Appunti Di Information Retrieval

Massimo Meneghello

March 2018

Contents

1 Introduzione

Per poter reperire dei documenti è necessario prima di tutto poter rappresentare il loro contenuto informativo in modo conciso. Questo passo è reso possibile dall'**analisi del testo** che deve poter essere effettuata in modo **automatico**, **rapido**, **affidabile** e **consistente**.
Esistono due principali approcci all'analisi del testo:

- l'approccio statistico;
- l'approccio linguistico.

Il primo di questi approcci è stato quello più utilizzato e studiato fin dagli albori della disciplina. Le prime osservazioni riguardanti un'analisi statistica dei testi sono state proposte da *Hans Peter Luhn*, prima, e in seguito formalizzate da

George Kingsley Zipf.
Luhn si era reso conto infatti che:

- la distribuzione delle parole non è uniforme in un testo;
- poche parole compaiono molto di frequente;
- molte parole compaiono raramente.

La distribuzione è quindi **asimmetrica** - *skew distribution*. A Zipf si deve la formulazione di varie leggi empiriche che mettono in relazione la **frequenza** di una parola con la sua **forma** e il suo **significato**. La più nota è la legge che porta il suo nome, la **Legge di Zipf** appunto.

$$r \times f = costante \tag{1}$$

dove f è il valore della frequenza di una parola in un testo (o in un campione di testi) mentre r è il rango di quella parola dopo che tutte le parole sono state ordinate per frequenze decrescenti. Si ottiene quindi un andamento iperbolico della frequenza sul rango.
Si può anche pensare di riscrivere la legge di Zipf in termini probabilistici (in questo caso le frequenze assolute con cui una parola compare diventano le probabilità che una parola ha di comparire all'interno di un testo).
La scelta dei descrittori che permettono di **discriminare meglio il contenuto informativo di un testo** - *resolving power* - si basa sulle osservazioni di Luhn e sulla legge di Zipf.
Una volta fissate una soglia inferiore di cut-off - *lower cut-off* - e una soglia superiore di cut-off - *upper cut-off* - si considerano solamente i descrittori che hanno rango compreso tra queste due soglie. Operando in questo modo si escludono:

- i descrittori che hanno frequenze troppo elevate, chiamati anche *stop words*, che portano poca informazioni sul contenuto dei singoli documenti;
- i descrittori che hanno frequenze troppo basse, che invece costituiscono rumore.

2 Indicizzazione

L'indicizzazione si compone di 4 fasi principali, da eseguire sequenzialmente ma non tutte indispensabili ai fini di rappresentare il contenuto informativo di un documento. Scopo dell'indicizzazione è proprio quello di fornire una rappresentazione più compatta e facilmente utilizzabile di un documento, grazie all'estrazione dei suoi descrittori.

2.1 Analisi Lessicale

L'analisi lessicale ha come fine l'estrazione dei *token*, ovvero i potenziali descrittori di un documento. Questa fase è fortemente influenzata dalla lingua

nella quale i documenti sono scritti, poiché questa influisce nei caratteri di separazione, nella punteggiatura, nella codifica del documento, nella rappresentazione della date.

2.2 Rimozione Delle Stop Word

Le stop word sono parole con scarso contenuto informativo, solitamente a carattere funzionale (aricoli, pronomi, preposizioni, congiunzioni). In questa fase possono essere utilizzate delle liste precompilate (implementate solitamente per mezzo di tabelle hash) oppure si preparano dei modelli statistici.
La rimozione delle stop word influisce sostanzialmente nella dimensione dell'indice, infatti pur essendo solitamente meno di un centinaio di elementi, ad esse sono associate delle *posting list* molto voluminose.
Tuttavia è sempre bene chiedersi quali possano essere gli effetti di questa fase. Un esempio molto noto riguarda la celebre frace *To be or not to be*, che difficilmente potrebbe essere ricostruita dopo la rimozione delle stop word.

