Università degli Studi di Padova

Corso di laurea triennale in
Ingegneria dell'Informazione

Analisi di una rete di autori di pubblicazioni scientifiche

Laureando:
Pietro Maria Nobili

Relatore:
Prof.ssa Cinzia Pizzi

Matricola:
1067941

Correlatore:
Dott. Mattia Samory

18 Luglio 2018
Anno Accademico 2017/2018

Indice

Abstract

L'interazione tra ricercatori è evidente nelle loro collaborazioni per la pubblicazione di articoli scientifici. La rete di co-autorialità tra ricercatori è pertanto un modello ricco di informazioni sulle dinamiche della ricerca scientifica.

In questa tesi si è sviluppato un sistema per ricostruire la rete di collaborazioni tra autori di pubblicazioni scientifiche afferenti al Dipartimento di Ingegneria dell'Informazione dell'università di Padova a partire dai dati di *Microsoft Academic Graph* gestendo problemi dovuti ad omonimie, presenza di più identificatori per un singolo autore, presenza di record con informazioni mancanti o errate.

Per il partizionamento in comunità, definite dall'appartenenza di un autore ad un dato Settore Scientifico Disciplinare, sono stati utilizzati tre metodi per classificare gli autori in comunità corrispondenti ai gruppi di ricerca: Girvan-Newman, Clauset-Newman-Moore, e Stocastica Block Model.

Le problematiche evidenziate nella fase di estrazione non consentono una mappatura perfetta con le aree di ricerca attive in dipartimento, ma la struttura macroscopica che si ricava dai grafi fornisce quantomeno un'indicazione di tale composizione, in particolare quando si utilizza l'algoritmo di Girvan e Newman.

Introduzione

L'analisi delle reti ha ricevuto molto interesse negli ultimi 20 anni, grazie alle sue numerose applicazioni in ambito scientifico e industriale in campi che spaziano dall'ecologia all'economia, dallo studio delle reti sociali alle neuroscienze, e più recentemente per le sue applicazioni agli studi di machine learning.

Le reti che descrivono fenomeni e situazioni naturali sono spesso caratterizzate da un'organizzazione interna. Studiando la struttura della rete si possono mettere in luce le relazioni e i meccanismi che la governano, ricavando utili indicazioni sul comportamento globale del sistema complesso studiato.

La struttura interna di una rete, come nel caso della rete sociale oggetto di questo studio, si può riassumere aggregando i nodi in comunità. L'individuazione accurata delle comunità è pertanto fondamentale per la descrizione della rete e per questo motivo è un importante oggetto di studio.

La ricostruzione accurate della rete delle pubblicazioni scientifiche permette una miglior comprensione dell'attività accademica. Le applicazioni spaziano dal valutare lo scambio di conoscenza tra i diversi settori di studio, alla misura dell'impatto a lungo termine delle ricerche, al tracciare la mobilità dei ricercatori [3]. Ciò è cruciale per permettere agli enti istituzionali ed ai responsabili politici di destinare risorse per promuovere l'eccellenza accademica.

In questo contesto è stato sviluppato un sistema per l'estrazione di informazioni da record di pubblicazioni per creare una rete di associazioni tra co-autori afferenti al Dipartimento di Ingegneria dell'Informazione dell'Università di Padova e sono state ricostruite le corrispondenti comunità.

L'elaborato segue la seguente struttura. Nella prima sezione viene descritto il problema del community detection. Vengono descritti tre dei metodi esistenti per partizionare un grafo. I primo due metodi sono deterministici: l'algoritmo proposto da Girvan e Newman e l'algoritmo sviluppato da Clauset, Newman e Moore. Il terzo è un metodo probabilistico, lo *Stochastic Block Model*. Viene descritta la *V-measure*, una misura esterna utilizzata per valutare la validità dei cluster generati.

Nella seconda sezione viene descritta la struttura del *Microsoft Academic Graph* da cui sono stati estratti i dati, il metodo che si è usato per estrarli e i problemi riscontrati. Vengono proposti e implementati dei metodi alternativi per risolverli.

Nella terza sezione viene descritto il metodo con cui sono generate le comunità e vengono presentati i risultati ottenuti.

Nell'ultima sezione si traggono le conclusioni sul lavoro svolto.

Il codice scritto per la manipolazione dei dati è disponibile alla pagina `github.com/Pitrified/authorship-network`.

Capitolo 1

Community detection

1.1 Definizione del problema

Le relazioni tra elementi di un sistema complesso possono essere rappresentate con un grafo. Gli elementi del sistema sono rappresentati dai nodi e le loro interazioni dagli archi. A livello globale, la topologia del grafo fornisce informazioni sul sistema. A livello intermedio fra il grafo nel suo insieme e le connessioni fra singoli nodi, molti grafi dimostrano un certo livello di organizzazione.

Una comunità viene definita come un insieme di nodi tali per cui i nodi di una stessa comunità sono fortemente connessi tra loro, mentre le connessioni tra nodi in diverse comunità sono meno probabili.

Questa definizione di comunità è stata preferita perché rispecchia la struttura della rete dei co-autori di pubblicazioni scientifiche, dove un alto numero di collaborazioni comporta in generale l'appartenenza allo stesso gruppo di ricerca o, a livello più macroscopico, allo stesso Settore Scientifico Disciplinare.

Nella sezione seguente, vengono presentati tre metodi per estrapolare la struttura in comunità di un grafo.

1.2 Metodi usati

Per partizionare il set di nodi di un grafo in cluster sono stati utilizzati tre algoritmi: l'algoritmo di Girvan-Newman [4] e l'algoritmo di Clauset-Newman-Moore [2], implementati nella libreria Stanford Network Analysis Platform (SNAP) [5], e l'algoritmo *Stochastic Block Model* [8], implementato in *graph-tool* [9].

Questi metodi sono stati scelti perché sono stati sviluppati considerando una definizione di comunità aderente a quella proposta nella sezione precedente, che è considerata descrivere adeguatamente la struttura interna di una rete di co-autori.

1.2.1 Girvan–Newman

Girvan e Newman introducono la *Edge Betweenness* come misura di quanto un arco agisca da ponte fra due comunità in un grafo. Ispirandosi alla *Vertex Betweenness*, che valuta il numero di cammini minimi fra coppie di nodi che passano per il nodo in esame, definiscono la centralità di un arco come numero di cammini minimi tra i nodi del grafo che passano per tale arco.

Per suddividere il grafo in comunità, vengono progressivamente eliminati gli archi con centralità massima, dopo aver ricalcolato i cammini minimi modificati dall'eliminazione degli archi.

Il risultato dell'algoritmo è un dendrogramma, da cui si estrapola la struttura delle comunità del grafo.

Ha complessità $O(E^2N)$ nel caso peggiore, dove E è il numero di edge e N è il numero di nodi, che diventa $O(N^3)$ nel caso di un grafo sparso in cui $N \sim E$.

1.2.2 Clauset-Newman-Moore

Girvan e Newman introducono la *modularity* [6] come valutazione della qualità della suddivisione in cluster dei nodi di un grafo. La *modularity* viene definita come la frazione di edge in un grafo che connettono nodi nella stessa comunità meno la frazione di edge che connetterebbero quei nodi se gli archi del grafo fossero casuali ma la struttura in comunità fosse la stessa. Se la frazione di archi all'interno di una comunità non è maggiore di quella prevista in un grafo casuale, la *modularity* è nulla, se il grafo ha una struttura in comunità ben definita, il valore si avvicina a 1.

L'algoritmo sviluppato da Clauset, Newman e Moore utilizza un approccio *greedy* per trovare una suddivisione dei nodi che massimizzi la *modularity*. Partendo da una una suddivisione che consiste in un nodo per comunità, ad ogni iterazione si uniscono le due comunità la cui aggregazione comporta il massimo incremento della *modularity*.

L'algoritmo ha complessità $O(ED \log N)$ dove E è il numero di edge, N è il numero di nodi e D è la profondità del dendrogramma che descrive la struttura delle comunità del grafo. La complessità diventa $O(N \log^2(N))$ nel caso di un grafo sparso in cui $N \sim E$, e con struttura delle comunità gerarchica in cui $D \sim \log N$.

1.2.3 Stochastic Block Model

A differenza dei due metodi descritti nelle sezioni precedenti, l'algoritmo del *Stochastic Block Model* segue un approccio stocastico al problema del *community detection*. Inoltre questo algoritmo non favorisce necessariamente la struttura delle comunità formate da nodi strettamente connessi tra loro e meno connessi con membri delle altre comunità.

Il modello stocastico a blocchi generalizza la struttura di comunità di un grafo raggruppando i nodi in blocchi e definendo una matrice delle probabilità di esistenza degli archi tra nodi di ciascun blocco.

Il problema del partizionamento viene quindi convertito in un processo statistico di inferenza dei parametri del modello basandosi sul grafo osservato.

Anche questo algoritmo è implementato con una strategia *greedy* che risulta in una complessità $O(Nln^2(N))$.

Si evidenzia il fatto che il primo algoritmo descritto ha complessità $O(N^3)$, che lo rende inadeguato nel caso di reti numerose. I successivi algoritmi scalano meglio con la dimensione del grafo, avendo complessità $O(Nln^2(N))$.

1.3 V-measure

La *V-measure*, una misura esterna della validità dei cluster basata sull'entropia, presentata da Rosenberg e Hirschberg [10], è stata utilizzata per valutare la qualità delle partizioni dei grafi ricavati con i vari metodi descritti nelle sezioni successive.

