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

È stato generato un grafo delle pubblicazioni del DEI.

Sono stati cercati cluster nel grafo.

Sono stati confrontati con la struttura delle comunità del dipartimento.

Introduzione

Breve descrizione del community detection e della sua importanza, in generale e nel caso particolare delle comunità di autori di pubblicazioni scientifiche.

Presentazione della struttura della tesi.

Capitolo 1

Community detection

1.1 Studi precedenti

Descrizione delle comunità - edge più probabili tra nodi interni alla comunità e meno tra le comunità

Descrizione della modularity come metrica.

Descrizione della community detection.

Descrizione della bibliografia attuale sui grafi di coautori.

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 e l'algoritmo di Clauset-Newman-Moore, implementati nella libreria Stanford Network Analysis Platform (SNAP) [1], e l'algoritmo Stochastic Blockmodel, implementato in graph-tool [3].

1.2.1 Girvan–Newman

Descrizione metodo GN. Ha complessità $O(E^2N)$

1.2.2 Blockmodel

Descrizione metodo blockmodel. Ha complessità $O(Nln^2(N))$

1.2.3 Clauset-Newman-Moore

Descrizione metodo Clauset-Newman-Moore. Ha complessità $O(Nln^2(N))$

Capitolo 2

Estrazione dati

2.1 Struttura database Microsoft

Il database da cui sono stati estratti i dati è il Microsoft Academic Graph [5], 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.

L'ultimo aggiornamento del database disponibile risale all'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.

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.

I nomi propri degli autori sono stati abbreviati in tutte le possibili combinazioni per rispecchiare la struttura del database Microsoft, ed è stato creato un file con la struttura seguente:

