UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Gene-Disease association analyzing the scientific literature

Salvatore Calderaro

Indice

1 Introduzione

In questo progetto è stato implementato un sistema che dato in input un gene (il suo ID), controlla se l'ID inserito appartiene a un gene realmente esistente, se il controllo va a buon fine vengono memorizzate all'interno di un dataframe una serie di informazioni inerenti il gene, in particolare: la tassonomia, l'ID, il simbolo e il nome ufficiale completo. Fatto ciò si procede con l'estrazione dalla piattaforma *PubMed* - mediante *web scraping* - dei duecento articoli più rilevanti (se disponibili) in cui il gene è stato studiato. Di quest'ultimi vengono estratti e caricati all'interno di un dataframe titolo ed abstract. Per ogni articolo presente nel dataframe viene effettuato un pre-processing: eliminazione delle stop words, della punteggiatura e altre tecniche di natural language processing che verranno descritte più approfonditamente nella prossima sezione. Fatto ciò si procede con la *Named Entity Recognition* (NER), per stabilire all'interno di un testo quali parole o insiemi parole possono essere etichettate come malattie. La lista di malattie cosi ottenuta viene filtrata in modo tale da eliminare eventuali duplicati o parole che si hanno a che fare con l'ambito biomedico, ma che in realtà non sono nomi riconducibili a malattie. Infine per valutare la bontà dei risultati ottenuti, la lista di malattie viene confrontata, mediante *fuzzy string matching* con la lista di malattie che sono associate al gene che si sta studiando estratta dal database *DisGenNet*. In output verrà restituita: la percentuale di malattie esatte trovate, la lista della malattie e una wordcloud per rappresentare graficamente i risultati. Il codice sorgente del software è reperibile alla seguente repository GitHub: https://github.com/Calder10/Gene-Desease-Association.

2 Descrizione del software

Il linguaggio di programmazione utilizzato per l'implementazione del software è *Python*. Come IDE per effettuare lo sviluppo è stato scelto *Atom*. Le librerie utilizzate per l'implementazione del software sono le seguenti:

- *pyspark*:
- *nltk*;
- *biopython*;
- *scispacy*;
- *spacy*;
- *textblob*;
- *fuzzywuzzy*;
- *wordcloud*;
- *matplotlib*.

I dati sono stati reperiti mediante *Entrez*, un sistema di ricerca integrato tra banche dati biomediche contenenti informazioni di tipo differente coordinato dal *NCBI*. Di seguito verranno descritte in dettaglio le varie fasi della progettazione del software.

2.1 Estrazione delle informazioni inerenti il gene

Il sistema, una volta che l'utente ha inserito in input l'ID del gene di cui vuole ricavare le malattie associate, verifica se a quell'ID è associato effettivamente un gene mediante la funzione *check_gene(gene_id)*. Tale funzione esegue una query tramite *Entrez* sul database associato *Gene*, che raccoglie informazioni di sequenza centrate sui singoli geni. Qualora la query avesse esito positivo viene restituito un file XML, dalla quale vengono estratte ed inserite all'interno di un dataframe le seguenti informazioni:

- *TaxonomyName*;
- *ID*;
- *OfficialSymbol*;
- *OfficiallFullName*.

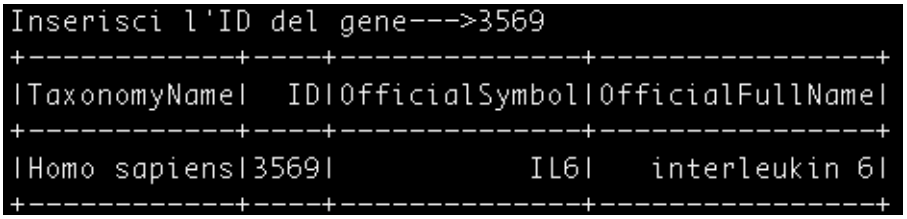

Figura 1: Dataframe contenente le informazioni sul gene

Qualora la query dovesse avere esito negativo viene visualizzato un messaggio di errore e viene richiesto l'inserimento di nuovo ID.

2.2 Estrazione degli articoli scientifici

Una volta identificato il gene di cui si vogliono ricavare le malattie associate, si procede - mediante *web scraping* - all'estrazione della letteratura scientifica inerente il gene in questione tramite la funzione *find_papers(gene_id)*. Quest'ultima prende in input l'ID del gene e sempre mediante l'utilizzo di *Entrez* prima effettua una query per verificare l'esistenza di riferimenti (link) agli articoli più rilevanti correlati all'ID cercato (l'ID del gene). Se non si dovessero trovare riferimenti viene mostrato all'utente un messaggio di errore. Successivamente per ogni link trovato si effettua una query sul database *PubMed* il quale raccoglie i riferimenti agli articoli scientifici apparsi su un numero elevato di riviste scientifiche, principalmente di tipo biomedico. Il risultato di tale query è sempre organizzato in formato XML, e contiene tutta una serie di informazioni inerenti l'articolo: l'anno di pubblicazione, la rivista, il titolo, l'abstract etc.

Degli articoli trovati ne vengono considerati solamente duecento - se disponibili. Di quest'ultimi le informazioni che vengono estratte sono il titolo e l'abstract se è disponibile. Infine questi duecento articoli con relativi titoli ed abstract vengono memorizzati all'interno di un dataframe.

Title	Abstract
Interleukin-6 in ...	The role of inter...
Role of Interleuk...	COVID-19 is viral...
EBV Rta-induced I...	Rta, a transactiv...
Elevated levels o...	Coronavirus disea...
Prognostic value ...	The inflammatory ...
IL-6 produced by ...	Trichomonas vagin...
Association betwe...	We aimed to compa...
Association of IL...	The -174G>C (rs18...
Interleukin-6 gen...	Several studies h...
IL-6 is present i...	IL-6 is a pro-inf...
IL-6 mediated JAK...	We investigated t...
Does serum interl...	The diagnosis of ...
Association of -1...	Interleukin-6 (IL...
Baseline Interleu...	The objective of ...
Interleukin-6 med...	The phosphoinosit...
Association betwe...	The association b...
Association of <i...	Autoimmune thyroi...
The Role of Inter...	Studies have show...
Association betwe...	
Association of Va...	Lung cancer is kn...

only showing top 20 rows

Figura 2: Dataframe articoli scientifici