2.3 Stemming

Lo stemming (da *stem*, radice) è quella fase che permette di catturare le relazioni sussistenti tra le varie forme di un termine. In particolare, si occupare di ricondurre le parole alla loro radice semantica.
I due principali approcci con i quali si esegue questa fase sono:

- **algoritmico**, un programma decide se due parole sono semanticamente connesse;
- **basato su dizionario**, uno o più dizionari preparati in precedenza, mantegono le relazioni esistenti fra i vari termini.

Il più famoso stemmer algoritmico è il *Porter Stemmer*, creato per la lingua inglese.
Le principali problematiche relative a questa fase riguardano la creazione di uno stemmer per ciascuna lingua o, seguendo il secondo approccio, la creazione e l'aggiornamento dei dizionari.
Bisogna poi assicurarsi che lo stemmer non rimuova o non lasci troppa informazione in un termine: questi effetti prendono il nome di *over-stemming* e *under-stemming*.

2.4 Composizione Dei Termini

Un problema frequente che i sistemi di reperimento dell'informazione devono affrontare è la ricerca di query costituite da pochi termini (solitamente due o tre), per i quali l'utente si attende come risultato quei documenti che contengono proprio quelle frasi.
Ci si pone dunque il problema di cosa sia una frase e di quali frasi è bene mantenere traccia. Tuttavia questa fase è strettamente legata al modello di reperimento che viene utilizzato dall'IRS.

2.5 Struttura Dati

La struttura dati che viene generata al termine dell'indicizzazione prende il nome di **inverted index**, sebbene con questo nome si indica spesso una famiglia di strutture con caratteristiche simili.
L'indice serve alla memorizzazione dell'elenco dei termini, o *feature*, presenti nei documenti. Ogni termine possiede la sua **inverted list** (o **posting list**), che mantiene le informazioni relative a quel termine.
La posting list contine solitamente gli identificatori dei documenti che contengono quel termine, più ulteriori informazioni (spesso dipendenti dal modello) che permettono di calcolare lo **score** più efficientemente. Una tipica scelta per la posting list è quella di assciare all'identificatore del documento la frequenza relativa del termine.

3 Modello Booleano

Un modello di reperimento dell'informazione è un insieme di costrutti che sono stati ideati e poi formalizzati allo scopo di rendere possibile:

- la rappresentazione del contenuto dei documenti;
- la rappresentazione delle interrogazioni;
- la realizzazione degli algoritmi di reperimento dei documenti in risposta ad un'interrogazione.

Nel modello booleano le interrogazioni permettono all'utente di esprimere la propria esigenza informativa per mezzo di propozioni booleane:

- un termine x rappresenta una proposizione booleana atomica ed è resa vera da tutti i documenti che contengono quel termine;
- $x\ OR\ y$ rappresenta una proposizione composta resa vera da tutti i documenti che contengo il termine x oppure il termine y;
- $x\ AND\ y$ rappresenta una proposizione composta resa vera da tutti i documenti che contengo sia il termine x che il termine y;
- $NOT\ x$ rappresenta una proposizione composta resa vera da tutti i documenti che non contengono il termine x.

Nel modello booleano la funzione di reperimento associa all'interrogazione l'insieme dei documenti che soddisfano la proposizione indicata. Tuttavia, questa funzione non consente di ordinare i documenti secondo la loro rilevanza.

3.1 Livello Di Coordinamento

4 Modello Probabilistico

5 Modello Vettoriale

6 Valutazione Dei Sistemi Di Reperimento

Per poter valutare un IRS è necessario disporre dei **giudizi di rilevanza**, ovvero dei valori associati a ciascuna coppia documento-topic. Esistono più metodi per la creazione di questi giudizi:

- **giudizi completi**, per ogni documenti si giudica la sua rilevanza relativamente ad ogni topic (lavoro troppo oneroso);
- **campionamento casuale**, i giudizi vengono assegnati soltanto ad un campione casuale di documenti, per ogni topic;
- **campionamento basato sugli esperimenti dei partecipanti**, è il metodo utilizzato da TREC ed ormai divenuto stardard per tutte le campagne di valutazione, noto anche come **pooling**.