In questo elaborato, viene considerata buona una partizione degli autori che rispecchi la loro suddivisione nei Settori Scientifici Disciplinari.

Come terminologia, si parlerà di due distinte partizioni del set di nodi da analizzare: un set di classi C, considerate la suddivisione reale dei nodi, e un set di cluster K per indicare la partizione generata.

Rosenberg e Hirschberg hanno sviluppato due concetti, l'omogeneità e la completezza, la cui media armonica è la *V-measure*.

Una partizione di cluster è considerata perfettamente omogenea quando i cluster contengono *solo* membri di una singola classe, come esemplificato nella Tabella 1.2.

In maniera simmetrica una partizione è perfettamente completa quando un singolo cluster contiene *tutti* i membri di una classe, come esemplificato nella Tabella 1.3.

Per valutare l'omogeneità di una partizione si considera l'entropia condizionata della distribuzione di classi dato il clustering generato $H(C|K)$. Nel caso di perfetta omogeneità questo valore è nullo in quanto il cluster contiene una singola classe, mentre è massimo e vale $H(C)$ quando la distribuzione delle classi all'interno del cluster è identica alla distribuzione delle classi nell'intero set, per cui il clustering non fornisce alcuna informazione aggiuntiva.
Considerando che il valore di entropia è massimo quando la partizione è pessima, si definisce l'omogeneità come

$$h = \begin{cases} 1 & \text{se } H(C) = 0 \\ 1 - \frac{H(C|K)}{H(C)} & \text{altrimenti} \end{cases} \tag{1.1}$$

Analogamente si definisce la completezza come

$$c = \begin{cases} 1 & \text{se } H(K) = 0 \\ 1 - \frac{H(K|C)}{H(K)} & \text{altrimenti} \end{cases} \tag{1.2}$$

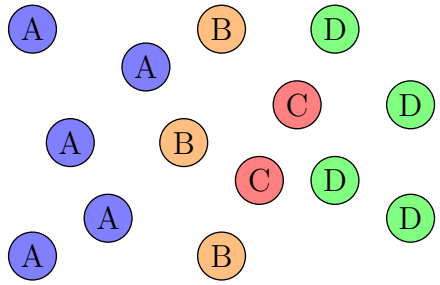
La *V-measure* è definita come la media armonica dei due valori di omogeneità e completezza

$$V = \frac{h \cdot c}{h + c} \tag{1.3}$$

I seguenti diagrammi illustrano graficamente le tre situazioni limite. Le classi sono {A, B, C, D}, i colori rappresentano i cluster.

Tabella 1.1: Clustering perfetto

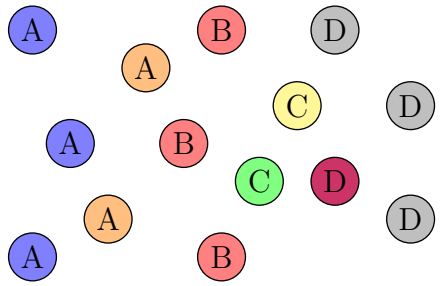
h	1
c	1
V	1



Classe	A	A	A	A	A	B	B	B	C	C	D	D	D	D
Cluster	1	1	1	1	1	2	2	2	3	3	4	4	4	4

Tabella 1.2: Clustering omogeneo ma non completo

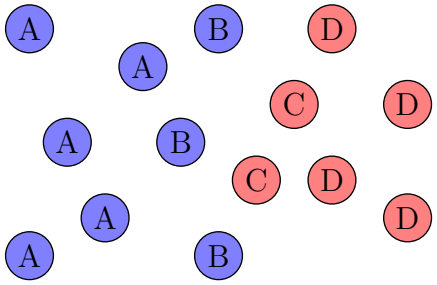
h	1
c	0.727
V	0.842



Classe	A	A	A	A	A	B	B	B	C	C	D	D	D	D
Cluster	1	1	1	2	2	3	3	3	4	5	6	7	7	7

Tabella 1.3: Clustering completo ma non omogeneo

h	0.512
c	1
V	0.677



Classe	A	A	A	A	A	B	B	B	C	C	D	D	D	D
Cluster	1	1	1	1	1	1	1	1	2	2	2	2	2	2

L'algoritmo utilizzato per calcolare la *V-measure* è `homogeneity_completeness_v_measure`, incluso nella libreria *scikit-learn* [7].

Capitolo 2

Estrazione dati

In questo capitolo viene descritta la struttura del database *Microsoft*. Viene descritto il metodo di estrazione dei dati, il processo di eliminazione dei dati spuri e di aggregazione dei nodi. Infine vengono riassunti i problemi intrinseci rilevati nel database.

2.1 Struttura database Microsoft

Il database da cui sono stati estratti i dati è il Microsoft Academic Graph [11], che contiene informazioni relative a pubblicazioni scientifiche, autori, istituzioni accademiche, riviste, conferenze e settori di studio. I record presenti nei file forniscono le relazioni tra queste entità.

Il database è composto da undici file di testo che contengono un record per riga, con i campi separati da tabulazione.

Per il lavoro oggetto di questa tesi sono stati utilizzati in particolare quattro file del database, la cui struttura è illustrata nella tabella 2.1.

La versione del database *Microsoft* utilizzato è quella dell'agosto 2015.

Tabella 2.1: Struttura del database

Nome file (Numero record)	Campi
Authors.txt (123.017.489)	Author ID Author Name
PaperAuthorAffiliations.txt (325.498.063)	Paper ID Author ID Affiliation ID Original affiliation name Normalized affiliation name Author sequence number

Affiliations.txt (2.719.436)	Affiliation ID Affiliation name
Papers.txt (122.695.085)	Paper ID Original paper title Normalized paper title Paper publish year Paper publish date Paper Document Object Identifier (DOI) Original venue name Normalized venue name Journal ID mapped to venue name Conference series ID mapped to venue name Paper rank

2.2 Processo di estrazione

Dal sito di dipartimento (`http://www.dei.unipd.it/lista-docenti`) sono stati estratti i nomi degli attuali afferenti DEI, includendo Docenti, Assegnisti di ricerca, Collaboratori di ricerca e Dottorandi. Questa lista è necessaria per avere un set di nomi degli autori afferenti al DEI da cui partire per estrarre i dati.

Si è reso necessario intervenire manualmente sulla lista per risolvere alcune criticità:

- I caratteri Unicode non sono ben gestiti nel database, a volte si trova la lettera non accentata, altre volte è presente un apostrofo invece dell'accento, in altri casi il carattere è assente del tutto. I nomi che contengono tali caratteri sono stati inseriti in tutte le varianti identificate.

- Afferenti con nomi composti uniti da trattino o con molteplici secondi nomi sono stati inseriti in multiple varianti

La lista ottenuta comprende 379 autori. Dove presente, è stato estratto il Settore Scientifico Disciplinare dell'afferente, che ha fornito la partizione in classi utilizzata come rifermento alla fine dell'elaborazione dei dati. Il Settore Scientifico Disciplinare è una categoria definita dal MIUR che codifica l'afferenza disciplinare dei docenti universitari italiani. Nel caso di assegnisti e dottorandi, è stata assegnata come classe *"N/A"*, che include 204 nomi.

I nomi propri degli autori sono stati abbreviati in tutte le possibili combinazioni per rispecchiare la struttura del database Microsoft. Se nel momento dell'abbreviazione si generano conflitti fra possibili etichette del nome, viene associata l'etichetta *"Multipli"*. Il file generato ha la struttura indicata nella tabella 2.3.

Tabella 2.3: PersoneComunitaDEI.txt

```
a a pietracaprina             INF/01 - INFORMATICA
a alberto pietracaprina       INF/01 - INFORMATICA
andrea a pietracaprina        INF/01 - INFORMATICA
andrea alberto pietracaprina INF/01 - INFORMATICA
...
mattia zorzi                  INF/04 - AUTOMATICA
michele zorzi                 INF/03 - TELECOMUNICAZIONI
m zorzi                       Multipli
...
```

Utilizzando questa lista di nomi ed abbreviazioni, sono stati estratti dal file *Authors.txt* le coppie (ID autore, nome autore), che risultano essere 8.135, con una media di 21,5 ID per nome.

Con il set di ID autori ottenuto, sono stati estratti dal file *PaperAuthorAffiliations.txt* i record relativi ai paper, nella forma (ID paper, ID autore, ID affiliation), per un totale di 62.291 paper.

Dalla lista così ottenuta, non necessariamente ordinata, sono stati creati gli edge fra ID autore, considerando una collaborazione tra autori per ciascun paper pubblicato assieme. Gli edge sono poi stati aggregati creando una edge list pesata, in cui i pesi corrispondono al numero totale di collaborazioni tra due autori. Questi due passaggi sono esemplificati nella tabella 2.4.

Tabella 2.4: Creazione edge

```
IDpaper1  IDautore1
IDpaper1  IDautore2          IDautore1  IDautore2  1
IDpaper1  IDautore3    →     IDautore1  IDautore3  1
IDpaper2  IDautore2          IDautore2  IDautore3  2
IDpaper2  IDautore3
```

A partire da questi edge è stato generato il primo grafo che rappresenta la rete degli autori.

Questo metodo di creazione del grafo, che considera le collaborazioni tra autori, riduce a 778 il set di ID autori, eliminando tutti i nodi sconnessi, che si assume siano riferiti ad omonimi di un afferente DEI. Nel grafo compaiono 287 nomi univoci, connessi da 1.830 edge con 7.803 di peso totale.