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
...
```

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, che sono poi stati aggregati creando una edge list pesata, come indicato in 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 per un totale di 287 nomi univoci. Il grafo è formato 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.

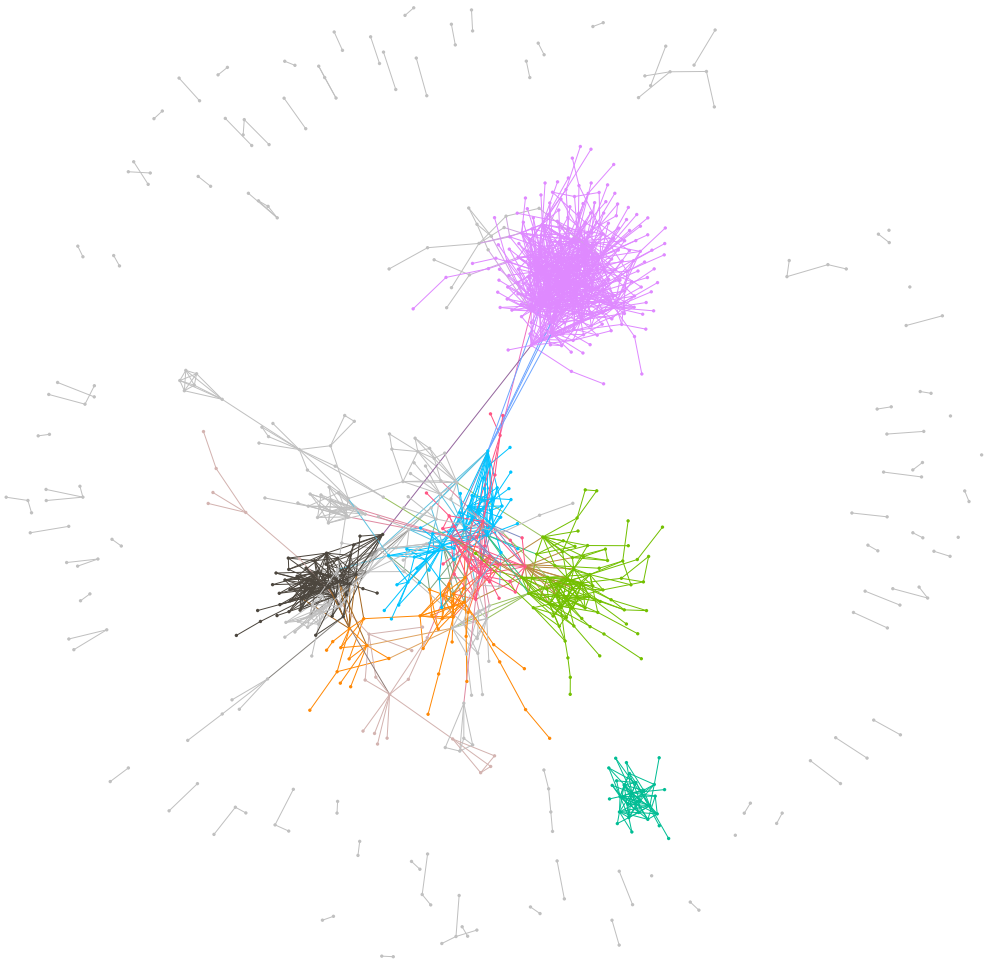


Figura 2.1: Grafo collaborazioni

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.

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.

Ispezionando manualmente il grafo, si rileva la presenza di due nodi riferiti a "c pizzi", che però sono connessi ad autori con cui la professoressa Pizzi non ha mai collaborato. Analizzando gli ID, le loro collaborazioni e i paper a cui hanno collaborato, si è dedotto che sono ID relativi probabilmente a Carmine Pizzi e Claudia Pizzi, medici rispettivamente a Bologna e Milano. Questi ID sono stati inclusi perché risultano collaborazioni con altri autori il cui nome è presente nella lista di partenza estratta dal sito di dipartimento.

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.

Gli ID autore riferiti a "c pizzi", non avendo neanche un paper con affiliation padovana, vengono correttamente filtrati.

