F2. Autore: CeO.
- `spalmaF2.awk` Assegna le etichette con i valori grammaticali a tutte le forme del corpus. *Input*: testo del *Corpus Taurinense* in formato `ascii`. Autore: CeO. Cfr. § 7.5.1.1.
- `subs.awk` Effettua alcune sostituzioni di caratteri per il passaggio dal formato Gatto al formato *Corpus Taurinense*. *Input*: testo in formato Gatto. Autore: CeO. Cfr. § 8.5.2.2.
- `transord_1.awk` Riformatta i dati di frequenza dei token sulla base delle transcategorizzazioni - Modulo 1. *Input*: F (*effeliscio*). Autore: MT. Cfr. § 7.5.1.6.

`transord_2.awk` Riformatta i dati di frequenza dei token sulla base delle transcategorizzazioni - Modulo 2. *Input*: file di uscita del `transord_1`. Autore: MT. Cfr. § 7.5.1.6.

`transord_3.awk` Riformatta i dati di frequenza dei token sulla base delle transcategorizzazioni - Modulo 3. *Input*: file di uscita del `transord_2`. Autore: MT. Cfr. § 7.5.1.6..

`verif4` Effettua verifiche formali sulle etichette all'interno delle righe. *Input*: testo taggato. Autore: CeO.

`vertic.awk` Converte un testo taggato in layout orizzontale formato "word\_tag" in layout verticale con `word` e `tag` su due colonne. *Input*: testo disambiguato. Autore: MB. Cfr. § 7.6.2.

25.2 INDICE ONOMASTICO. [MCo] I numeri di pagina preceduti dalla diesis <#> rimandano ai riferimenti bibliografici; quelli dal più <+> a quelli in citazioni; quelli dall'asterisco <\*> ai contenuti in nota; quelli dalla sterlina <£> ai contenuti nel glossario; quelli, infine, sottolineati presentano citazioni esplicite.

Come d'uso, la virgola è l'operatore di inversione; i "de" ecc. non sono alfabetizzati in spagnolo e francese, ma lo sono in italiano, e così i "van" non sono alfabetizzati in tedesco ma lo sono in olandese; cognome e nome non sono separati dalla virgola in lingue (e.g. ungherese e giapponese) dove costituiscono l'ordine normale (e pertanto non richiedono un operatore di inversione); i nomi storici (e.g. Dante Alighieri, Richart de Barbezilh) sono alfabetizzati sotto il nome anziché il cognome. Non si è indicizzato il nome dell'autore.

Aarts, Jaan #1112  
 Abney, Steven \*43, \*43, #1112  
 Adeodato 1150  
 Adjukiewicz, Kazimierz \*410, #1112  
 Aebischer, Paul \*670, #1112  
 Affini, Guido 6  
 Agostini, Francesco \*271, \*274, 346, 408, 408,  
 421, 421, 423, 424, 424, \*424, 427, 431, 431,  
 437, \*641, #1112  
 Aho, Alfred V. #1112, £1062, £1063  
 Aijmer, Karin #1112  
 Ainsworth, Geoffrey Clough \*25, #1112  
 Aitchinson, Jean #1120  
 Alberti, Giovanni #1130  
 Alessio, Giovanni #1118  
 Alfonzetti, Giovanna 267, 315, #1112  
 Alighieri, Dante \*33, @39, 133, 215, \*215,  
 216, 240, +253, \*274, \*308, \*308, +324, 430,  
 +431, 527, \*536, 565, 649, 651, 652, 666,  
 672, 713, +815, +844, \*847, 1030, \*1033,  
 1034, 1040, \*1040, 1041, \*1043, 1045,  
 #1121, #1132, 1171  
 - dantesco 359, 423, \*540  
 Alisova, Tatjana #1112  
 Allegranza, Valerio \*46, \*51, #1112  
 Allora, Adriano 5, #1113, £1081  
 Altemberg, Bengt #1112  
 Ambrosini, Riccardo 249, 251, \*251, 688,  
 #1113  
 Anderson, Susan #1146  
 Andorno, Cecilia 196, 524, \*525, 527, \*527,  
 #1113  
 Andreose, Alvise #1160  
 Angulo, Jaime de \*0  
 Antinucci, Francesco #1120, #1125  
 Antony, Louise M. , #1113  
 Anttila, Arto #1143  
 Apollonio Alessandrino 675, #1113  
 Aristarco di Samotracia £1074  
 Aristotele di Stagira £1057, £1058, #1113  
 - aristotelico #1127, #1168  
 Armstrong, Susan #1113  
 Ascoli, Graziadio Ascoli #1114  
 Aston, Guy #1117  
 Atwell, Eric #1114, #1166  
 Auerbach, Erich #1114, £1071  
 Austin, John Langshaw #1114, £1069, £1070  
 Avalu, D'Arco Silvio 1074, 1075, 1076, 1077,  
 1084, 1092, 1102, #1114, #1121, #1185,  
£1074, £1075, £1076, £1077, £1084, £1092,  
£1102  
 Avalu, Giuseppe #1181  
 Avins, Styra 23, #1114  
 Baker, Paul #1114, £1086  
 Bailly, Charles #1114, £1071, #1164