

Primo progetto di Social Computing

Studenti:

Gabriele Dominici:

mail di ateneo: dominici.gabriele@spes.uniud.it
numero di matricola: 143859

Andrea Cantarutti:

mail di ateneo: cantarutti.andrea@spes.uniud.it
numero di matricola: 141808

Francesco Bombassei De Bona:

mail di ateneo: bombasseidebona.francesco@spes.uniud.it
numero di matricola: 144665

Lorenzo Bellina:

mail di ateneo: bellina.lorenzo@spes.uniud.it
numero di matricola: 142544

Novembre 2020

Indice

1	Introduzione	2
1.1	Strumenti utilizzati	2
2	Ottenimento dei dati	3
2.1	Endpoint interrogati	3
2.2	Procedimento adottato	4
3	Rappresentazione grafica	4
4	Ego Graph	5
5	Analisi della rete	6
6	Conclusioni	7
7	Appendice	7
7.1	Grafo completo alternativo	7
7.2	Sottografo completo di @damiano10	8

Capitolo 1

Introduzione

L'obiettivo del seguente progetto prevede la realizzazione di una rete sociale avente come protagonisti i cinque seguenti account Twitter:

- @mizzaro
- @damiano10
- @Miccighel_
- @eglu81
- @KevinRoitero

A tal proposito, per ciascuno di essi sono stati scaricati gli account *followers* e *following* i quali, assieme ad ulteriori utenti casuali a loro legati, hanno permesso di costituire l'insieme di nodi atto a realizzare la rete sociale preposta. Di questa è stato prodotto un grafo con visualizzazione interattiva, del quale sono stati osservati diversi parametri descrittivi.

1.1 Strumenti utilizzati

I processi di interrogazione degli endpoint di Twitter API e di produzione del grafo sono stati implementati in Python. A tal fine, sono state incluse le seguenti librerie:

- **Jupyter Lab**
- **Tweepy** (wrapper per l'interazione con il servizio REST offerto da Twitter)
- **NetworkX** (per realizzare il grafo diretto)
- **Pyvis** (per produrre una visualizzazione interattiva del grafo ottenuto).

Capitolo 2

Ottenimento dei dati

2.1 Endpoint interrogati

Al fine di ottenere le informazioni necessarie al completamento dei task, sono stati interrogati i seguenti endpoint:

- `get_user` (ottiene le informazioni specifiche relative ad un singolo utente)
- `followers` (ottiene una lista con informazioni relative ai followers dell'account riferito)
- `friends` (ottiene una lista con informazioni relative ai friends/followings dell'account riferito)
- `show_friendship` (ottiene la codifica della relazione che lega i due account riferiti, anche se inesistente).

In particolare, per quanto concerne il metodo **get_user**, in relazione a tutti gli utenti scaricati sono state selezionate le seguenti proprietà, al fine di soddisfare i fini analitici preposti:

- `ID` (*id univoco dell'account all'interno del social network*)
- `name` (*nome dell'account*)
- `screen_name` (*nome univoco dell'account*)
- `description` (*descrizione del profilo*)
- `followers_count` (*numero di followers*)
- `friends_count` (*numero di following*)
- `profile_image_url_https` (*link alla miniatura dell'immagine di profilo dell'account*).

Le richieste all'endpoint **show_friendships** sono, inoltre, state interpretate al fine di ricondurre la relazione fra due utenti (identificati dal loro ID come **source** e **target**) ad una delle tre seguenti categorie:

- **lr** (relazione di following da source a target)
- **rl** (relazione di following da target a source)
- **bi** (relazione di following bidirezionale fra source e target)

2.2 Procedimento adottato

Al fine di ottenere i dati necessari, sono state acquisite le informazioni relative ai cinque utenti riferiti nell'introduzione. Assieme ai dati ottenuti sono stati richiesti i rispettivi **friends** e **followers**. Sia per *friends* che per *followers*, inoltre, sono stati selezionati cinque utenti casuali, per ognuno dei quali sono stati richiesti i dati relativi a dieci utenti *followers* e dieci utenti *following*.

