Leonardo Lerose
1197669

Homework 1 - Relazione

Specifiche del sistema di IR

Il sistema di reperimento dell'informazione considerato è Terrier, versione 4.4. Terrier permette di effettuare analisi lessicale, indicizzazione ma anche reperimento e valutazione, incapsulando tutte le operazioni tipiche del reperimento dell'informazione all'interno di un unico sistema.
Seguendo la documentazione di Terrier si possono eseguire tutte le operazioni citate sopra manipolando il file di configurazione `terrier.properties` e lanciando dei comandi da shell.
La collezione di documenti presa in considerazione è TIPSTER: Trec 4 e 5.
La valutazione nello specifico è stata svolta usando gli strumenti proposti da Trec eval, comunque integrati all'interno di Terrier.

Indicizzazione

Per eseguire l'indicizzazione con Terrier, è stato lanciato il comando `./trec_terrier.sh -i`.
Per soddisfare i requisiti richiesti dalle varie run (eseguire del pre-processing con combinazione di stopword removal e stemming prima della creazione dell'indice) è necessario modificare l'impostazione `termpipelines` all'interno del file `terrier.properties` nel seguente modo:

- `termpipelines=Stopwords,PorterStemmer` nella Run #1 e #2
- `termpipelines=PorterStemmer` nella Run #3
- `termpipelines=#` nella Run #4

Reperimento

Per quanto riguarda il reperimento con Terrier, inizialmente è stato necessario settare l'attributo `trec.topics` al path del file contenente i topic da utilizzare per il reperimento (`topics.351-400_trec7.txt`) nel file `terrier.properties`. Inoltre, sempre in nelle properties, sono stati settati la proprietà `ignore.low.idf.terms=true` per ignorare i term con idf bassa e
il modo in cui vengono interpretate ed elaborate le query, impostando gli attributi di `TrecQueryTags, doctag=TOP, idtag=NUM, process=TITLE,DESC,skip=NARR`.
Una volta eseguite le precedenti operazioni, si può procedere al reperimento effettivo lanciando il comando `./trec_terrier.sh -r -Dtrec.model=BM25` con il modello probabilistico BM25 richiesto nelle Run #1 e #3, oppure `./trec_terrier.sh -r -Dtrec.model=TF_IDF` per il reperimento con il modello TF*IDF richiesto nelle Run #2 e #4.
I risultati del reperimento vengono salvati in file `.res` visualizzabili all'interno del repository.

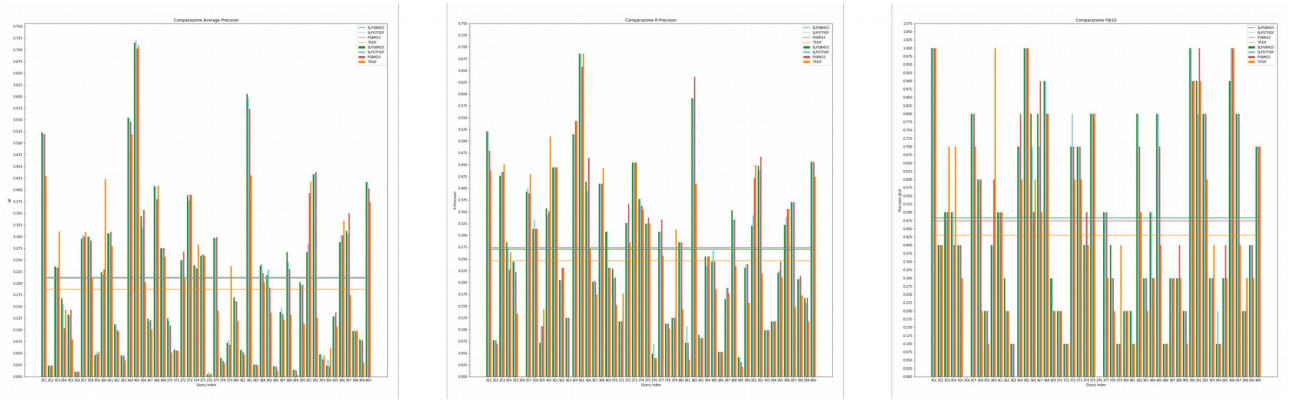
Valutazione

Ottenuti i risultati del reperimento, per l'operazione di valutazione è stato usato trec_eval. Il comando in questione calcola in maniera automatica varie measure utili ai fini della valutazione. Per funzionare necessita ovviamente di un file di qrels ovvero un file che contiene una lista di documenti considerati rilevanti per ogni query considerata nel reperimento. Per valutare le differenze tra i vari sistemi sono stati svolti su alcune measure il test Anova one-way e Tukey-HSD. Le varie run sono state valutate in funzione di AP, R-Precision e Precision at 10. Per queste measure sono stati svolti i test Anova one-way e Tukey-HSD.
Le conclusioni che si possono trarre analizzando i risultati sono le seguenti:

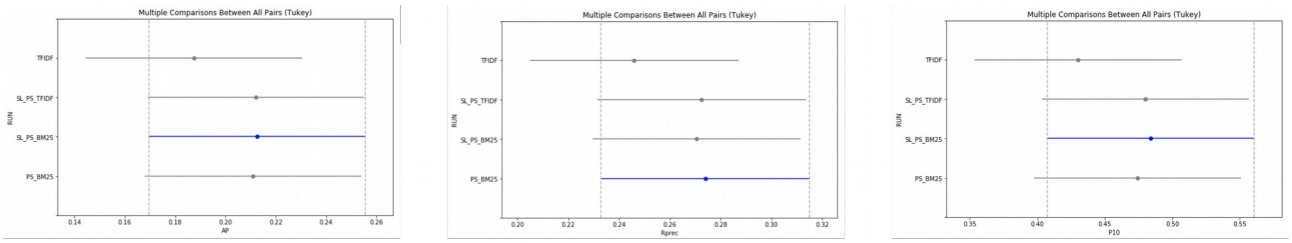
Per quanto riguarda l'hypothesis testing, osservando i risultati del test Anova sulle varie measure, si ottengono dei valori del p-value alti (tra ~0.78 e ~0.85 a seconda della measure). Questa informazione ci permette di affermare che "falliamo nel rigettare" la null hypothesis e quindi le varie run sono simili tra loro. Il test Tukey-HSD ci permette di affermare che tutte le run appartengono a quello che è definito "top group", con la run con porter stemmer, stop word e

modello BM25 che in due casi su tre (analizzando le measure) definisce quali sono i bound per l'appartenenza o meno al top group.
Ad ogni modo per quanto riguarda le performance delle varie run, possiamo osservare come le run che usano porter stemmer e/o rimozione delle stopword (run #1, run #2, run #3) abbiano delle prestazioni migliori rispetto alla run che non usa ne porter stemmer ne rimozione delle stopword.
Le prime tre run, ottegono valori di AP attorno al 21%, R-precision attorno al 27%, Precision at 10 attorno al 48%. Mentre la quarta AP attorno al 18%, R-precision attorno al 24%, Precision at 10 attorno al 43%. Si rimanda al notebook python per i risultati numerici in dettaglio.

Si riportano alcuni risultati grafici(osservabili dettagliatamente nel notebook e nel repository):



In figura: plot comparativi delle varie run sulle measure AP, R-precision, Precision at 10. Le linee orizzontali indicano il valori medi su tutti i topic (dettagli in repository)



Multiple Comparison of Means - Tukey HSD,FWER=0.05

group1	group2	meandiff	lower	upper	reject
PS_BM25	SL_PS_BM25	0.0018	-0.0042	0.0077	False
PS_BM25	SL_PS_TFIDF	0.0012	-0.0048	0.0072	False
PS_BM25	TFIDF	-0.0233	-0.1093	0.0626	False
SL_PS_BM25	SL_PS_TFIDF	-0.0005	-0.0865	0.0855	False
SL_PS_BM25	TFIDF	-0.0251	-0.1111	0.0609	False
SL_PS_TFIDF	TFIDF	-0.0246	-0.1106	0.0614	False

Multiple Comparison of Means - Tukey HSD,FWER=0.05

group1	group2	meandiff	lower	upper	reject
PS_BM25	SL_PS_BM25	-0.0034	-0.0854	0.0786	False
PS_BM25	SL_PS_TFIDF	-0.0015	-0.0835	0.0805	False
PS_BM25	TFIDF	-0.028	-0.11	0.054	False
SL_PS_BM25	SL_PS_TFIDF	0.0019	-0.0801	0.0839	False
SL_PS_BM25	TFIDF	-0.0246	-0.1066	0.0574	False
SL_PS_TFIDF	TFIDF	-0.0265	-0.1085	0.0555	False

Multiple Comparison of Means - Tukey HSD,FWER=0.05

group1	group2	meandiff	lower	upper	reject
PS_BM25	SL_PS_BM25	0.01	-0.1432	0.1632	False
PS_BM25	SL_PS_TFIDF	0.006	-0.1472	0.1592	False
PS_BM25	TFIDF	-0.044	-0.1972	0.1092	False
SL_PS_BM25	SL_PS_TFIDF	-0.004	-0.1572	0.1492	False
SL_PS_BM25	TFIDF	-0.054	-0.2072	0.0992	False
SL_PS_TFIDF	TFIDF	-0.05	-0.2032	0.1032	False

In figura: tabelle e plot dei confidence interval ottenuti dal test di Tukey sulle misure AP, R-precision, Precision at 10 (dettagli in repository)

Notebook Python e link al repository

Per analizzare i risultati ottenuti e riuscire a visualizzarli graficamente è stato usato un notebook Python, accessibile dal repository del quale si lascia il link successivamente.
I moduli Python importati ed utilizzati sono:

- `csv` e `re`, con metodi utilizzati per eseguire il parsing del testo di input
- `numpy`, per l'handling facilitato di vettori numerici contenenti valori di measure
- `matplotlib`, per il plot di tutti i risultati
- da `statsmodels.stats.multicomp` i moduli `pairwise_tukeyhsd` e `MultiComparison` per dei metodi usati nel test Tukey-HSD

Il lavoro svolto è visualizzabile al seguente link github:

https://github.com/LeonardoLerose/homework1-ir