I grafi in questo capitolo sono stati divisi in comunità utilizzando l'algoritmo Modularity di Gephi [1].

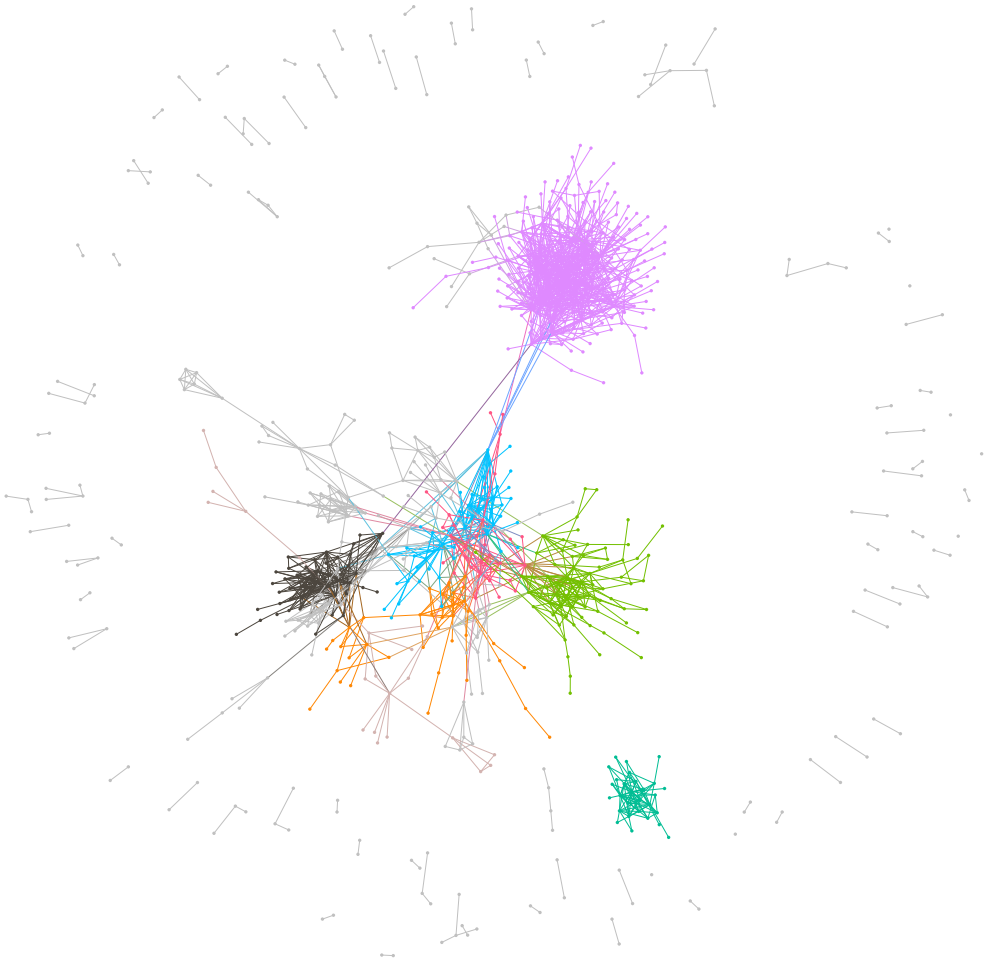


Figura 2.1: Grafo delle collaborazioni

Analisi della prima estrazione

I due problemi principali di questo processo emergono al momento dell'estrazione degli ID autore dal file *Authors.txt*:

- Vengono selezionati anche gli ID riferiti ad autori omonimi, non affiliati al DEI.
- Ad un singolo autore DEI sono associati più ID autore.

Nelle sezioni successive si presenteranno le soluzioni che sono state implementate per cercare di risolvere questi problemi.

2.2.1 Filtro degli autori per affiliation

Il primo problema è già in parte risolto dal metodo di creazione del grafo, che considera solo gli ID autori che hanno almeno una collaborazione con un altro ID nel set. In questo modo più del 90% degli ID viene filtrato.

Tuttavia, nel caso di collaborazioni tra omonimi di autori DEI, quantomeno in forma abbreviata, autori non del DEI non vengono filtrati.

Per ovviare a questo problema, è stato sviluppato un secondo metodo di estrazione dei dati che esclude gli ID autore se non hanno mai pubblicato un paper a Padova, considerando le informazioni sulle affiliation dei paper, come viene illustrato di seguito.

- Dal file *Affiliations.txt* si estrae la lista delle affiliation in cui risulta nel nome un match all'espressione regolare "$pad(ov|u)a$".
- Dalla lista di terne (ID paper, ID autore, ID affiliation) si mantengono solo quelle in cui l'ID affiliation compare nel set di affiliation padovane appena estratte.
- Dalle terne selezionate, si estrae un set di ID autori che hanno pubblicato almeno un paper con affiliation padovana.
- Si estraggono i paper scritti da questi ID autore e si procede alla generazione degli edge pesati.

Applicando questo metodo, gli ID autore si riducono a 306, relativi a 201 nomi univoci. Il grafo generato include 850 edge con un peso totale di 5.195.

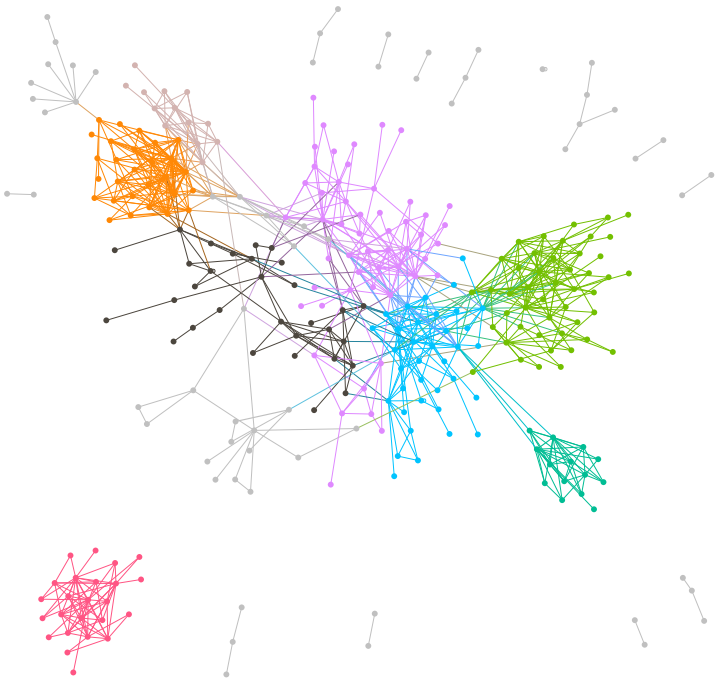


Figura 2.2: Grafo delle collaborazioni tra autori con nominativi associati alla lista di afferenti al DEI che hanno pubblicato almeno un lavoro con affiliation presente nella lista di affiliation padovane estratte

2.2.2 Unione di ID autore in singoli nodi

Per risolvere il secondo problema insorto nell'estrazione dei dati, per cui ad una singola persona fisica, effettivamente affiliata al DEI, possono essere associati più ID autore, sono state seguite tre vie alternative fra loro.

Il primo metodo tiene conto solo dei nomi associati ai nodi del grafo, il secondo e il terzo metodo sfruttano la struttura del grafo per stabilire se due nodi siano riferiti alla stessa persona.

Unione di nodi in base al nome

I grafi precedentemente generati vengono rielaborati, ipotizzando che nodi con nomi uguali o le cui abbreviazioni sono uguali possano essere considerati un unico nodo.

In questo modo "*a alberto pietracaprina*" viene associato a "*a a pietracaprina*" ma anche a "*andrea a pietracaprina*" come pure a "*andrea alberto pietracaprina*".

I nodi del grafo in figura 2.1 si riducono da 778 a 199. Gli edge scendono da 1830 a 661, mantenendo lo stesso peso totale di 7.803 (Figura 2.3).

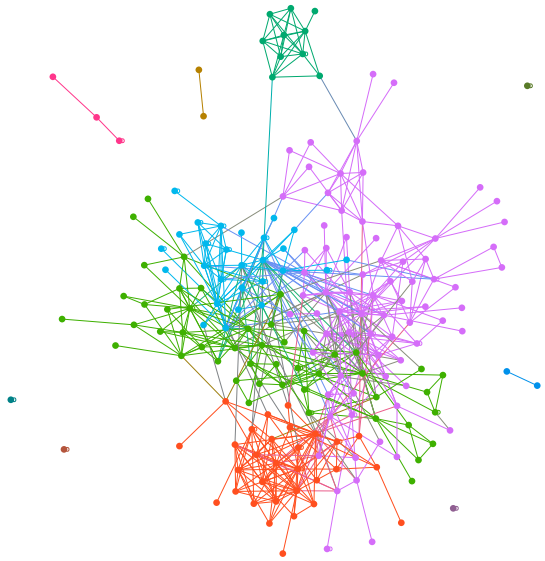


Figura 2.3: Grafo delle collaborazioni - nodi unificati per nome

I nodi del grafo in figura 2.2 si riducono da 306 a 158. Gli edge scendono da 850 a 463, mantenendo il peso totale di 5.195 (Figura 2.4).