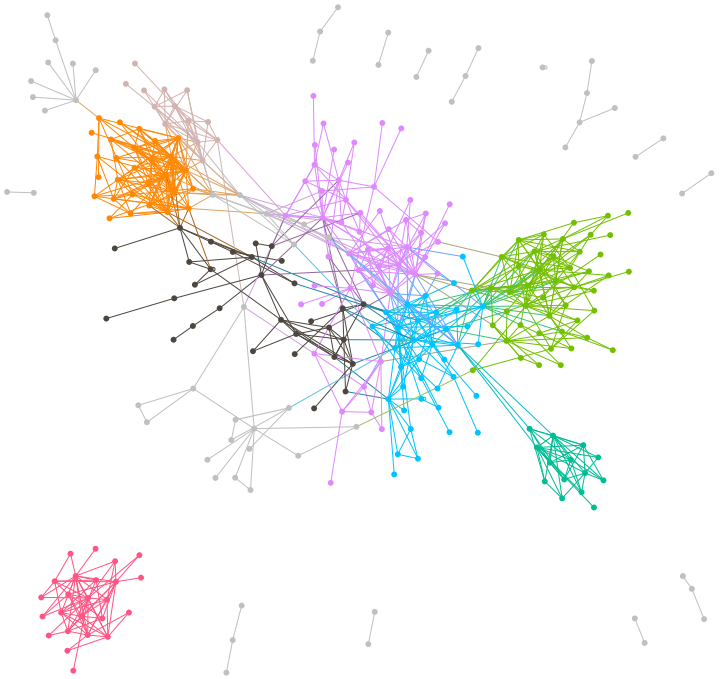


Figura 2.2: Grafo Padovani

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 due vie alternative fra loro.

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

Per 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.

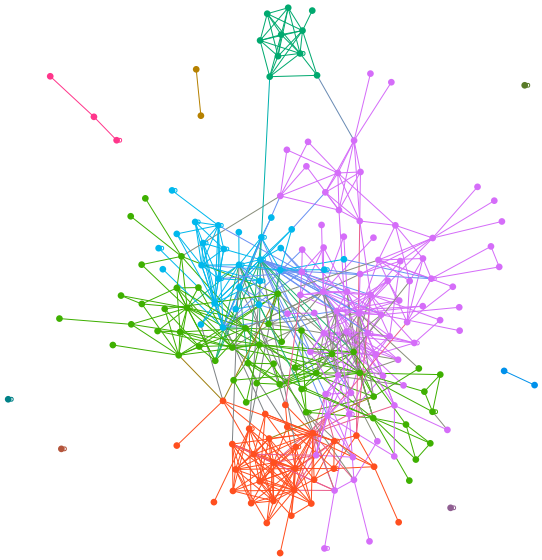


Figura 2.3: Grafo collaborazioni - nodi unificati per nome

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: Grafo padovani - nodi unificati per nome

Per distanza

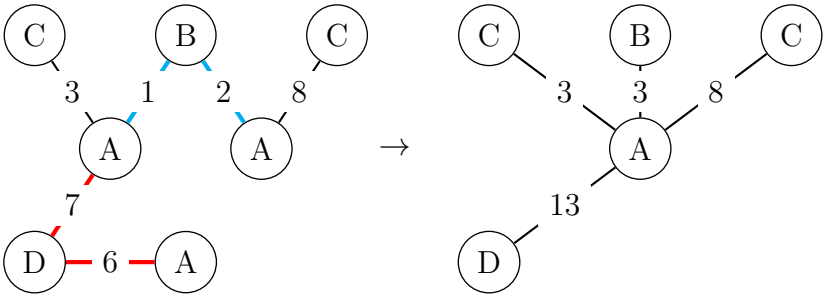
Il metodo che considera i nomi precedentemente descritto introduce una criticità nel caso un cui due autori abbiano un nome che viene abbreviato nello stesso modo. Questo fa sì che più persone vengano assimilate erroneamente in un unico nodo.

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

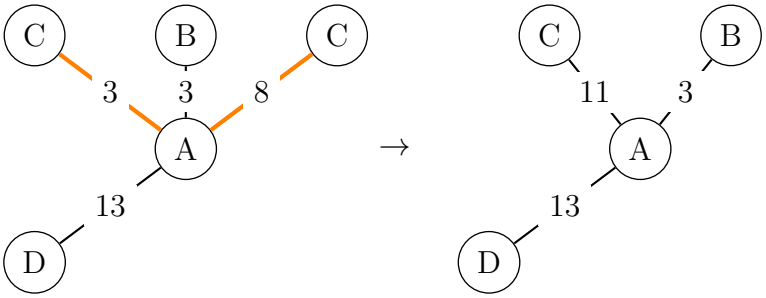
In dataset più ampi oppure relativi a comunità che anche se ristrette presentano una bassa variabilità dei cognomi, il fenomeno dei falsi positivi incide in maniera molto più marcata sulla veridicità del grafo.

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.

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.

