Indice analitico della Costituzione italiana

effettuato con gli strumenti dell'informatica libera[1]

Alberto Cammozzo[2]
il 2 gennaio 2006

Il sistema GNU/Linux, come il suo predecessore Unix, è ricchissimo di programmi destinati all'elaborazione di testi. Questi, data la vocazione tecnica di Unix/Linux sono nati soprattutto per la manipolazione di quella particolare categoria di testi che sono i sorgenti di programmi. Tra codice sorgente e codice legale vi sono forti affinità, per scopi, metodi, possibilità di errori e altro. Perciò non deve sorprendere se gli strumenti che vanno bene per l'uno possano andare bene per l'altro. La Costituzione italiana può essere definita il nucleo, il *kernel*, delle norme che regolano ("fanno girare", più o meno bene) la nostra convivenza civile, e quello in cui più profondi sono i riferimenti alla libertà e ai diritti che tanto cari sono alla comunità del software libero. Va anche detto che il testo della Costituzione, anche se non liberamente modificabile, è libero da obblighi derivanti dal copyright.
Sul testo costituzionale si sono usati gli strumenti in dotazione standard di un sistema GNU/Linux per generarne l'indice analitico, prendendo in esame in crescente dettaglio i passi compiuti e i programmi impiegati.

L'indice analitico[3]

L'indice analitico riporta ogni occorrenza di una parola chiave all'interno del testo indicizzato. Il sistema GNU/LINUX fornisce tra le *utility* di base un programma per generare l'indice analitico che si chiama "ptx" (*permuted index*).
Il programma ptx riporta nel suo output il contesto in cui si trova ogni singola parola, ordinato alfabeticamente secondo la parola chiave. La Costituzione conta circa 10.000 parole, con un lessico di circa 2300 vocaboli. Con un elenco di parole chiave di 2000 elementi il programma genera un indice di più di 5700 righe.

L'output di ptx è costituito da due colonne:
1. il **numero** dell'articolo della costituzione in cui compare il testo (oppure la stringa "DTF" per le disposizioni transitorie e finali),
2. il **testo**, limitato alle parole più prossime alla parola chiave, con al centro la parola indicizzata, preceduta da tre spazi. Eventuali troncature della frase sono rappresentate dal segno "/"

1 Questo documento puo' essere liberamente copiato e diffuso nella sua interezza, a condizione che questa nota venga riprodotta. Reperibile in Internet ricercando la stringa "CostPTXmmzz".

2 mmzz -at- pluto.it. L'autore ringrazia il Dipartimento di Informatica per Non Informatici dell'Università "Immanuel Kant" della Gianozia Orientale (http://www.gianoziaorientale.it) per il determinante contributo nella riscoperta del programma "ptx" e per l'innovativa luce gettata sulle potenzialità dell'informatica per umanisti.

3 L'indice analitico completo della Costituzione italiana può essere consultato a questo indirizzo: http://homes.stat.unipd.it/mmzz/Costituzione/IndiceCostituzione.html

Nel caso la parola chiave sia all'inizio della frase, quest'ultima proseguirà nel lato sinistro, come è visibile dall'esempio qui appresso negli articoli 13 e 15. Analogamente potrà iniziare sul lato destro come ad es. negli artt. 10, 68 e 111.

Nell'indice analitico della Costituzione alla parola “libertà” si troveranno queste voci:

```
Art.3.          /, limitando di fatto la   libertà e l' eguaglianza dei/
Art.10.       effettivo esercizio delle    libertà democratiche/      /l'
Art.11.       /strumento di offesa alla    libertà degli altri popoli e/
Art.13.     inviolabile. Non è/      La     libertà personale è
Art.13.        /altra restrizione della   libertà personale, se non/
Art.13.               /a restrizioni di   libertà.  La legge/
Art.14.            /per la tutela della   libertà personale.  Gli/
Art.15.     della/                   La    libertà e la segretezza
Art.33.       /assicurare ad esse piena   libertà e ai loro alunni un/
Art.35.      /del lavoro.  Riconosce la   libertà di emigrazione,/
Art.41.           /alla sicurezza, alla   libertà, alla dignità umana./
Art.68.         altrimenti privato della  libertà personale, o/       /o
Art.111.          i provvedimenti sulla   libertà personale,/    /contro
```

