

UNIVERSITÀ POLITECNICA DELLE MARCHE

FACOLTÀ DI INGEGNERIA

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE



Analisi della rete di personaggi della serie TV "Game Of Thrones" tramite la libreria NetworkX di Python

Docenti:

Domenico URSINO
Luca VIRGILI

Studenti:

Lorenzo FRATINI
Federico MISCIA
Andrea PINCIAROLI

Anno Accademico 2021/2022

Indice

1	Introduzione	2
1.1	La Social Network Analysis	2
1.2	Game of Thrones Social Network	3
1.3	Strumenti utilizzati	4
1.3.1	NetworkX	4
1.3.2	Colab	5
2	Analisi descrittiva	6
2.1	Metriche Generali	6
2.2	Distribuzione dei pesi	8
2.3	Visualizzazioni e Layout	8
3	Analisi delle centralità	11
3.1	Degree Centrality	11
3.2	Closeness Centrality	14
3.3	Eigenvector Centrality	16
3.4	Betweenness Centrality	18
4	Analisi delle strutture	20
4.1	Triadi	20
4.2	Clique	21
4.3	K-core	22
4.4	Ego Network: Tyrion-Lennister	23
5	Group Centrality	25

Capitolo 1

Introduzione

Lo scopo di questa relazione è quello di ripercorrere le diverse fasi di lavoro svolte nell'ambito dell'analisi di una rete sociale basata sui personaggi di una serie TV, ponendo l'attenzione sugli strumenti utilizzati, le scelte effettuate e i risultati ottenuti, nel tentativo di derivare conoscenza a partire da un insieme di dati a primo impatto caotico, come quello di una rete.

1.1 La Social Network Analysis

La Social Network Analysis (abbr. SNA) costituisce un ramo molto importante della Data Analytics, con l'obiettivo di studiare il comportamento degli individui, a livello microscopico, la struttura delle relazioni tra essi, a livello macroscopico, e l'interazione che sussiste tra questi due aspetti. Oltre a ciò, la SNA permette di capire la dinamica dei flussi di informazione, aspetto critico per migliorare la comunicazione e mobilitare la conoscenza. La SNA affonda le sue radici diversi anni prima dell'avvento dei social network, come li intendiamo oggi; contrariamente a quanto si potrebbe pensare, infatti, si tratta di una metodologia che non è nata nell'ambito prettamente informatico bensì nel campo della sociologia, con l'intento di studiare i rapporti sociali che si creano tra le persone. Difatti, come sostiene Stokman (*International Encyclopedia of the Social Behavioral Sciences*), "le reti sociali sono, allo stesso tempo, la causa e il risultato del comportamento dei singoli".

Ad ogni modo, ora che le reti hanno milioni di utenti e i social network sono diventati un'enorme miniera di informazioni, non basta più la statistica per approcciarsi alla SNA per cui entrano in gioco le potenzialità della Data Science.

La SNA è fondata sulla teoria dei grafi: gli elementi da studiare (potrebbero essere persone, dispositivi o altro) sono rappresentati sotto forma di nodi, le relazioni tra essi sono rappresentate da archi. Mappando la rete su un grafo, come si vedrà nel seguito, è possibile esaminare importanti proprietà, quali ad esempio il potere o l'influenza dei componenti, la coesione tra i nodi e la presenza di particolari pattern tra essi.

Nel paragrafo successivo viene presentata la social network oggetto di studio, utilizzata nel progetto.

1.2. GAME OF THRONES SOCIAL NETWORK

1.2 Game of Thrones Social Network

Il dataset di riferimento per le attività di SNA svolte contiene la rete di relazioni che interessano i personaggi del mondo di "Game of Thrones", ed è disponibile al seguente link: <https://github.com/mathbeveridge/asoiaf/blob/master/data/asoiaf-all-edges.csv>.

Game Of Thrones (nota in Italia come "Il trono di spade") è una famosa serie TV statunitense di genere fantastico, trasmessa dall'emittente HBO. Si tratta di una vera e propria saga articolata in 8 stagioni, la quale racconta le avventure di molti personaggi all'interno di un mondo immaginario, impegnati nella lotta per la conquista del Trono di Spade dei Sette Regni.

L'appetibilità, nel contesto della SNA, di un dataset basato su questo mondo, deriva dal fatto che nell'arco della storia compaiono numerosi personaggi, appartenenti a diverse famiglie, portati a scontrarsi o allearsi vicendevolmente in un contorto gioco di potere. Si cercherà, dunque, di rintracciare i rapporti più forti tra i personaggi ed alcune delle molteplici strade che hanno portato i protagonisti a giocare un ruolo importante.

Nella Figura 1.1 è riportato un estratto del file csv relativo al dataset, così da avere un'idea di come esso sia strutturato.