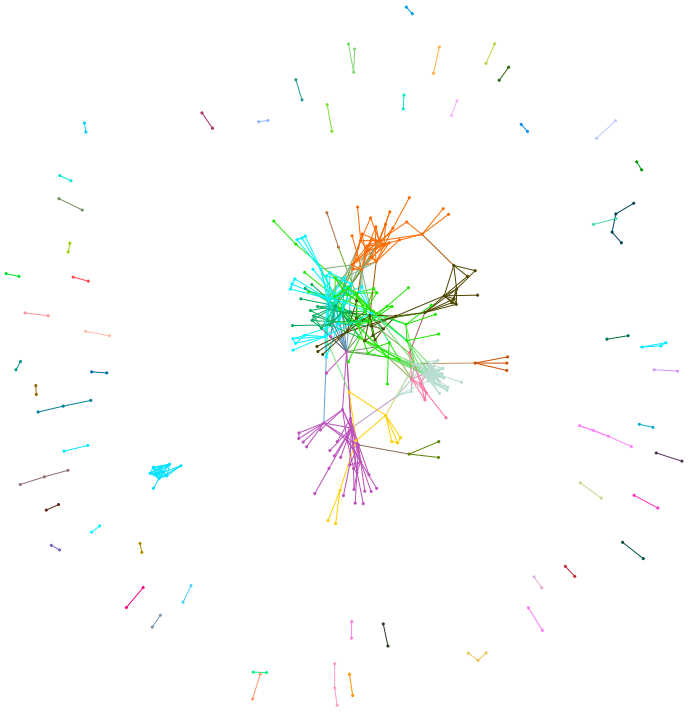


Figura 2.5: Grafo collaborazioni unificati per distanza

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.

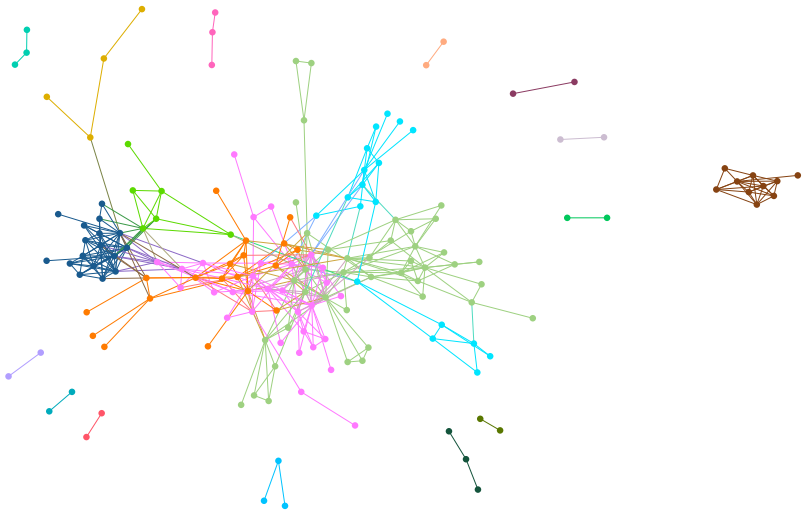


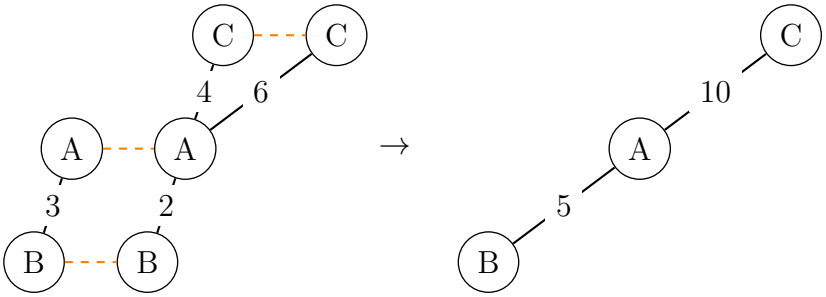
Figura 2.6: Grafo Padovani unificati per distanza

Per 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.

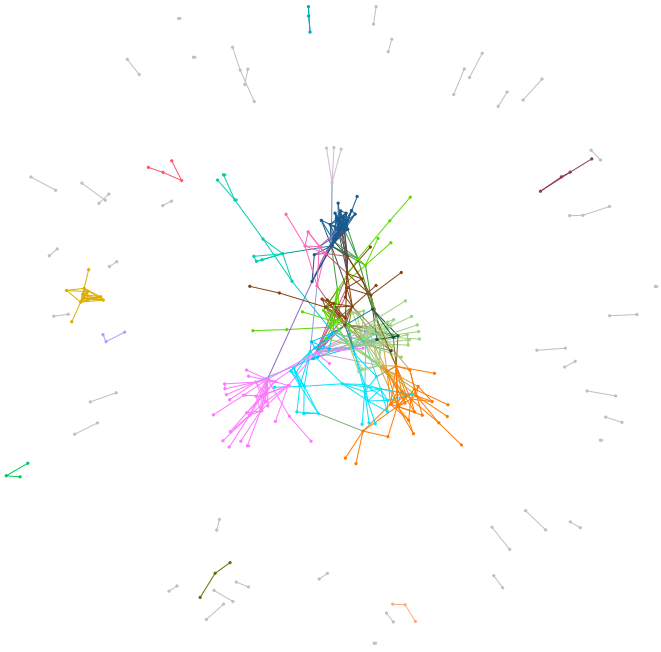


Figura 2.7: Grafo collaborazioni unificati per coppie di nodi adiacenti

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.

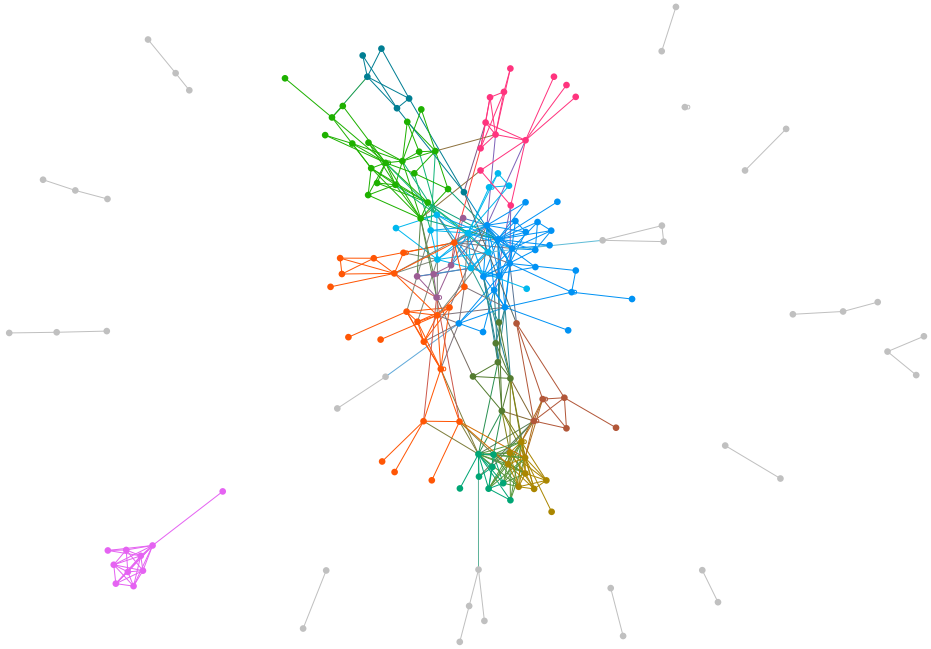


Figura 2.8: Grafo Padovani unificati per coppie di nodi adiacenti

2.2.3 Errori intrinseci al database

- affiliation non compilate autori grandi hanno un paper padovano che li mantiene, autori piccoli scompaiono potrei guardare i coautori dei paper se sono padovani
- erdos erds e caratteri speciali
- paper mancanti della professoressa
- self loop
- nomi abbreviati - database obsoleto

Capitolo 3

Troubleshooting

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

La lista di partenza include gli afferenti DEI attuali, mentre il database è aggiornato al 2015. Questo comporta la mancanza di professori, non più a Padova, che avevano ruolo di aggregatore di una comunità.