Di seguito i testi degli artt. 11 e 10:

Art.10. L' ordinamento giuridico italiano si conforma alle norme del diritto internazionale generalmente riconosciute. La condizione giuridica dello straniero è regolata dalla legge in conformità delle norme e dei trattati internazionali. Lo straniero, al quale sia impedito nel suo paese l' effettivo esercizio delle libertà democratiche garantite dalla Costituzione italiana, ha diritto d'asilo nel territorio della Repubblica secondo le condizioni stabilite dalla legge. Non è ammessa l' estradizione dello straniero per reati politici.

Art.11. L' Italia ripudia la guerra come strumento di offesa alla libertà degli altri popoli e come mezzo di risoluzione delle controversie internazionali; consente, in condizioni di parità con gli altri Stati, alle limitazioni di sovranità necessarie ad un ordinamento che assicuri la pace e la giustizia fra le Nazioni; promuove e favorisce le organizzazioni internazionali rivolte a tale scopo.

Le fasi principali dell'elaborazione

Il testo della Costituzione viene presentato in una forma adatta alla lettura, perciò va adattato all'elaborazione automatica. Inoltre non tutte le parole sono utili per un indice analitico: lasciando nell'indice parole come “del”, “è” o “che” il numero delle righe dell'indice raddoppierebbe introducendo nel nostro caso più di 5000 righe completamente inutili. perciò anche l'indice delle parole chiave va adattato alle nostre esigenze.

Queste sono le fasi dell'elaborazione del testo originale:

1. salvataggio in formato testo della Costituzione, prelevato tramite un *browser* dal sito del Quirinale[4].
2. Adattamento manuale del testo: rimozione delle intestazioni, firme, titoli delle parti e simili. Separazione dei singoli articoli delle disposizioni transitorie e finali e loro etichettatura con il prefisso DTF. Questa operazione va fatta con un editor di testi, quali “EMACS” o “vi”.

4 http://www.quirinale.it/costituzione/costituzione.htm

3. Riformattazione automatica del testo in modo che ogni articolo occupi una e una sola linea del file, separazione delle parole dall'apostrofo che le precede, e infine correzione di alcuni errori nella trasposizione dell'apostrofo (dovuti probabilmente alla codifica del browser). Queste operazioni sono state effettuate tramite un solo comando composto da 8 elementi, che verranno descritti in dettaglio più avanti.
4. Estrazione automatica dal testo originale dell'elenco delle parole utili che lo costituiscono, ed esclusione automatica di quelle costituite da una sola o due lettere. L'operazione va monitorata per reintrodurre eventuali parole importanti che venissero escluse.
5. Manipolazione manuale dell'elenco appena creato per togliere altre parole indesiderate ("dall, dallo, dalle, dello, con, dei, degli, nelle, nei, non, che", ecc.) e per reintrodurre le parole eliminate nella fase precedente che invece si vogliono tenere (nel nostro caso la parola di due lettere “re”).
6. Creazione automatica dell'indice analitico basato sull'elenco di parole chiave identificate.

Le elaborazioni automatiche in dettaglio.