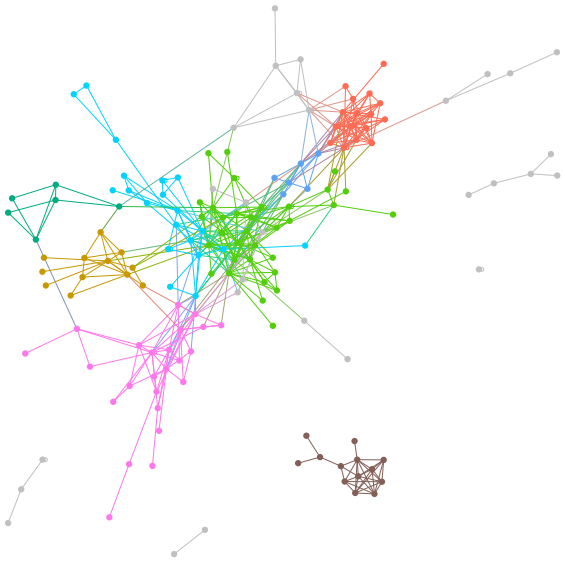


Figura 2.4: Grafo autori con affiliation padovana - nodi unificati per nome

Unione di nodi in base alla distanza

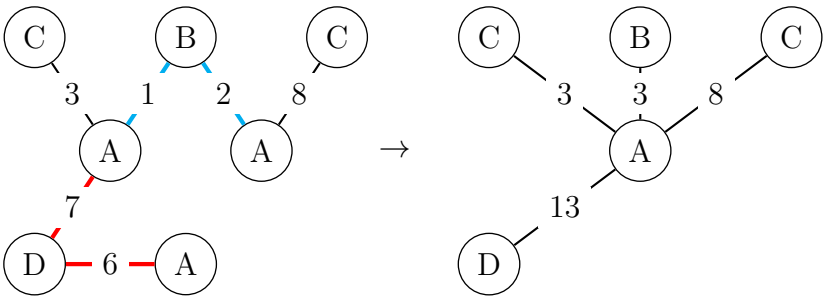
Il metodo che considera i nomi precedentemente descritto introduce una criticità nel caso un cui due autori abbiano un nome che viene abbreviato nello stesso modo. Questo fa sì che più persone vengano assimilate erroneamente in un unico nodo.

Nei dati che sono stati trattati questo succede solo nel caso di "*Mattia Zorzi*" e "*Michele Zorzi*", che si abbreviano entrambi in "*m zorzi*".

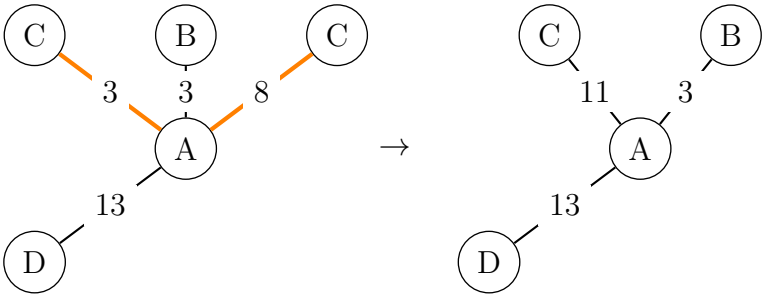
In dataset con bassa variabilità dei cognomi, o in generale in dataset più ampi, il fenomeno dei falsi positivi incide in maniera più marcata sulla veridicità del grafo ottenuto con questo metodo.

Nel secondo metodo sviluppato per unificare i nodi, viene richiesta un'ulteriore condizione per considerare due nodi come relativi alla stessa persona fisica. Oltre a verificare la corrispondenza delle abbreviazioni dei nomi si calcola anche la distanza minima tra i due nodi nel grafo. Se questa distanza è minore di una certa soglia, i due nodi vengono unificati. Questo processo può essere iterato più volte, per sfruttare le informazioni ottenute nei passi precedenti.

Prima iterazione unione nodi con distanza minima minore di 2



Seconda iterazione unione nodi con distanza minima minore di 2



I valori ottimi di soglia e numero di iterazioni sono stati cercati sperimentalmente, ma non è emersa dai risultati una coppia di valori migliore in maniera rilevante rispetto alle altre. Ispezionando manualmente il grafo, si è constatato che ID autore riferiti alla stessa persona risultano generalmente distanti 2. Il procedimento è stato ripetuto tre volte, in modo da sfruttare le informazioni generate nei passi predecenti. Dopo questo numero di iterazioni il grafo raggiunge quasi una situazione di stabilità, per cui nelle iterazioni successive non vengono uniti altri nodi.

I nodi del grafo in figura 2.1 si riducono da 778 a 336. Gli edge scendono da 1830 a 634, mantenendo lo stesso peso totale di 7.803 (Figura 2.5).

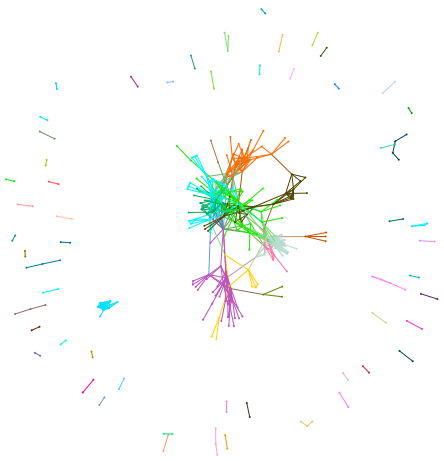


Figura 2.5: Grafo delle collaborazioni - nodi unificati per distanza

I nodi del grafo in figura 2.2 si riducono da 306 a 173. Gli edge scendono da 850 a 439, mantenendo il peso totale di 5.195 (Figura 2.6).

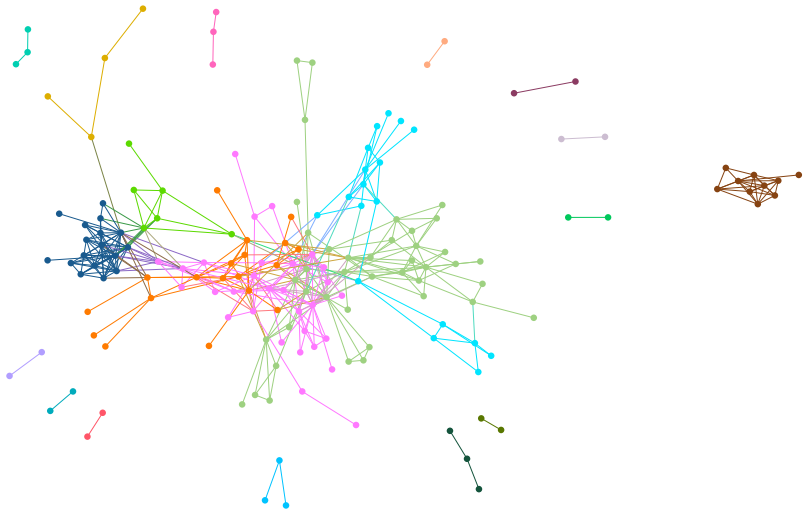


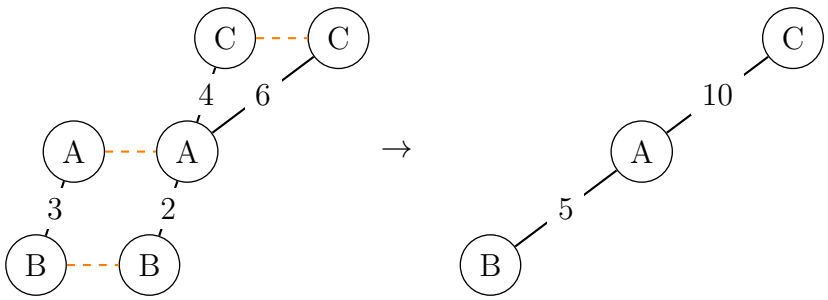
Figura 2.6: Grafo autori con affiliation padovana - nodi unificati per distanza

Unione di nodi in base a nodi adiacenti

Osservando i grafi generati senza unire i nodi, si nota la presenza di un elevato numero di componenti sconnesse formate da 2-4 autori, che sono principalmente effettivi collaboratori tra loro. È stato sviluppato un metodo di deduplicazione dei nodi che si basa sulle collaborazioni tra coppie di autori.

Per ogni edge si estraggono i nomi relativi agli estremi, se due edge hanno gli estremi con i nomi coincidenti, si considerano le coppie di nodi come relative alla stessa persona.

Unione dei nodi per coppie di nodi adiacenti



I nodi del grafo in figura 2.1 si riducono da 778 a 313. Gli edge scendono da 1830 a 615, mantenendo lo stesso peso totale di 7.803 (Figura 2.7).

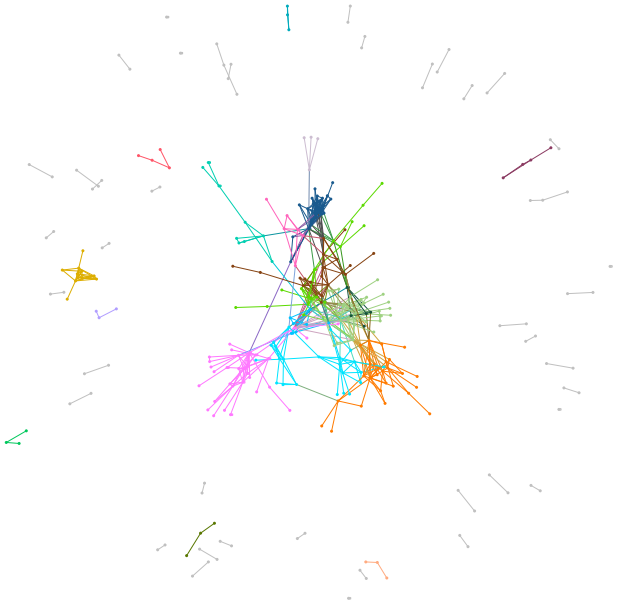


Figura 2.7: Grafo delle collaborazioni - nodi unificati per coppie di nodi adiacenti

I nodi del grafo in figura 2.2 si riducono da 306 a 174. Gli edge scendono da 850 a 453, mantenendo il peso totale di 5.195 (Figura 2.8).