Per poter dare una definizione formale del pooling è necessario introdurre alcuni concetti. Si definiscono:

$$D = \{d_1, ..., d_n\} \text{ un insieme di documenti}$$
$$T = \{t_1, ..., t_m\} \text{ un insieme di topic}$$

Dato un numero naturale $N \in \mathbb{N}^+$ detto *lunghezza della run*, una **run** è definita come

$$R : T \to D^N$$
$$t \mapsto \mathbf{r_t} = (d_1, ..., d_N)$$

tale che $\forall t \in T, \forall j, k \in [1, N] : j \neq k \implies \mathbf{r_t}[j] \neq \mathbf{r_t}[\mathbf{k}]$ dove $\mathbf{r_t}[j]$ indica il $j - esimo$ elemento del vettore $\mathbf{r_t}$.
Fornito un valore k detto **profondità del pool**, si selezionano i **top-k** elementi di ciascuna run e si considera quindi l'insieme unione dei documeti ottenuti (il valore solitamente utilizzato è $k = 100$). Infine si creano i giudizi di rilevanza soltanto per i documenti dell'insieme così definito.
I documenti non giudicati vengono di solito giudicati non rilevanti (non sempre è così però).
A questo punto è lecito porsi due questioni:

- la valutazione dei sistemi usando un pool (e non un insieme completo) di giudizi di rilevanza è stabile?

- il pooling penalizza i sistemi le cui run non sono state incluse nel pool e/o i sistemi che utilizzano metodologie molto differenti da quelli utilizzati per la creazione del pool?

La prima questione l'accento sulla **completezza** del pool. Tuttavia le ricerche condotte hanno mostrato che pur aggiungedo al pool documenti oltre la posizione $k - esima$, si possono trovare alcuni documenti rilevanti dopo questa posizione, ma non abbastanza da influenzare significativamente la valutazione. Inoltre, i topic con molti documenti rilevanti tendono ad avere molti documenti rilevanti anche dopo la posizione $k - esima$.
La seconda questione considera invece il problema della **robustezza** del pool. Per testare la robustezza del pooling è stato messo in pratica un esperimento su di un gruppo di run (utilizzate per i giudizi di rilevanza di TREC). Ogni volta che una run doveva essere valutata si escludevano (indicandoli come *non rilevanti*) i giudizi di rilevanza ottenuti da tale run.
Con questo test è stato possibile mostrare che:

- la qualità del pool non influenza in modo significativo la qualità della collezione di test;
- le collezioni delle campagne di valutazione sono *unbaiased* verso i sistemi che non vengono utilizzati per la creazione dei giudizi di rilevanza;
- le prestazioni delle run valutate con il pool oiginale erano mediamente più alte delle prestazioni con il pool ridotto.

Una **collezione di test** è definita come una tripla:

$$\mathcal{C} = \{D, T, GT\}$$

dove D è l'insieme dei documenti, T è l'insieme dei topic e GT è il **ground truth** della collezione.
Sia REL un insieme finito di **gradi di rilevanza** e sia $\preceq$ una relazione d'ordine totale su REL tale che $(REL, \preceq)$ sia un insieme totalmente ordinato.
Dato un insieme T di topic e un insieme D di documenti, il ground truth è una funzione:

$$GT : T \times D \to REL$$
$$(t, d) \mapsto rel$$

6.1 Misure Con Rilevanza Binaria

Data una run $R(t) = \mathbf{r}_t$, il **relevance score** della run è una funzione

$$\hat{R} : T \times D^N \to REL^N$$
$$(t, \mathbf{r}_t) \mapsto \hat{\mathbf{r}}_t = (rel_1, rel_2, ..., rel_N)$$

dove

$$\hat{\mathbf{r}}_t[j] = GT(t, \mathbf{r}_t[j])$$

Sia $W \subset \mathcal{Z}$, REL