L'elaborazione è avvenuta su un sistema GNU/Linux Debian. I programmi usati vanno brevemente descritti per consentire la comprensione dei comandi che ne fanno uso:

1. **cat**[5]: apre un file e ne propone il contenuto per elaborazioni successive,
2. l'elemento “|”, detto *pipe*, ovvero “tubazione”, che raccorda una catena di comandi. In particolare collega l'output del programma alla propria sinistra con l'input del programma alla propria destra, esattamente come farebbe un pezzo di tubazione idraulica. E' l'elemento che rende possibile la concatenazione di comandi semplici per la costruzione di comandi complessi, detti appunto *pipe,*
3. **sed**[6]: (*stream editor,* o editor di flusso) apporta delle modifiche al proprio input, in particolare consente di effettuare delle sostituzioni di stringhe, ad esempio per trasformare le stringhe “Costituzione” in “Cost.” Per ottenere questo si scriverà: “sed s/Costituzione/Cost/”. Le due stringhe sono delimitate dal carattere “/” e il carattere “s” sta per sostituisci,
4. **tr**[7]: traduce un carattere, o un gruppo di caratteri, in altri caratteri. Ad esempio permette di trasformare tutte le minuscole in maiuscole, e viceversa. A differenza di sed, opera esclusivamente su caratteri,
5. **grep**: estrae dalla *pipe* o da un file le parole che combaciano con una certa srtinga o un certo schema (*pattern*). Ad esempio il comando “grep libertà costituzione.txt” estrarrà dal file *costituzione.txt* tutte le righe che contengono la parola “libertà”. E' uno dei comandi più utilizzati,
6. **sort**[8]: ordina in modo alfabetico o numerico il proprio input,
7. **uniq**[9]: elimina le righe duplicate presenti nel proprio input,
8. **ptx**[10]: il programma che crea l'indice analitico vero e proprio, dato un testo in input ed un indice (opzionale) di parole.

Ora verranno di seguito sommariamente illustrati i comandi e le funzioni svolta dalle loro singole

5 scritto da Torbjorn Granlund and Richard M. Stallman per la Free Software Foundation, comando originariamente presente anche su Unix.
6 Free Software Foundation, comando originariamente presente anche su Unix.
7 scritto da Jim Meyering per la Free Software Foundation, comando originariamente presente anche su Unix.
8 scritto da Mike Haertel and Paul Eggert per la Free Software Foundation, comando originariamente presente anche sul sistema Unix.
9 scritto da Richard Stallman and David MacKenzie per la Free Software Foundation, comando originariamente presente anche sul sistema Unix.
10 scritto da F. Pinard per la Free Software Foundation, comando originariamente presente anche su l sistema Unix.

parti, rimandando alle pagine di manuale *online*[11] le descrizioni dettagliate del funzionamento dei comandi.
Il punto 3 della sequenza precedentemente illustrata prevede la riformattazione del testo in modo che:

1. ogni articolo stia su una sola linea di testo,
2. vengano separate le parole dall'apostrofo che eventualmente le precede,
3. vengano corretti alcuni errori nella trasposizione dell'apostrofo.

Il comando per ottenere questo effetto è il seguente[12]:

```
cat costituzione.txt | tr "\n" " " | sed "s/DTF/\nDTF/g" |
  sed "s/Art. /\nArt./g" | sed "s/l'/l' /g" | sed "s/L'/L' /g"
  | tr "?" "'" > costituzione-prep
```

Apparentemente il comando è molto complesso, ma se scomposto nei vari elementi separati tra loro dalla "|" che li raccorda tra loro, risulta composto da elementi in se semplici. Separando ogni elemento della *pipe* ed analizzandone la funzione otteniamo:

`cat costituzione.txt |`	Introduce il file costituzione.txt nella *pipe*
`tr "\n" " " |`	trasforma i caratteri di fine riga (per convenzione `\n`) in spazi. In questo modo tutto il testo della costituzione sarà su una sola riga.
`sed "s/Art. /\nArt./g" |`	Trasforma le stringhe "Art. " in un a capo (\n) e "Art.", in questo modo spezza l'unica riga in una riga per articolo. Toglie anche lo spazio dopo il punto.
`sed s/DTF/\nDTF/g |`	Analogo a quanto fatto prima per "Art" relativamente alle disposizioni temporaneee: Trasforma la stringa "DTF" in "a capo + DTF". La stringa DTF è presente all'inizio degli articoli delle disposizioni finali.
`sed "s/l'/l' /g" |`	Introduce uno spazio tra la stringa l' e la parola che segue, per consentire che parole come "l'uguaglianza" siano indicizzate come "uguaglianza".
`sed "s/L'/L' /g" |`	Idem per le elle maiuscole,
`tr \? \'`	Sostituisce alcuni punti interrogativi che rappresentavano alcuni apostrofi per errore.
`> costituzione-prep`	salva il risultato di tutta l'elaborazione nel file `costituzione-prep`.

