Da mettere alla fine degli esperimenti

Alessandro Stefani[1], Caterina Buranelli[2], and Cristi Gutu[3]

[1] Corso di laurea in Statistica per le tecnologie e le scienze, matricola 1148387
alessandro.stefani.6@studenti.unipd.it
[2] Corso di laurea in Statistica per le tecnologie e le scienze, matricola 1234567
caterina.buranelli@studenti.unipd.it
[3] Corso di laurea in Statistica per le tecnologie e le scienze, matricola 1147351
gheorghecristi.gutu@studenti.unipd.it

Sommario DA FINIRE QUANDO ABBIAMO MESSO A POSTO LA SEZIONE ESPERIMENTI. Questo progetto tratta la realizzazione attraverso il pacchetto *Whoosh*, di un motore di ricerca volto al reperimento di documenti della collezione sperimentale *OHSUMED* indicizzata opportunamente. Il progetto anche corredato di un webserver che permette all'utente di interrogare il motore di ricerca in forma interattiva attraverso un browser a scelta.

Keywords: *Information Retrieval · IR · statistica · reperimento · indicizzazione · Whoosh · Python · webserver · web · interrogazione web*

1 Introduzione

Lo scopo finale di un Sistema di Information Retrieval (IRS) e' quello di reperire documenti rilevanti relativi a una certa esigenza informativa[4]; dunque i documenti sono il primo ingresso del sistema, mentre il secondo e' costitutito dalle interrogazioni; i documenti devono essere indicizzati e nell'indice creato, si andra' a effettuare la ricerca per reperire documenti rilevanti[5]. Questa seconda parte e' detta reperimento e non si occupa solo di ricercare tra i documenti, ma anche di riordinare secondo un certo ordine di rilevanza. Cio' che descrive i documenti e cio' che descrive le interrogazioni deve essere confrontabile, infatti nei programmi di indicizzaizone e di reperimento si usa uno schema, che deve essere uguale in entrtambi i casi. L'indicizzazione e' un trade-off tra il miglioramento della rapprerentazione del contenuto informativo dei documenti (efficacia) e la gestione degli indici (efficienza). Benche' ci sia sempre tensione tra queste due caratteristiche, ad oggi la questione piu' studiata e' quella dell'efficienza.

[4] insieme delle circostanze in cui unapersona ha un problema da risolvere o un compito da svolgeree richiede informazioni importanti, utili o necessarie per larisoluzione del problema o lo svolgimento del compito

[5] la rilevanza e' la proprieta' che rende l'informazione importante, utile o necessaria a soddisfare l'esigenza informativa dell'utente

Esistono diversi modi per risolvere questo problema, nel nostro caso abbiamo ricercato la configurazione migliore tra un sistema di reperimento con o senza uso di stopword e il tuning dei parametri dello schema di pesatura[6] BM25F .

L'obiettivo principale della relazione è da una parte la documentazione del progetto di un servizio di IR e dall'altra una misura del grado in cui si sia riusciti a mettere in pratica i contenuti della disciplina illustrati durante le lezioni.

A tal scopo la relazione dovrà illustrare nelle sezioni successive:

- i metodi di indicizzazione,
- i modelli di reperimento,
- l'interfaccia basata su un *browser* per il WWW
- i risultati della *valutazione* condotta con la collezione sperimentale OHSUMED.

Il lettore della relazione è lo studente medio di un corso di laurea in statistica al quale la relazione deve dare tutti gli strumenti per comprendere il contenuto. Ci si metta nei suoi panni e si scriva tutto ciò e solo ciè che serve. Chiedersi qual è il messaggio che lo studente deve "portarsi a casa", esplicitarlo in questo paragrafo e concentrarsi su quello nel resto della relazione.

L'introduzione della relazione deve servire al lettore a capire se vale la pena continuare a leggere il resto. Si possono riassumere i contenuti delle sezioni successive e metterne in evidenza i punti principali. La relazione consiste di tre paragrafi principali dopo questa introduzione e prima della bibliografia, per la quale si suggerisce BibTEX se si scrive con LATEX.