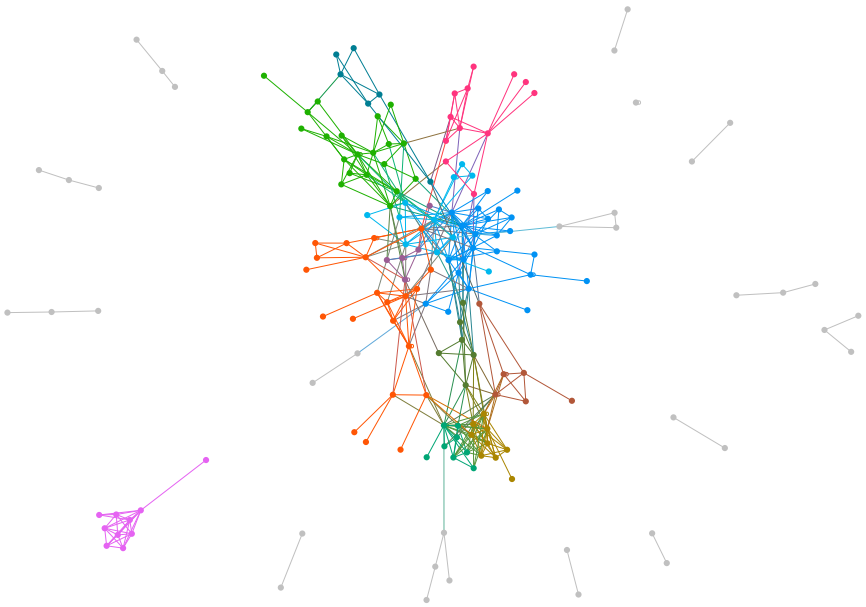


Figura 2.8: Grafo autori con affiliation padovana - nodi unificati per coppie di nodi adiacenti

2.2.3 Errori intrinseci al database

Nel corso delle varie iterazioni dell'estrazione dei dati, sono state identificate varie lacune nei record del database, riassunte di seguito

- I record di Paper-Autore-Affiliation non contengono l'ID affiliation in 264.069.014 record su 325.498.062. In particolare per autori con pochi paper, la fase di pruning basata sulle affiliation è molto influenzata da questa mancanza.

- Nei record Paper-Autore-Affiliation, può mancare il record relativo a uno dei co-autori. In altri casi l'ID autore presente può essere errato.

- Tra i co-autori dei un paper, possono essere riportati più ID autore riferiti alla stessa persona.

- I nomi con caratteri non ASCII sono trattati in maniera imprevedibile.

Nella prossima sezione si propongono dei metodi alternativi di estrazione dei dati per tentare di aggirare questi problemi.

2.3 Troubleshooting

Il grafo generato attraverso gli accorgimenti discussi in precedenza presenta ancora delle carenze.

In particolare, la lista di partenza include solo gli attuali afferenti DEI, estratti dal sito di dipartimento. Questo comporta la mancanza di professori, non più attivi a Padova, che avevano ruolo di aggregatore di una comunità.

Ad esempio, inserendo manualmente Alberto Apostolico, prolifico autore di paper all'Università di Padova, nella lista di partenza, il grafo, in prossimità del professore, cambia in maniera rilevante, come illustrato in figura 2.9. I nodi non sono stati uniti per evidenziare come l'assenza dei nodi riferiti al professor Apostolico influenzerebbe la deduplicazione dei nodi. Utilizzando questa procedura, emerge una comunità di collaboratori che precendentemente erano distanti nel grafo.

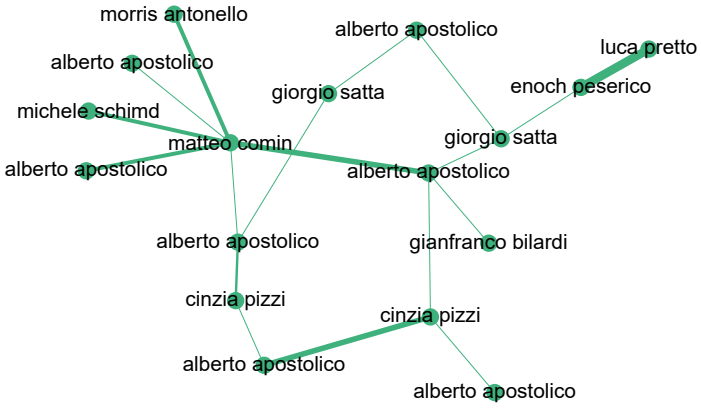


Figura 2.9: Rete di collaborazioni ottenuta aggiungendo manualmente il nome del professor Apostolico

2.3.1 Metodo per ampliare il set di autori

Un metodo generale proposto per identificare anche autori non presenti nella lista degli attuali afferenti DEI è:

1. Estrarre gli ID autore dalla lista di partenza
2. Estrarre le terne di ID paper-autore-affiliation
3. Estrarre tutti i record paper-autore-affiliation relativi ai paper identificati al punto precedente
4. Estrarre una nuova lista di ID autore dai paper appena estratti, ossia i co-autori degli autori nella lista utilizzata al punto 2 per estrarre le terne
5. Eventualmente ridurre il set di autori basandosi sulle affiliation dei loro paper
6. Ripetere i passi 2-5 per un numero predefinito di volte o fino alla convergenza della rete, nel momento in cui non vengono identificati ulteriori co-autori

In questo modo si estraggono tutte le comunità relative al set di affiliation usato come filtro. Una singola collaborazione con un dipartimento esterno comporta alle iterazioni successive l'inclusione di molti autori di quel dipartimento. Un metodo proposto per risolvere il problema e generare una rete di autori relativa solo al dipartimento di interesse è, alla fine delle iterazioni e della creazione dei cluster, considerare solo quelli che contengono almeno un nome presente nella lista originale. In questo modo dipartimenti esterni, anche se connessi al grafo, vengono filtrati.

Implementazione del metodo

Gli step 2,3 e 4 sono stati implementati, estraendo 3.845.835 record Paper-Autore-Affiliation, relativi a 56.450 paper univoci, da cui si ricavano 256.917 ID autore e 193.396 nomi univoci. Questi

valori decisamente elevati di co-autori e paper sono dovuti alla presenza di riviste, classificate come paper nel database, a cui hanno collaborato centinaia e anche migliaia di autori.

Filtrando i paper con più di 30 collaborazioni, il set di paper viene ridotto a 52.001 paper univoci, con 311.816 record Paper-Autore-Affiliation. Vengono identificati 148.124 ID autore per un totale di 112.440 nomi univoci. Da questa selezione di paper ed autori è stato generato un grafo applicando l'algoritmo di deduplicazione per nodi adiacenti. Un dettaglio di alcune collaborazioni ottenute in questo modo è incluso nella Figura 2.10.

Il metodo per nodi adiacenti si è rivelato valido nell'aggregare i dati estratti; ad esempio, nel grafo è presente un nodo etichettato *"Cinzia Pizzi"*, correttamente individuato e connesso con tutti i co-autori con cui la docente DEI ha collaborato. Sono inoltre presenti altri due nodi etichettati *"c pizzi"*, che si riferiscono a Claudio Pizzi, affiliato alla facoltà di Economia a Venezia e Cristina Pizzi, docente a Modena, che l'algoritmo ha mantenuto correttamente separati.

Tra la professoressa Cinzia Pizzi e un team di archeologi dell'Università di Milano, sono presenti connessioni spurie, causate da un record errato nel database, che attribuisce un paper a cui ha collaborato Chiara Pizzi all'ID autore della professoressa.

Il set di autori può essere ridotto, mantenendo solo gli autori che hanno pubblicato almeno un paper con affiliation padovana. In questo modo il set si riduce a 2.442 ID autori padovani per un totale di 2.094 nomi univoci, che hanno pubblicato 9.572 paper a cui corrispondono 23.417 record paper-autore-affiliation.

Per i motivi elencati nella sezione precedente, questo comporta una mancata identificazione di alcuni autori, i cui record paper-autore-affiliation non sono correttamente compilati. Per identificare anche questi autori, un possibile metodo proposto è di analizzare i vicini del nodo, considerandolo padovano se una certa frazione dei suoi vicini lo è.

Un altro metodo, proposto ma non implementato, per affinare la rete estratta è di, una volta deduplicati i nodi, scegliere fra quelli con lo stesso nome solo quello che si reputa essere più verosimilmente afferente al DEI.