Un metodo proposto per includere i nomi mancanti è:

1. Estrarre gli ID autore
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 coautori della lista di partenza.
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

In questo modo si estraggono tutte le comunità relative alle affiliation usate 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 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, non vengono inclusi.

Per verificare l'efficacia di questo metodo, è stato analizzato il caso del professor Apostolico, prolifico autore di paper all'Università di Padova, che essendo mancato qualche anno fa non compare più nella lista degli attuali afferenti al DEI. Dopo aver reinserito il suo nome nella lista di autori, è stata rieseguita l'estrazione dei dati e rigenerato il grafo.

Come illustrato in figura 3.1, utilizzando questa procedura, emerge una comunità di collaboratori che precendentemente erano distanti nel grafo.

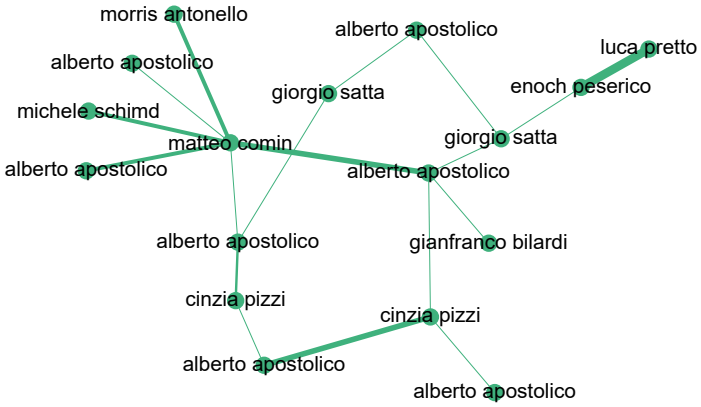


Figura 3.1: Rete di collaborazioni del professor Apostolico

Capitolo 4

Risultati

4.1 V-measure