In seguito a tale procedimento, sono state ottenute le relazioni fra i nodi serializzati, al fine di identificare gli *archi* e la loro direzionalità. In particolare, sulla base del numero di relazioni che coinvolgono ogni nodo, è stato possibile determinarne un grado di centralità atto a proporzionare la relativa dimensione nel contesto di una visualizzazione interattiva.

I dati raccolti sono stati progressivamente serializzati in memoria in formati **.json** e **.pkl** ai path specificati.

Capitolo 3

Rappresentazione grafica

La rappresentazione grafica dell'intera rete è stata realizzata mediante la libreria *Pyvis*. Si osserva, in particolare, come i nodi rappresentanti i cinque profili iniziali siano distinguibili. La dimensione dei nodi è, infatti, stata resa proporzionale al peso degli stessi (definito dalla dimensionalità della relativa lista di adiacenza ($node["value"] = len(neighbor_map[node["id"]])$)).

Un'ulteriore e fondamentale distinzione è dovuta alla presenza dell'immagine profilo dell'account Twitter, ottenuta mediante l'attributo *profile_image_url_https*. Questa è stata assegnata solamente ai cinque nodi principali, al fine di renderli distinguibili all'interno della rete.

Trattandosi di un grafo orientato, ciascun arco è direzionato sulla base della casistica descritta nella sezione dedicata agli endpoint interrogati (2.1).

Il risultato ottenuto è osservabile di seguito:

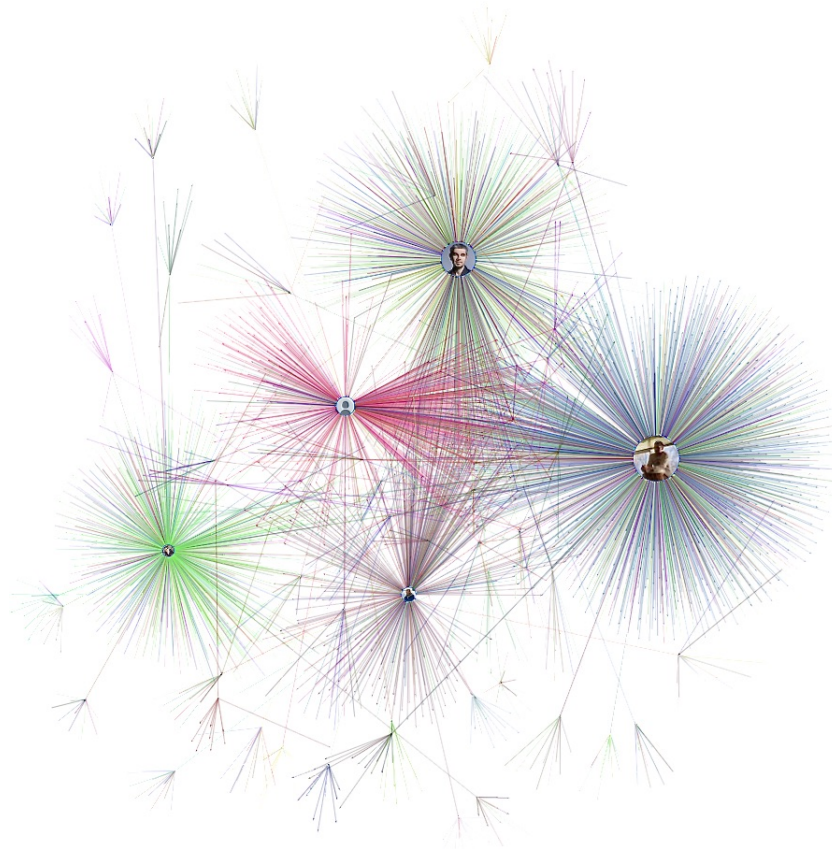


Figura 3.1: Rappresentazione grafica della rete sociale completa.

Capitolo 4

Ego Graph

In aggiunta alla rappresentazione visiva dell'intera rete, è stato realizzato un *ego_graph* specificatamente relativo al profilo @damiano10 che, sulla base dei dati osservati precedentemente (3), risulta essere l'utente più coinvolto all'interno della rete. I punti in comune con la rappresentazione generale, descritta nel capitolo precedente (3), sono molti.