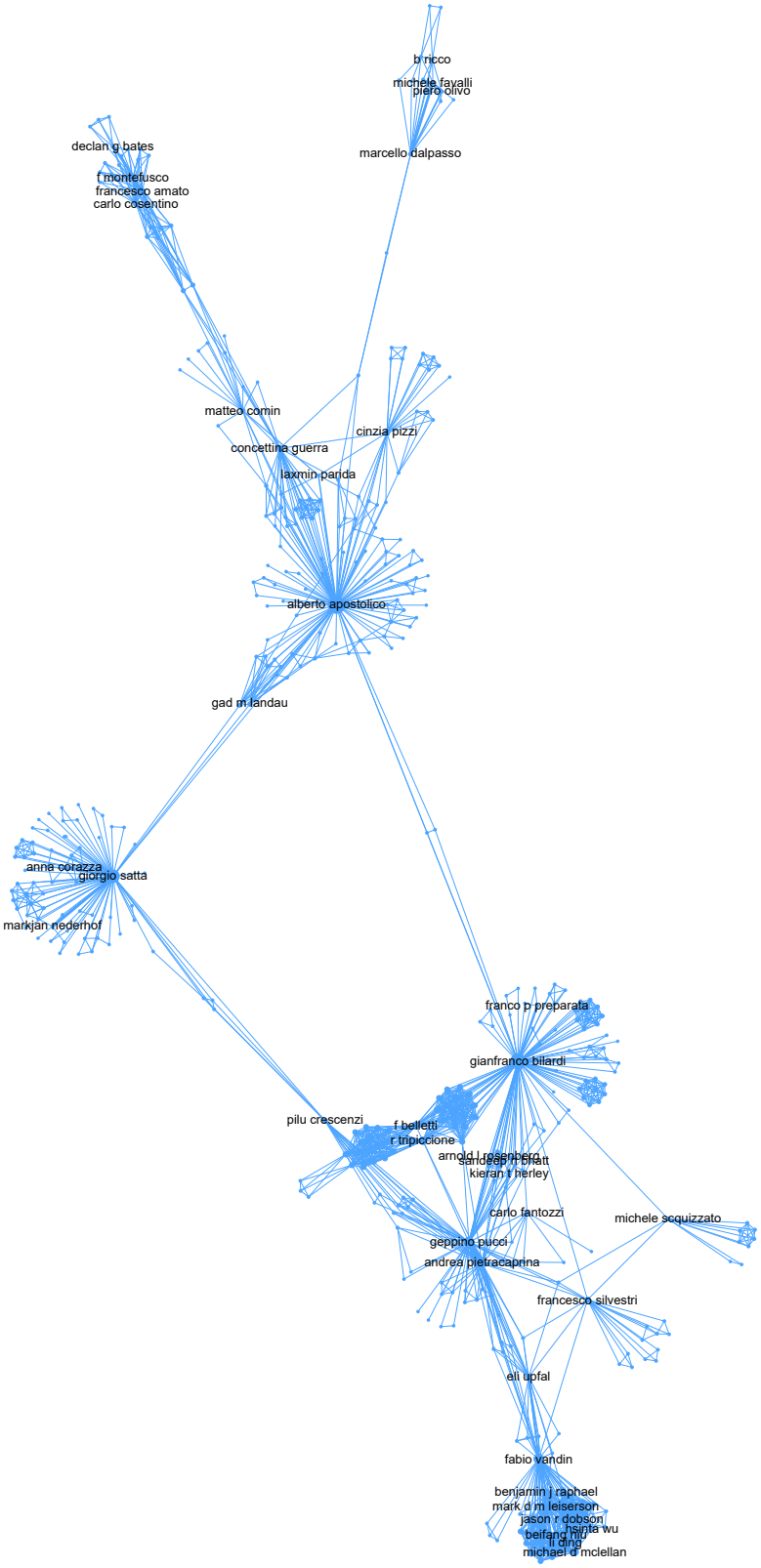


Figura 2.10: Dettaglio di alcune collaborazioni di ricerca ottenute ampliando il set degli autori

Capitolo 3

Risultati

3.1 Generazione delle comunità

I grafi generati secondo i metodi descritti nella sezione precedente sono stati suddivisi in comunità utilizzando gli algoritmi descritti nella sezione 1.2.

Le comunità sono inoltre state generate anche considerando solo la componene centrale dei grafi.

Il numero di comunità in cui viene suddiviso il grafo è calcolato dagli algoritmi utilizzati in modo da generare la suddivisione ottimale.

3.2 Analisi dei risultati

Per valutare la qualità delle partizioni generate, è stata calcolata la *V-measure* considerando il Settore Scientifico Disciplinare estratto dal sito di dipartimento come classe corretta di appartenenza degli autori.

Si è scelto di non considerare gli autori con etichetta *"N/A"* per il calcolo della *V-measure*, perché la classe corrispondente risulta frammentata in modo quasi uniforme tra i cluster, riducendo la completezza e quindi il valore di *V-measure* calcolato.

3.2.1 Dati generati

Nella tabella 3.1 sono indicati i valori di *V-measure* per tutte le possibili combinazioni di metodologia di estrazione paper, unione dei nodi e creazione dei cluster.

Tabella 3.1: Numero di cluster e valore di *V-measure* delle partizioni generate

Set paper	Unione	Algoritmo di clustering	Num. di cluster	*V-measure*
Tutti	Nomi	Blockmodel	4	0.51
		Clauset-Newman-Moore	15	0.63
		Girvan-Newman	18	0.57
		Blockmodel GC	6	0.54
		CNM Componente Centrale	9	0.62
		GN Componente Centrale	12	0.56
	Distanza	Blockmodel	10	0.36
		Clauset-Newman-Moore	64	0.44
		Girvan-Newman	64	0.48
		Blockmodel GC	6	0.44
		CNM Componente Centrale	8	0.40
		GN Componente Centrale	15	0.52
	Edge	Blockmodel	5	0.35
		Clauset-Newman-Moore	56	0.45
		Girvan-Newman	56	0.48
		Blockmodel GC	6	0.47
		CNM Componente Centrale	8	0.42
		GN Componente Centrale	15	0.50
Padovani	Nomi	Blockmodel	3	0.40
		Clauset-Newman-Moore	13	0.58
		Girvan-Newman	16	0.62
		Blockmodel GC	3	0.41
		CNM Componente Centrale	8	0.52
		GN Componente Centrale	11	0.61
	Distanza	Blockmodel	6	0.48
		Clauset-Newman-Moore	20	0.52
		Girvan-Newman	25	0.64
		Blockmodel GC	4	0.52
		CNM Componente Centrale	9	0.61
		GN Componente Centrale	12	0.63
	Edge	Blockmodel	5	0.49
		Clauset-Newman-Moore	20	0.52
		Girvan-Newman	25	0.65
		Blockmodel GC	6	0.60
		CNM Componente Centrale	9	0.63
		GN Componente Centrale	12	0.65

Gli stessi dati sono riportati graficamente in figura 3.1, utilizzando le seguenti etichette:
Tu: tutti gli autori estratti per collaborazione
Pa: autori con almeno un paper con affiliation padovana
No: Unione dei nodi in base ai nomi
Di: Unione dei nodi in base alla distanza
Ed: Unione dei nodi in base a nodi adiacenti
Bl: comunità generate dall'algoritmo Blockmodel
Cl: comunità generate dall'algoritmo Clauset-Newman-Moore
Gi: comunità generate dall'algoritmo Girvan-Newman
GC: indica l'elaborazione effettuata sulla Componente Centrale

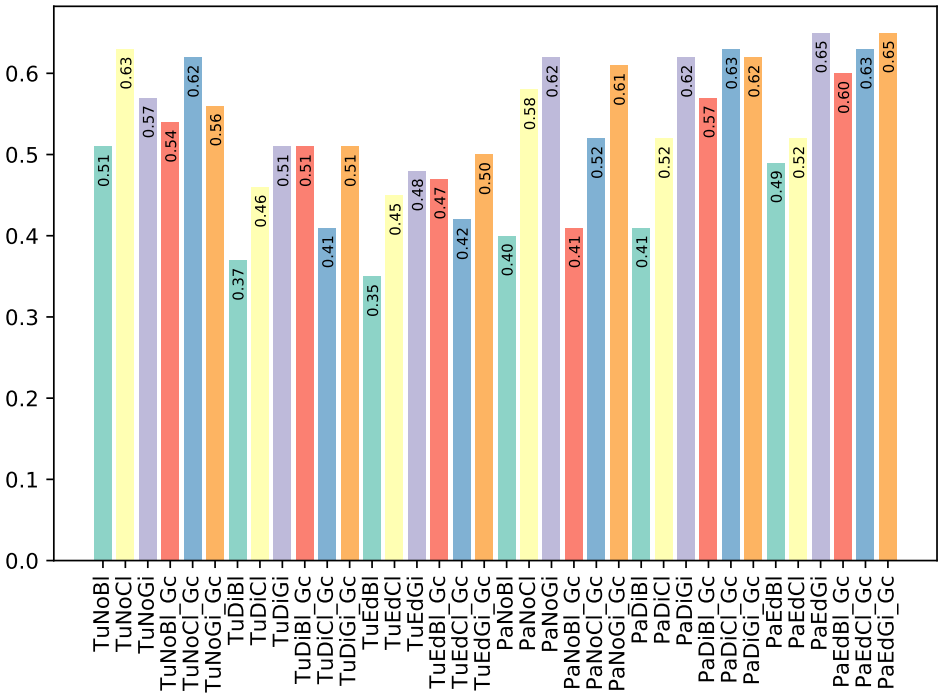


Figura 3.1: Valori di *V-measure* delle partizioni generate

3.2.2 Analisi

Confronto tra i metodi di estrazione

I valori di *V-measure*, aggregati per metodo di estrazione dei dati ed unione dei nodi, sono riassunti in figura 3.2.

I grafi generati considerando solo i paper scritti da autori con almeno un affiliation padovana hanno un valore medio di *V-measure* di 0.56, lievemente superiore al valore medio ottenuto dai grafi generati considerando tutti i paper estratti, pari a 0.49.