La *V-measure*, una misura esterna della validità dei cluster basata sull'entropia, presentata da Rosenberg e Hirschberg [4], è stata utilizzata per valutare la qualità delle partizioni dei grafi ricavati con i vari metodi descritti nelle sezioni precedenti.

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.

In maniera simmetrica una partizione è perfettamente completa quando un singolo cluster contiene *tutti* i membri di una classe.

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

$$h = \begin{cases} 1 & \text{se } H(C) = 0 \\ 1 - \frac{H(C|K)}{H(C)} & \text{altrimenti} \end{cases} \tag{4.1}$$

Analogamente si definisce la completezza come

$$c = \begin{cases} 1 & \text{se } H(K) = 0 \\ 1 - \frac{H(K|C)}{H(K)} & \text{altrimenti} \end{cases} \tag{4.2}$$

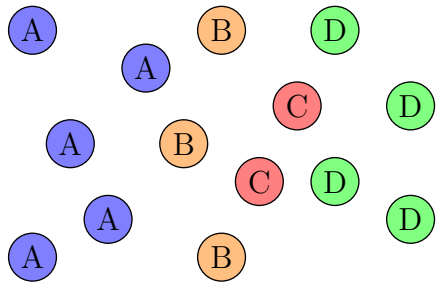
La *V-measure* è definita come la media armonica dei due valori di omogeneità e completezza

$$V = \frac{h \cdot c}{h + c} \tag{4.3}$$

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

Clustering perfetto

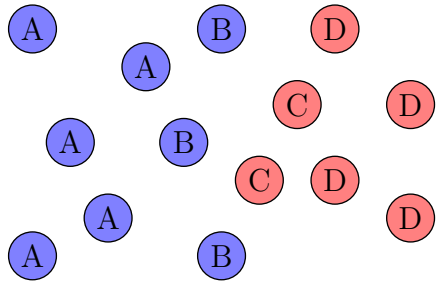
h	1
c	1
V	1



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

Clustering completo ma non omogeneo

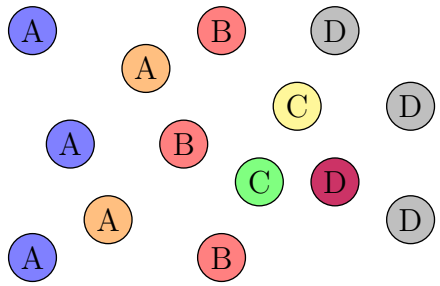
h	0.512
c	1
V	0.677



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

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