2 Base di partenza

La base di partenza è formata dai metodi documentati nei libri di testo. Si eviti di trascrivere pari pari, si cerchi piuttosto di rielaborare i contenuti in modo da renderli *coerenti* col resto della relazione; in particolare, si descrivano tutti e solo i metodi usati negli esperimenti e si eviti di parlare di quei metodi che poi non sono stati usati; ad esempio, se si conducono degli esperimenti con BM25F, si deve descrivere questo schema di pesatura in questa sezione.

"Best Match 25 Model with Extension to Multiple Weighted Fields"

Le stopword[2] sono parole che non portano informazione significativa al contenuto informativo come congiunzioni, articoli, avverbi..

3 Esperimenti e Risultati

Per gli esperimenti si è utilizzata parte della già citata collezione sperimentale chiamata OHSUMED[7].

[6] funzione che assegna per ogni documento diversi livelli dimportanza dei termini mediante dei pesi, che possono variare con linterrogazione

[7] https://bit.ly/2wpOynZ

Si tratta di una collezione di oltre 300'000 articoli provenienti dal database bibliografico MEDLINE pubblicati tra il 1987 ed il 1991, di questi documenti ne sono stati utilizzati 54'711.

I programmi utilizzati per effettuare indicizzazione e reperimento sono stati scritti in python(versione 2.7), sfruttando oggetti e funzioni del modulo whoosh[8].

Per la valutazione dei risultati ottenuti nei vari esperimenti si è invece utilizzato lo strumento standard trec_eval[9]. Questo programma permette di ottenere alcune misure della qualità di un sistema di information retrieval se si conoscono i documenti rilevanti per le query sperimentali utilizzate.

Come misura da usare nel confronto tra risultati di diverse run si è scelta la Mean Average Precision(MAP)[3] , quantità calcolabile sia a livello di singola query che a livello complessivo su un insieme di query. Inoltre, per verificare se i valori di MAP ottenuti in run diverse si discostano in modo statisticamente significativo gli uni dagli altri si effettuano test di Wilcoxon[] su coppie di risultati.

3.1 Scelte iniziali

Prima di iniziare gli esperimenti sono state fatte delle scelte riguardo certi aspetti dei metodi di reperimento che rimanessero costanti durante tutti gli esperimenti. Queste scelte sono state fatte in seguito ad alcune run preliminari.

Per la funzione di reperimento e ranking come schema di pesatura si è deciso di utilizzare il BM25F, descritto nella sezione precedente, inquanto "state-of-the-art" dal punto di vista dei modelli per information retrieval su documenti strutturati, e come operatore logico per raggruppare le parole delle query si è deciso di utilizzare l'operatore OR, ovvero si cercano documenti in cui è presente almeno una parola della query.

Sempre a seguito delle run preliminari si è deciso di utilizzare il campo "desc" delle query anzichè il campo "title" inquanto dà risultati migliori, inoltre, si è notato che in alcune query sperimentali sono presenti parole scritte in modo errato e che quindi non risultano presenti nell'indice. Per questo dopo aver verificato, leggendo i documenti rilevanti per quelle query, quali fossero gli errori si è ritenuto opportuno effettuare una correzione al momento del reperimento. A grandi linee, la correzione viene fatta su parole che non risultano presenti nell'indice cercando in quest'ultimo delle parole alternative che si discostano di al più una lettera dalla parola originale; queste parole vengono quindi aggiunte alla query di partenza e si procede con la ricerca.

Un'ultima scelta è stata fatta riguardo il numero massimo di documenti reperiti. Si è optato per 100 documenti poiche un numero maggiore non porta grandi miglioramenti dal punto di vista del MAP.

[8] https://whoosh.readthedocs.io/en/latest/index.html
[9] https://trec.nist.gov/trec_eval/

Affinchè Whoosh possa indicizzare una collezione di documenti, necessita la specificazione di uno schema che include, per ogni possibile campo dei documenti della collezine, il nome del campo ed il tipo del campo.

Lo schema di base per la collezione qui utilizzata è il seguente:

```
schema = Schema(docid          = ID(stored=True),
                title          = TEXT(stored=True),
                identifier     = ID(stored=True),
                terms          = NGRAM(stored=True),
                authors        = NGRAM(stored=True),
                abstract       = TEXT(stored=True),
                publication    = TEXT(stored=True),
                source         = TEXT(stored=True))
```

Figura 1. Schema necessario all'indicizzazione dei documenti. "stored=True" indica che il campo viene salvato ed è quindi successivamente accessibile.