Si osserva principalmente la presenza di hub corrispondenti agli altri quattro profili iniziali. Mentre nella rappresentazione generale un'osservazione simile sarebbe risultata scontata, in questa seconda raffigurazione ci fa intuire che i cinque profili iniziali siano legati a @damiano10 da diverse conoscenze comuni.

In seguito alla realizzazione di questo grafo, sono stati ricavati alcuni parametri analitici al fine di stabilirne la cricca massima (composta dai profili di @cikm2020, @damiano10, @eglu81, @Miccighel e @KevinRoitero) e la dimensione di questa, pari a 6.

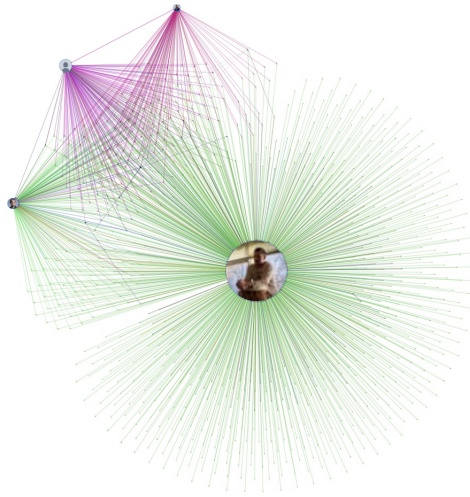


Figura 4.1: Rappresentazione grafica dell'ego graph avente per soggetto @damiano10.

Capitolo 5

Analisi della rete

In seguito alla realizzazione del grafo, sono stati analizzati i relativi parametri al fine di esaminare nel dettaglio la rete ottenuta. In primis, è stata effettuata un'interrogazione in relazione a **connessione** e **bipartizione**. Come previsto, il grafo è risultato connesso, poiché i cinque account di partenza sono in relazione diretta tra loro, garantendo, di conseguenza, tale proprietà. Negativo, invece, il riscontro relativo alla bipartizione.

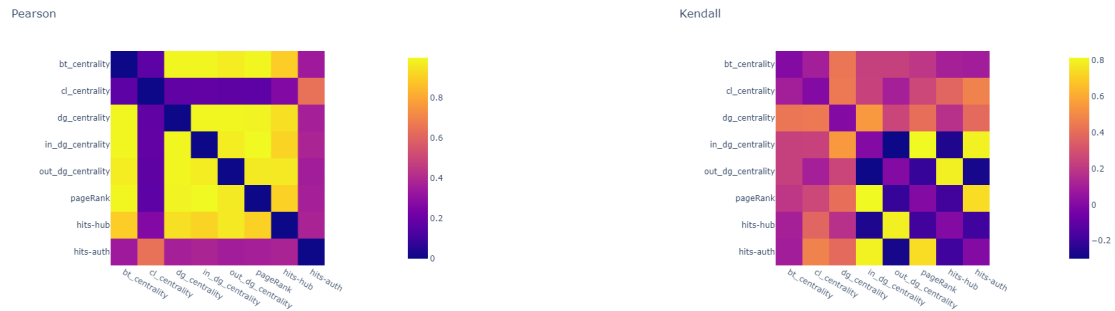
Sono, successivamente, stati calcolati i seguenti parametri di dimensionalità:

- **centro** (ovvero l'insieme di nodi dove la distanza massima dagli altri vertici è minima, composto da @damiano10, @eglu81, @KevinRoitero);
- **raggio** (come preventivato, di dimensione pari a 3);
- **diametro** (pari a 6 in conseguenza al valore assunto dal diametro).

In seguito, sono state calcolate misure di **centralità** in relazione ad ogni singolo nodo. I dati ottenuti sono stati riportati nei relativi metadati e, al fine di poterli visionare, sono stati resi disponibili nelle rispettive label durante la visualizzazione interattiva. Da questi si evince come pochi nodi (i cinque account di partenza) manifestino valori alti, contrariamente alla maggioranza che riporta, invece, valori bassi o nulli.