L'algoritmo utilizzato per calcolare la *V-measure* è `homogeneity_completeness_v_measure`, incluso nella libreria *scikit-learn* [2].

4.2 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.

4.2.1 Dati generati

Nella tabella 4.1 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 4.1: 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 Giant Component	9	0.62
		GN Giant Component	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 Giant Component	8	0.40
		GN Giant Component	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 Giant Component	8	0.42
		GN Giant Component	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 Giant Component	8	0.52
		GN Giant Component	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 Giant Component	9	0.61
		GN Giant Component	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 Giant Component	9	0.63
		GN Giant Component	12	0.65

Gli stessi dati sono riportati graficamente in figura 4.1, utilizzando le seguenti etichette:

Tu: Tutti; Pa: Padovani
No: Nomi; Di: Distanza; Ed: Edge
Bl: Blockmodel; Cl: Clauset-Newman-Moore; Gi: Girvan-Newman
GC indica l'elaborazione effettuata sulla Giant Component

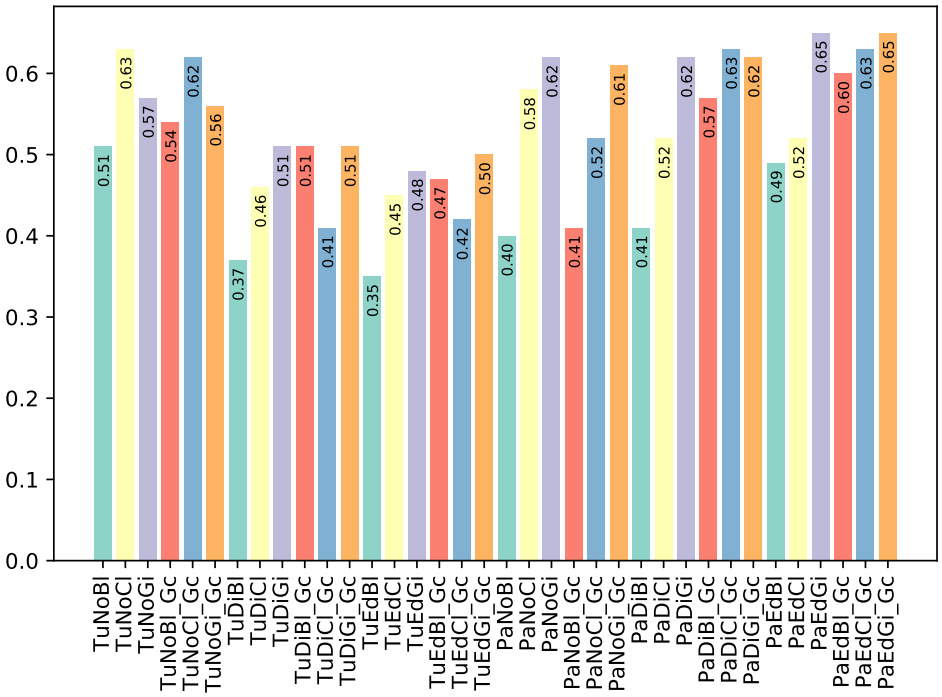


