Progettini per il corso IR 2012.

Per domande inviare email sia al sottoscritto sia a vitaled@gmail.com

Progetto 1.

L'obiettivo del progetto è quello di sperimentare una serie di algoritmi di clustering su un dataset di news in italiano. Ogni gruppo di lavoro (formato da 2 o 3 persone) sperimenterà un solo algoritmo, concordato con il docente.

Dataset: su dropbox () troverete un file .tgz che contiene due set di files. Un insieme di news (news-....) ciascuna di un tipo diverso di categoria (top, politica, italia, sport, intrattenimento, technology) e un insieme di cluster già calcolati su questi file di news.

Ogni file contiene ++++ news, ed è strutturato per righe, ciascuna avente la forma:

[epoch][\t][testata giornalistica][\t][titolo news][\t][corpo news][\t]

Ogni file di clustering è strutturato per righe, ciascuna avente la forma:

[intero non significativo][\t][sequenza termini importanti del cluster, divisi da spazi][\t][sequenza ID news ordinate per importanza nel cluster][\t][flag binario: 1 ok, 0 no]

Il flag binario indica se il cluster è interessante o no. E' auspicabile che i cluster siano ordinati in ordine decrescente di importanza. Se ciò non è possibile ordinarli in modo decrescente di size.

Clustering: Si vuole progettare un algoritmo di clustering soltanto usare codici C o C++ e sfruttare librerie open-source, p.e. GPL, tra quelle indicate qui di seguito. Altre proposte devono essere concordate con il docente.

1) Libreria Gmeans. http://www.dataminingresearch.com/index.php/2010/06/gmeans-clustering-software-compatible-with-gcc-4/
2) Co-clustering. http://www.cs.utexas.edu/users/dml/Software/cocluster.html
3) Riscrivere in C++ il codice dell'optimal bisect http://code.google.com/p/airhead-research/source/browse/branches/cluster/src/edu/ucla/sspace/clustering/BisectingKMeans.java?r=1046.
4) Costruire grafo di news, con arco che indica la misura di jaccard su shingles di lunghezza 3 per i termini stemmati. Le shingles potrebbero forse essere pesate con una sorta di TF-IDF (da valutare). Poi si fa clustering, ordinando gli archi per peso crescente, ed eliminando questi nell'ordine a partire da quelli di peso minore, e via via a crescere fintanto che si ottengono k clusters o l'arco da eliminare è maggiore di una certa soglia.
5) Latent Semantic Indexing: http://gibbslda.sourceforge.net/

Varie osservazioni:

- Si potrebbe pensare di usare come features non solo i singoli termini ma shingles di 3 termini ciascuno, eventualmente stemmati. O solo le shingles.
- Potreste studiare se il metodo adottato offre un modo semplice ma efficace per determinare i termini significativi di ogni cluster.

Visualizzazione: Fare una valutazione dei propri risultati con quelli forniti a esempio per la bontà della soluzione proposta. Il confronto è "a occhio" e deve essere fatto in modo oggettivo; l'accettazione del progetto non dipende dall'aver trovato un risultato migliore, ma di aver condotto un'analisi completa e seria.

Per la valutazione è stata messa a punto un'interfaccia grafica, mediante browser, disponibile su:

Progetto 2.

I dataset sono i clusters forniti nel punto precedente.
L'obiettivo del progetto è quello di calcolare *clique di termini* di size 5 di ogni cluster di news, con una valutazione della bontà di ogni clique trovata (possono essere anche più di una per cluster). Una clique è definita come: "Represents the most descriptive features via a graph in which two features are connected via an edge if and only if their co-occurrence frequency within the cluster is greater than their expected co-occurrence." Alcune librerie per determinare clique su grafi si trovano su: http://en.wikipedia.org/wiki/Clique_problem, dove è anche indicato un semplice algoritmo per calcolare la clique di dimensione 3 (triangoli: Cliques of fixed size).
I triangoli poi possono essere estesi a clique di dimensione 4 e poi 5. Interessante è anche la libreria http://igraph.sourceforge.net/ per il calcolo delle clique oppure per altre strutture che esprimono coesione tra termini (community?).

Progetto 3.

Progettare un crawler (in perl?) che scarica a intervalli di tempo predefiniti le rassegne stampa di quotidiani nazionali. Ad esempio:

- http://rstampa.pubblica.istruzione.it/rassegna/rassegna.asp
- http://rassegnastampa.unipi.it/sup/
- http://www.interno.gov.it/mininterno/site/it/sezioni/sala_stampa/rassegna_stampa/
- ...

L'idea è quella di costruire un db con un record per rassegna stampa, quindi contenente la data, la testata giornalistica, il titolo, e il link al pdf (o anche il pdf stesso, scaricato in locale), eventualmente la categoria della news (top, politica, sport, spettacolo,...) se individuata nella rassegna medesima.
Un compito è quello di trovare più siti di rassegne e quindi scaricare più cose, eventualmente eliminando i duplicati.

Si potrebbe poi immaginare di sviluppare un semplicissimo algoritmo che cerca il match tra una rassegna e un insieme di news (presenti nei vari file disponibili nei dataset) per trovare la news giusta da associare, p.e. basandosi sull'uguaglianza della sorgente, sulla quasi-uguaglianza del titolo [può accadere che una news dal feed rss al giornale stampato cambi il titolo, ciò però deve essere un piccolissimo cambio per essere accettabile].
IL test potrebbe consistere di ricevere in input un file di news, e per ciascuna cercare nel DB-rassegna quella che corrisponde alla news, se c'é, altrimenti non restituire nulla. Nel caso di match si stampa a video la news e il link al pdf, tanto per poter fare verifiche a occhio.

Progetto 4.

Progettare un sistema che, data la URL di una news da un quotidiano che ha **anche** un canale su twitter (potete considerare i più importanti e crearvi una tabellina), accede al canale di twitter di quel quotidiano e si scarica gli ultimi tweet. Poi verifica se ce ne è uno con un link (da risolvere, visto che sono tutti short) a quella URL (tenete conto che si tratta di news e tweet dello stesso quotidiano quindi ci dovrebbe essere una certa sincronizzazione).
Trovato il tweet, si scarica tutti i suoi **retweet**, e stampa a video un po' di statistiche che riassumo in un qualche modo la visione di Twitter su quella news e la sua "importanza":

- Numero di retweet ottenuti
- Hashtag più usati e loro frequenza
- i top-10 tweet che sono stati retweettati dalla comunità

- ...???...