Si vede come da una sequenza di comandi elementari semplici si ottiene una sequenza complessa, grazie all'elemento di raccordo "|". Ora abbiamo un file con un articolo per riga, formattato correttamente, con tutte le parole separate.

11 Il comando "man sed" presenta il manuale di sed, "man tr" quello di tr, eccetera.

12 Vi sono soluzioni più compatte ed eleganti di questa, usando comandi più complessi. Si vedano ad esempio i potentissimi linguaggi "awk" e "perl".

Il successivo comando complesso, quello precedentemente esposto al punto 4 e'

```
cat costituzione-prep | tr "[ ']" "\n" <c | sed s/[.,:\;]//g | sort | uniq | grep -v "^.$" | grep -v "^..$" > parole
```

che, scomposto nei suoi elementi componenti, separati dalla *pipe* ("|"):

1. immette nel flusso il file *costituzione.prep*, risultato della precedente operazione,
2. traduce ogni spazio e ogni apice (“[']”) in un fine riga (“\n”), per spezzare tutto il testo in un elenco di parole, una per riga,
3. cancella ogni punteggiaura (i caratteri “.,:;” diventano “”, cioè una singa vuota),
4. ordina il lisultato ottenuto alfabeticamente,
5. elimina le righe adiacenti duplicate,
6. riversa l'esito della sequenza di comandi nel file “parole”.

Ora abbiamo, oltre al testo nel file *costituzione-prep*, anche un file *parole* che contiene tutte le parole presenti nel testo della costituzione, una per riga. Non resta che effettuare l'indicizzazione.

L'ultimo comando, “ptx”, serve per creare l'indice vero e proprio. Non fa uso di *pipe* ma solo dei due file preparati in precedenza: *parole* nel quale trova l'elenco delle parole-chiave, e *costituzione-prep* nel quale si trova il testo da indicizzare. L'output viene riversato nel file *indice*.

```
ptx -f -r -o parole costituzione-prep > indice
```

L'opzione -f ordina al programma ptx di elencare di seguito nel proprio output parole che iniziano con la stessa lettera sia essa maiuscola o minuscola. L'opzione -r serve a creare il riferimento al numero dell'articolo della Costituzione, che viene preso dalla prima parola di ogni riga del file *costituzione-prep*.

Conclusioni

Chi necessiti di effettuare un qualche trattamento o elaborazione di testi (che è cosa ben diversa dalla *scrittura* dei testi) può avvantaggiarsi con profitto degli strumenti approntati da programmatori di calcolatori abituati a trattare una foma particolare di testo, il codice sorgente, e a dover (spesso a malincuore) scrivere la documentazione per i loro programmi.
GNU/Linux eredita da Unix la flessibilità di strumenti elementari molto potenti, che possono essere combinati in *pipe* e applicati con successo anche ad ambiti ben diversi da quelli della programmazione, al costo di un modesto sforzo nell'interazione con la tastiera, ma con grande soddisfazione.

Bibliografia

1. B.Kernighan, R.Pike. *The Unix programming environment*. Prentice-Hall, 1984
2. Chiara Paci, Software libero per umanisti, edizioni del Dipartimento di Informatica per Non Informatici dell'Università “Immanuel Kant” della Gianozia Orientale, 2005 (http://www.gianoziaorientale.it/unikant/dini/sl_umanisti.pdf).