3.2 Baseline

I valori di MAP scelti come baseline sono quelli ottenuti effettuando ricerche su un indice con lo schema di base, riportato sopra, senza l'utilizzo di una lista di stopword e cercando in due campi dei documenti: "title" e "abstract". Cercando su un campo ("title") o tre campi ("title", "abstract" e "terms") risulta solo in un peggioramento dei risultati.

Si possono vedere i risultati ottenuti usando la configurazione "baseline" riassunti in Tabella 1.

un campo	**due campi**	**tre campi**
num*q all 63*	num*q all 63*	num*q all 63*
num*ret all 6228*	num*ret all 6244*	num*ret all 6271*
num*rel all 670*	num*rel all 670*	num*rel all 670*
num*rel*ret all 349	num*rel*ret all 419	num*rel*ret all 367
map all 0.2130	map all **0.2744**	map all 0.2155

Tabella 1. Risultati treceval complessivi per tutte le query, nessuna manipolazione del testo, numero risultati restituiti per ogni query = 100, pesatura BM25F.

Come si nota dalla tabella il valore baseline di $M.A.P$ è 0.2744.

Il processo ed il codice per creare l'indice sono facilmente comprensibili visionando il file *indicizzatore_batch_baseline.py*.

Per eseguire l'indicizzazione baseline è sufficiente lanciare lo script python con il seguente comando:

```
indicizzatore_batch_baseline.py \
cartella_indice file_documenti.xml
```

Per eseguire il reperimento, che poi produce il file con i risultati in formato compatibile con trec_eval, è sufficiente lanciare lo script python con il comando:

```
python search_tk.py cartella_indice \
file_query.xml cartella_risultati nome_file_risultati
```

3.3 Esperimenti

Gli esperimenti sono consistiti in tre prove in cui si cambia l'insieme di stopword usato nell'indicizzazione. In particolare ad ogni prova si utilizza un insieme di stopword con parole in più rispetto a quello delle prove precedenti.

Come con la baseline, per ciascuna prova si effettua la ricerca per uno, due e tre campi.

Prima prova: Per la prima prova sono state utilizzate le stopword generali della lingua inglese. Queste comprendono termini come "the", "that", "is"; sono poi state aggiunte anche le lettere dell'alfabeto ed i numeri.

Come si vede in Tabella 2, non sembra si sia ottenuto un miglioramento significativo rispetto alla baseline. Il MAP ottenuto utilizzando due campi "migliora" solo dello 0.0001.

un campo	**due campi**	**tre campi**
num*q all 63*	num*q all 63*	num*q all 63*
num*ret all 6228*	num*ret all 6244*	num*ret all 6271*
num*rel all 670*	num*rel all 670*	num*rel all 670*
num*rel*ret all 350	num*rel*ret all 418	num*rel*ret all 367
map all 0.2137	map all **0.2745**	map all 0.2152

Tabella 2. Risultati treceval, rimozione delle stopword generali, numero risultati restituiti per ogni query = 100, pesatura BM25F.

Seconda prova: Nella seconda prova si è allora cercato di migliorare i risultati aggiungendo alle stopword generali, alcune parole che non sono normalmente considerate stopword ma che nel contesto medico, come nel caso della collezione OHSUMED, diventano tali.

Per questo abbiamo utilizzato un insieme di stopword denominate stopword_cliniche che oltre ad avere le stopword generali ha anche parole come "medical", "condition" o "family"[10]

I risultati sono stati accettabili, ma siamo riusciti a migliorarli togliendo anche stopword cliniche, cioè strettamente inerenti al contesto medico e abbiamo ottenuto i seguenti risultati:

[10] https://github.com/kavgan/clinical-concepts/blob/master/clinical-stopwords.txt

un campo	due campi	tre campi
num*q all 63*	num*q all 63*	num*q all 63*
num*ret all 6228*	num*ret all 6244*	num*ret all 6271*
num*rel all 670*	num*rel all 670*	num*rel all 670*
num*rel*ret all 350	num*rel*ret all 419	num*rel*ret all 376
map all 0.2156	map all **0.2837**	map all 0.2166

Tabella 3. Risultati treceval, rimozione delle stopword cliniche, numero risultati restituiti per ogni query = 100, pesatura BM25F.