A seguire, è stata calcolata la **copertura minima**: sono stati selezionati gli archi in maniera tale da operare su nodi tra loro connessi, rinunciando di conseguenza alla connessione generale. Infine, sul grafo indiretto sono stati calcolati i coefficienti ω e σ , risultati rispettivamente 0.115515230000202260026 e 0.8340797127717022. In particolare, i parametri predisposti di default dalla libreria per il calcolo di entrambi i coefficienti sono stati reimpostati al fine di ridurre i notevolmente onerosi tempi di computazione (il numero di redirezioni casuali per ogni arco è stato ridotto a dieci, mentre il numero di grafi casuali è stato ridotto a due). I risultati ottenuti permettono di osservare come la rete tenda a rispecchiare il modello delle **small-world networks**,

considerando una minima imprecisione dovuta alla reimpostazione dei parametri sopracitata. Al fine di analizzare la correlazione fra i parametri sopra descritti, sono stati realizzati i seguenti grafici, rispettivamente relativi ai coefficienti di correlazione di **Pearson**(ρ) e **Kendall**(τ).



La vicinanza a 1 del valore di ρ implica una correlazione lineare tra le variabili esaminate. Il coefficiente τ , invece, ha permesso di studiare la concordanza fra i vari parametri.

Capitolo 6

Conclusioni

Sulla base delle analisi effettuate, è stato possibile osservare come la rete sociale si dimostri connessa e centralizzata rispetto ai cinque account di partenza (che sono i più rilevanti nel grafo), anche se non bipartita. Segue, inoltre, il modello delle small-world networks e permette, di conseguenza, ad ogni nodo di raggiungerne un altro con un numero limitato e ridotto di passaggi.

Tuttavia, non tenendo conto di tutte le relazioni tra nodi, la rete risulta essere poco rappresentativa. A tal fine, si è scelto di realizzare un ulteriore grafo più dettagliato, il cui risultato è stato esposto nell'appendice dell'elaborato, che esclude le limitazioni sui legami imposte in precedenza.

Capitolo 7

Appendice

7.1 Grafo completo alternativo

A partire dal risultato ottenuto sulla base del task, è stato realizzato un ulteriore grafo che prende in considerazione tutti i legami possibili tra i nodi della rete. Si osserva, nello specifico, il notevole

aumento del numero di archi e il cambiamento nella rappresentazione grafica, che risulta più densa ed articolata.

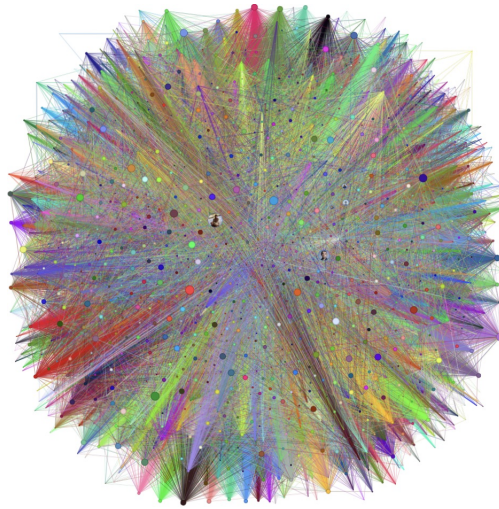


Figura 7.1: Rappresentazione grafica della rete completa includente tutti i legami.

Al fine di ottenere tutti i dati necessari, l'approccio impiegato per effettuare le richieste al servizio REST è stato modificato, includendo una doppia autenticazione e una parallelizzazione dell'accesso ai diversi endpoint. È possibile visionare una spiegazione più dettagliata all'interno del notebook. Come per le visualizzazioni precedenti, è possibile analizzare nel dettaglio i parametri analitici presentati in precedenza tramite le label di ogni singolo nodo, oppure utilizzando gli appositi script di codice riportati nel materiale allegato. In relazione a tali dati sono, inoltre, stati resi disponibili i relativi grafici statistici esplicativi (Pearson e Kendall).

7.2 Sottografo completo di @damiano10

Si osserva, inoltre, la variante del grafo precedente basata sul sottoinsieme di nodi in relazione con l'utente @damiano10.