Il valore della *V-measure* del grafo generato considerando tutti i paper estratti ed unendo i nodi per nome è fra i più alti, ma ciò non corrisponde a un grafo ben rappresentativo della rete di autori del DEI, per i motivi illustrati nella sottosezione 2.2.1.

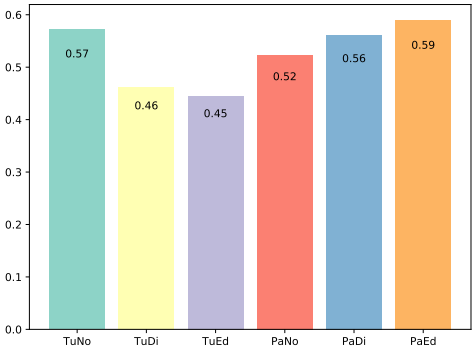


Figura 3.2: Medie dei valori di *V-measure* aggregate per metodo di estrazione degli autori e per metodo di unione dei nodi

Confronto tra i metodi di community detection

I valori di *V-measure*, aggregati per algoritmo di generazione dei cluster, sono riassunti in figura 3.3.

Il metodo Girvan-Newman si è rivelato essere il migliore in termini di *V-measure*, sia quando applicato all'intero grafo, sia quando applicato solo alla componente centrale dello stesso.

Nel caso del metodo Blockmodel, si nota una differenza tra l'analisi dell'intero grafo e della sua componente centrale, ed è stata rilevata una maggiore efficacia del metodo quando applicato solo alla componente centrale del grafo.

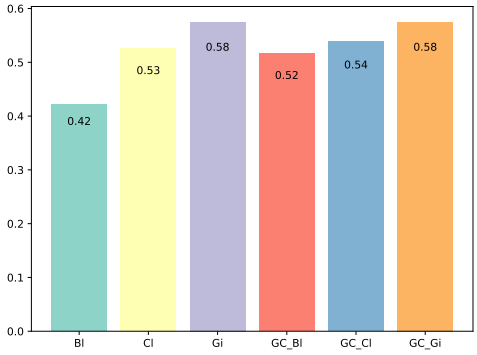
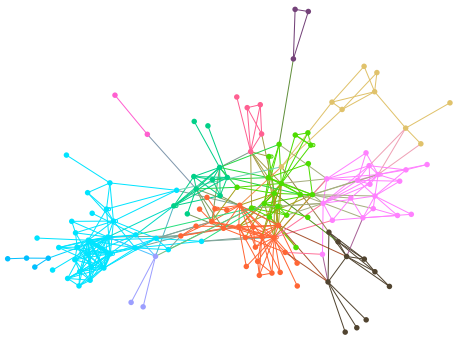


Figura 3.3: Medie dei valori di *V-measure* aggregate per algoritmo di generazione cluster

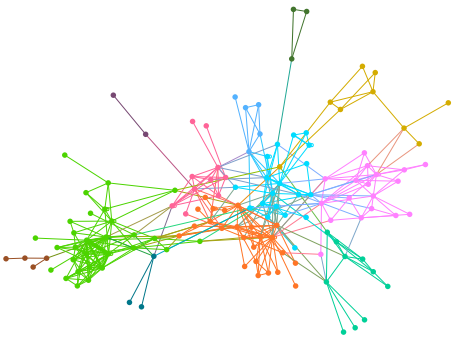
Nelle pagine successive sono riportati i grafi che hanno ottenuto i valori migliori di *V-measure*

In figura 3.4 sono presenti i grafi relativi ai grafi generati considerando solo gli autori con almeno un'affiliation padovana, unendo i nodi seguendo la strategia dei nodi adiacenti. In figura 3.5 i cluster degli stessi grafi sono stati aggregati per evidenziare le relazioni tra le comunità del grafo. Le figure evidenziano come le comunità generate, in particolare quelle di minore dimensione, siano discretamente omogenee, mentre quelle di maggiore dimensione siano più frazionate.

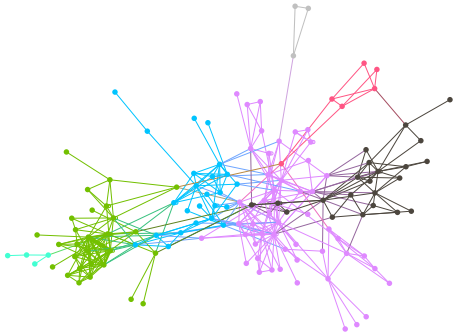
In figura 3.6 sono riportati i cluster ottenuti applicando l'algoritmo di Girvan e Newman alle componenti centrali dei grafi generati con i diversi approcci proposti.



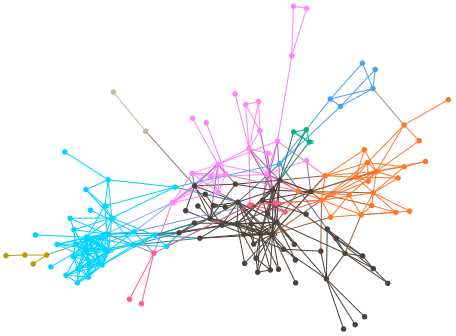
(a) Girvan Newman



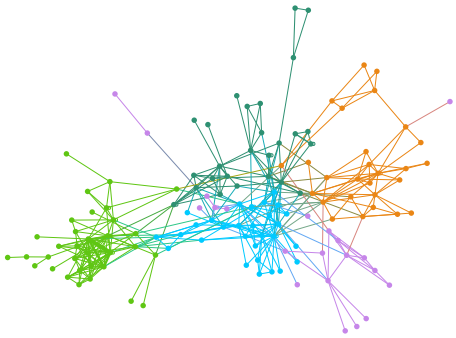
(b) GN - Componente centrale



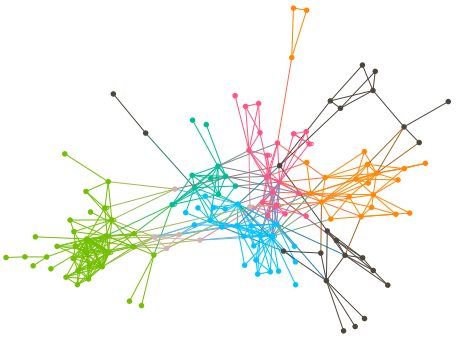
(c) Clauset Newman Moore



(d) CNM - componente centrale

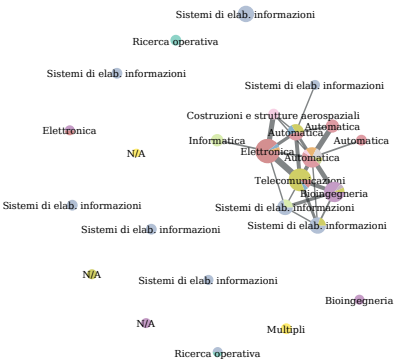


(e) Block Model

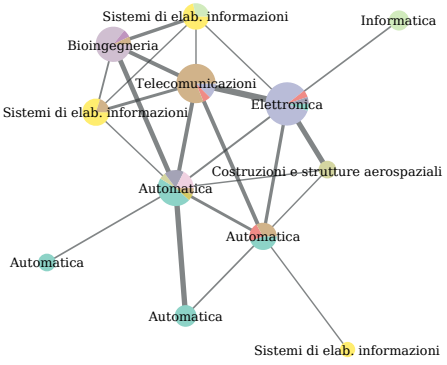


(f) BM - componente centrale

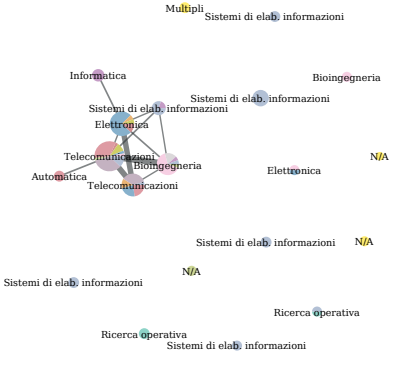
Figura 3.4: Confronto delle suddivisioni in cluster del grafo degli autori con almeno un'affiliation padovana, dove i nodi sono stati uniti utilizzando la strategia per nodi adiacenti