Figura 4.1: Valori di V-measure delle partizioni generate

4.2.2 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 4.2.

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 in 2.2.1.

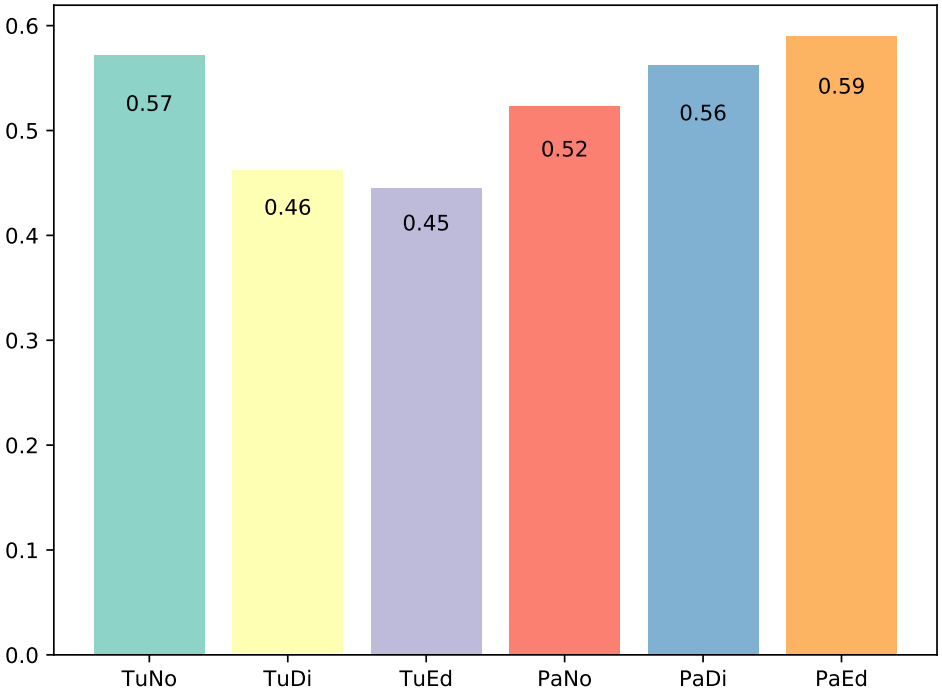


Figura 4.2: Medie dei valori di *V-measure* aggregate per metodo di estrazione

Confronto tra i metodi di community detection

I valori di *V-measure*, aggregati per algoritmo di generazione dei cluster, sono riassunti in figura 4.3.

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 migliore efficacia del metodo nel secondo caso.

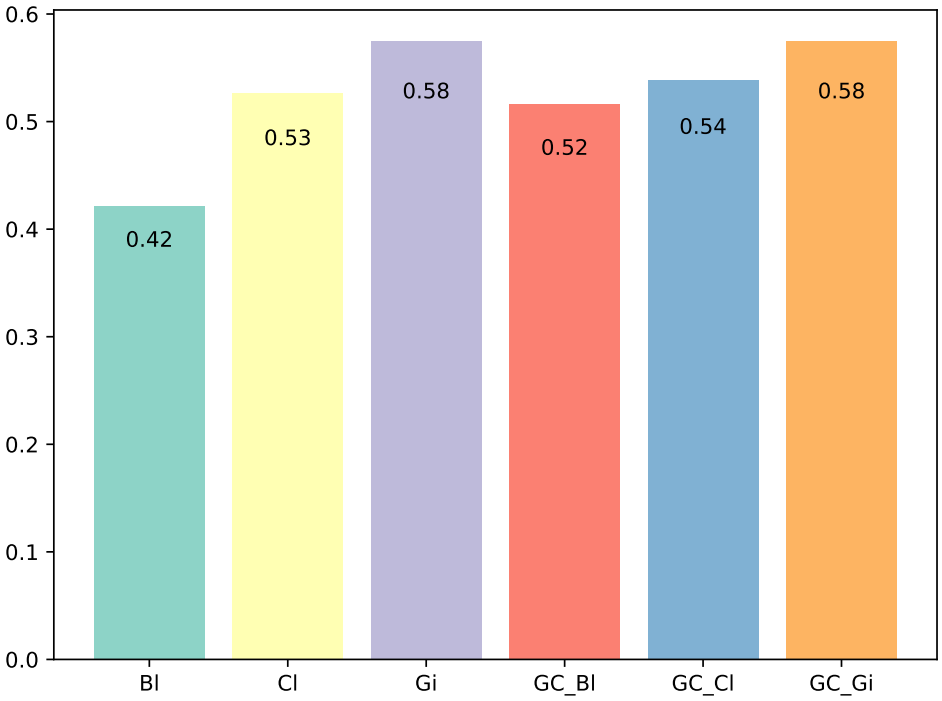


Figura 4.3: Medie dei valori di V-measure aggregate per algoritmo di generazione cluster

Capitolo 5

Conclusioni

Partendo da una lista di nomi di un dipartimento si può estrarre un grafo che lo rappresenti? Le comunità generate rispecchiano quelle reali?

TODO se vuoi fallo su matematica e mostra un grafo con qualche valore di v-measure

Bibliografia

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

[2] 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.

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

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

[5] 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`.