Terza prova: Nell'ultima prova sono state aggiunte altre parole alla lista delle stopword, la scelta delle parole da aggiungere è stata fatta, prima prendendo in considerazione le parole delle query e la frequenza con cui queste si trovano nei documenti rilevanti e poi cercando la presenza di possibili stopword nelle query che davano risultati peggiori.

I risultati sono stati accettabili, ma siamo riusciti a migliorarli togliendo anche stopword cliniche, cioè strettamente inerenti al contesto medico e abbiamo ottenuto i seguenti risultati:

un campo	due campi	tre campi
num*q all 63*	num*q all 63*	num*q all 63*
num*ret all 5282*	num*ret all 5690*	num*ret all 5788*
num*rel all 670*	num*rel all 670*	num*rel all 670*
num*rel*ret all 386	num*rel*ret all 463	num*rel*ret all 435
map all 0.2720	map all **0.3508**	map all 0.2813

Tabella 4. Risultati treceval, rimozione delle stopword cliniche migliorate, numero risultati restituiti per ogni query = 100, pesatura BM25F.

Riferimenti bibliografici

1. W. Bruce Croft and Donald Metzler and Trevor Strohman. Search Engines: Information Retrieval in Practice. Addison Wesley, (2009), pp. 250-252
2. W. Bruce Croft and Donald Metzler and Trevor Strohman. Search Engines: Information Retrieval in Practice. Addison Wesley, (2009), pp. 90
3. W. Bruce Croft and Donald Metzler and Trevor Strohman. Search Engines: Information Retrieval in Practice. Addison Wesley, (2009), pp. 313
4. Discovering Related Clinical Concepts Using Large Amounts of Clinical Notes. Ganesan, Kavita and Lloyd, Shane and Sarkar, Vikren, (2016), pp. 27-33

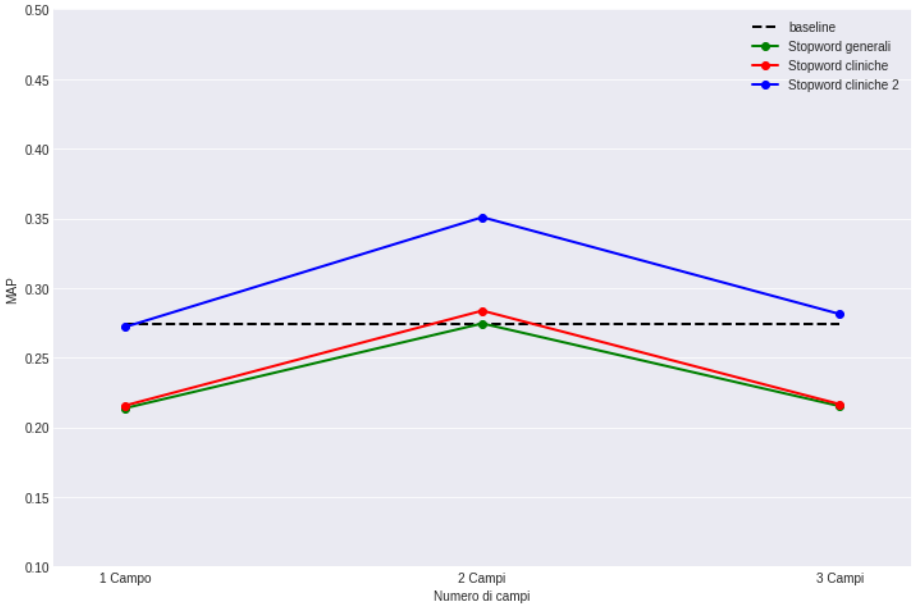

Figura 2. Performace modelli di classificazione sul test set in funzione della dimensione del training set.

(a) Alessandro Stefani

(b) Caterina Buranelli

(c) Cristi Gutu