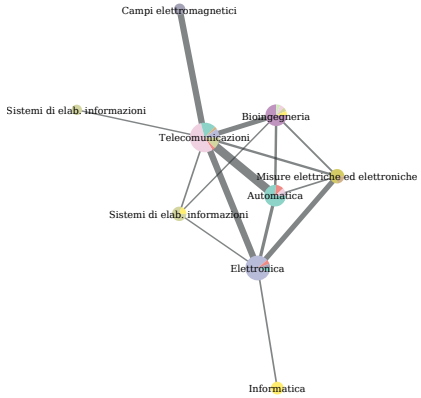
(a) Girvan Newman



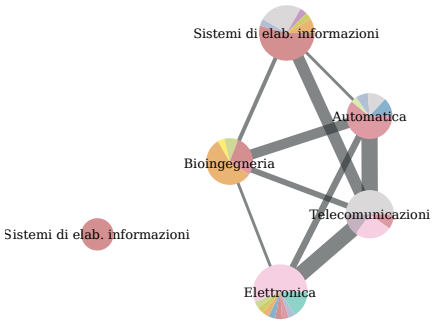
(b) GN - Componente centrale



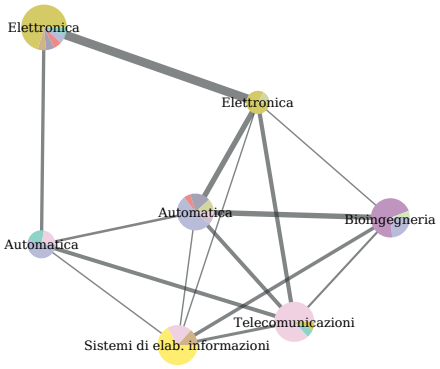
(c) Clauset Newman Moore



(d) CNM - componente centrale

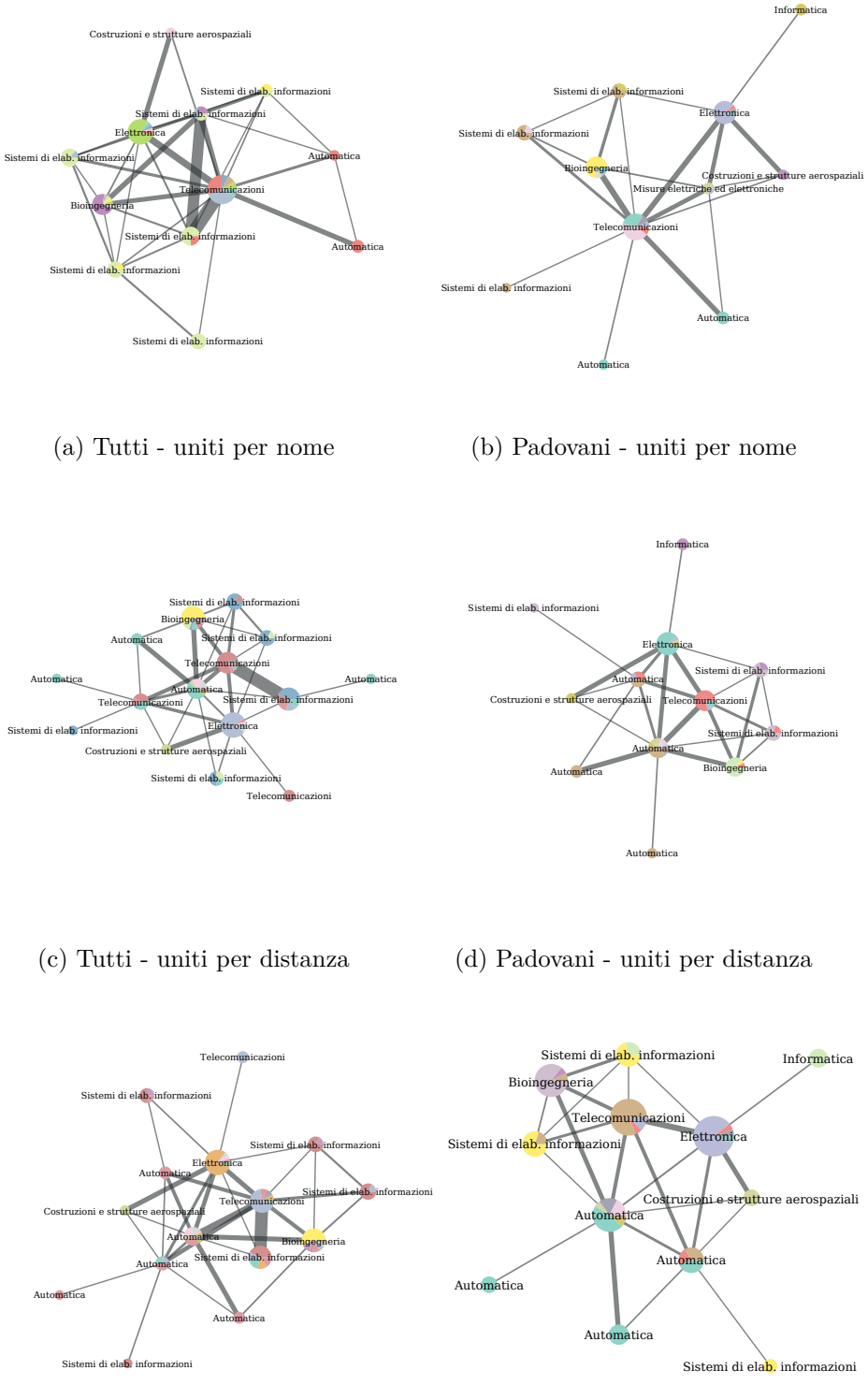


(e) Block Model



(f) BM - componente centrale

Figura 3.5: Confronto delle suddivisioni in cluster del grafo degli autori con almeno un'affiliation padovana, dove i nodi sono stati uniti utilizzando la strategia per nodi adiacenti; i cluster sono stati aggregati in nodi, i colori dei nodi indicano la suddivisione in classi dei cluster; l'etichetta corrisponde alla classe predominante nel cluster



(a) Tutti - uniti per nome

(b) Padovani - uniti per nome

(c) Tutti - uniti per distanza

(d) Padovani - uniti per distanza

(e) Tutti - uniti per nodi adiacenti

(f) Padovani - uniti per nodi adiacenti

Figura 3.6: Confronto delle suddivisioni in cluster generate dall'algoritmo Girvan-Newman applicato alla componente centrale del grafo; i cluster sono stati aggregati in nodi, i colori dei nodi indicano la suddivisione in classi dei cluster; l'etichetta corrisponde alla classe predominante nel cluster

Capitolo 4

Conclusioni

Questo lavoro si proponeva come obiettivo la creazione di una rete di co-autori di pubblicazioni scientifiche, in particolare degli autori afferenti al Dipartimento di Ingegneria dell'Informazione dell'Università di Padova.

L'approccio seguito è stato quello di estrarre dal database *Microsoft* le collaborazioni tra gli autori il cui nome figura nella lista di afferenti DEI. La rete generata è stata filtrata mantenendo solo gli autori con almeno una pubblicazione con affiliation padovana.

Nelle due reti generate in questo modo, compaiono nodi multipli associati allo stesso autore DEI. Sono stati sviluppati tre metodi di deduplicazione per aggregare i nodi relativi allo stesso autore. Il metodo dell'unione dei nodi per nome presenta delle lacune nel caso di omonimie. I metodi di unione dei nodi per distanza minima o per nodi adiacenti sono stati sviluppati per risolvere questo problema.

Le sei reti generate seguendo tutte le possibili combinazioni dei metodi precedenti sono state suddivise in cluster utilizzando tre algoritmi, applicati all'intero grafo o solo alla sua componente centrale.

Alcuni dei metodi seguiti hanno prodotto una rete di autori soddisfacente. In particolare quando la rete viene filtrata considerando gli autori con almeno un paper con affiliation padovana, gli autori estratti sono verosimilmente afferenti al DEI.

Le problematiche evidenziate (la presenza di nodi omonimi e di nodi multipli dello stesso autore) non consentono un'identificazione perfetta dei gruppi di ricerca, ma la struttura macroscopica che si ricava dai grafi fornisce un'indicazione della composizione del dipartimento, in particolare quando si utilizza l'algoritmo di Girvan e Newman.

Si è constatata l'importanza degli autori, non più afferenti dei, che in passato sono stati fulcro di una comunità.

Per cercare di identificare e includere questi autori, è stato proposto un metodo per ampliare il set di autori che considera i co-autori dei paper scritti dagli autori del dipartimento.

La prima iterazione di questo metodo proposto è stata implementata, ottenendo una rete molto completa ma che non è limitata al dipartimento.

Un futuro studio più approfondito delle strategie di rimozione dei nodi esterni può portare alla generazione di una rete che descrive in maniera accurata il dipartimento.

Bibliografia

[1] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 10:10008, October 2008.

[2] A. Clauset, M. E. J. Newman, and C. Moore. Finding community structure in very large networks. 70(6):066111, December 2004.

[3] Santo Fortunato, Carl T. Bergstrom, Katy Börner, James A. Evans, Dirk Helbing, Staša Milojević, Alexander M. Petersen, Filippo Radicchi, Roberta Sinatra, Brian Uzzi, Alessandro Vespignani, Ludo Waltman, Dashun Wang, and Albert-László Barabási. Science of science. *Science*, 359(6379), 2018. URL: `http://science.sciencemag.org/content/359/6379/eaao0185`.

[4] M. Girvan and M. E. J. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826, 2002. URL: `http://www.pnas.org/content/99/12/7821`.

[5] Jure Leskovec and Andrej Krevl. SNAP Datasets: Stanford large network dataset collection. `http://snap.stanford.edu/data`, June 2014.

[6] M. E. Newman and M. Girvan. Finding and evaluating community structure in networks. 69(2):026113, February 2004.

[7] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[8] T. P. Peixoto. Efficient Monte Carlo and greedy heuristic for the inference of stochastic block models. 89(1):012804, January 2014.

[9] Tiago P. Peixoto. The graph-tool python library. *figshare*, 2014. URL: http://figshare.com/articles/graph_tool/1164194.

[10] Andrew Rosenberg and Julia Hirschberg. V-measure: A conditional entropy-based external cluster evaluation measure. pages 410–420, 01 2007.

[11] Arnab Sinha, Zhihong Shen, Yang Song, Hao Ma, Darrin Eide, Bo-June (Paul) Hsu, and Kuansan Wang. An overview of microsoft academic service (mas) and applications. In *Proceedings of the 24th International Conference on World Wide Web*, WWW '15 Companion, pages 243–246, New York, NY, USA, 2015. ACM. URL: http://doi.acm.org/10.1145/2740908.2742839.