Source	Target	Type	id	weight
Addam-Marbrand	Brynden-Tully	Undirected	0	3
Addam-Marbrand	Cersei-Lannister	Undirected	1	3
Addam-Marbrand	Gyles-Rosby	Undirected	2	3
Addam-Marbrand	Jaime-Lannister	Undirected	3	14
Addam-Marbrand	Jalabhar-Xho	Undirected	4	3

Figura 1.1: Prime cinque righe presenti nel file csv *asoiaf-all-edges.csv*

Di seguito una breve spiegazione degli attributi del dataset:

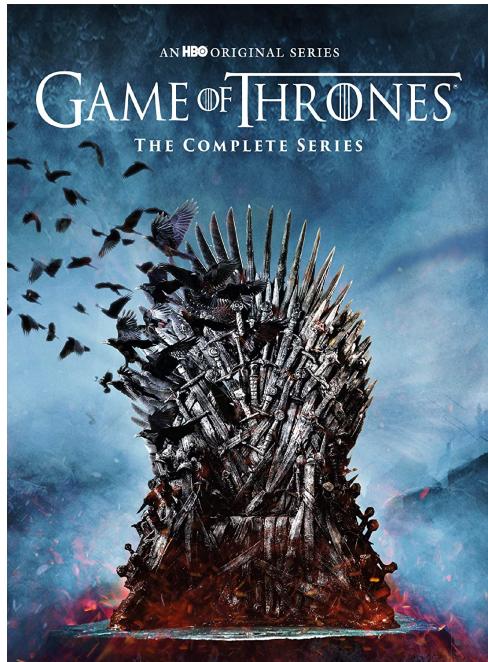
Game Of Thrones Social Network Dataset	
Attributo	Descrizione
Source	Nodo di partenza dell'arco
Target	Nodo di arrivo dell'arco
Type	Tipologia dell'arco: diretto o indiretto
id	Indice di riga (non utilizzato)
weight	Peso dell'arco

Ogni riga del dataset, dunque, rappresenta un arco del grafo, cioè una relazione che coinvolge due personaggi, i cui nomi completi compaiono, rispettivamente, negli attributi *Source* e *Target*.

Il grafo in questione è indiretto, ossia costituito esclusivamente da archi simmetrici, privi di direzionalità. Il peso dell'arco, come si legge dalla documentazione del dataset, è calcolato sulla base del numero di interazioni tra i personaggi coinvolti. Più precisamente, la rete è stata creata collegando due personaggi ogni volta che i loro nomi (o soprannomi) apparivano entro 15 parole l'uno dall'altro.

In conclusione, considerati la natura e il formato in cui appariva il dataset scelto, non si è resa necessaria una particolare attività di ETL.

1.3. STRUMENTI UTILIZZATI



1.3 Strumenti utilizzati

1.3.1 NetworkX

Al fine di poter disporre, all'interno dell'ambiente Python, di strumenti ad hoc per lo studio di una social network, si è fatto uso della libreria **NetworkX**.

Una volta aver importato il pacchetto *networkx*, infatti, si hanno a disposizione classi e metodi per gestire e manipolare grafi (diretti, indiretti, multigrafi...), convertire grafi tra vari formati, visualizzare grafi in 2D o 3D, esplorare proprietà quali grado, raggio, diametro etc... ed infine calcolare sottografi e strutture particolari. Gran parte di queste operazioni sono state svolte e, di conseguenza, saranno illustrate nel seguito.

NetworkX, inoltre, ha una notevole flessibilità di rappresentazione, infatti permette di associare a nodi ed archi qualsiasi tipo di dato, come ad esempio testo, immagini, numeri, record XML ...

In sostanza, si tratta di un framework ragionevolmente efficiente, ampiamente scalabile e portatile, motivi per cui si presta bene ad operazioni su grafi di grandi dimensioni ispirati al mondo reale.



1.3. STRUMENTI UTILIZZATI

1.3.2 Colab

Una peculiarità nell'approccio metodologico che ha accompagnato questo progetto riguarda l'uso dello strumento *Colab*.

Colaboratory, abbreviato in “Colab”, è un prodotto offerto da Google Research che permette a chiunque di scrivere ed eseguire codice Python sfruttando il browser web.

In particolare, *Colab* è un servizio che si basa su *Jupyter Notebook* e, come quest'ultimo, fornisce accesso gratuito a risorse computazionali per l'esecuzione del codice.

La particolarità che ha spinto all'utilizzo di questo tool è la possibilità di condividere Jupyter notebooks con altri collaboratori, senza la necessità di scaricare, installare o configurare nulla.

A differenza dei precedenti progetti, infatti, non è risultata agevole una ripartizione del lavoro per tematiche o per tipologia di analisi, che avrebbe reso possibile operare anche autonomamente in locale, bensì è stato necessario procedere in maniera simultanea sullo stesso blocco di codice.

Tramite *Colab*, dunque, i notebook vengono salvati su Google Drive e, di conseguenza, possono essere modificati da più utenti in parallelo.



Capitolo 2

Analisi descrittiva

In questo capitolo si provvede a fornire un resoconto, a livello generale, delle caratteristiche della rete in questione, attraverso sue visualizzazioni e l'esposizione di quelle che sono le prime metriche "conoscitive".

2.1 Metriche Generali

In primo luogo, sono state esplorate le proprietà strutturali tipiche di una rete, con l'obiettivo di ottenere un'idea di massima della social network che ci si apprestava ad analizzare. Le metriche prese in considerazione a tal proposito sono state le seguenti:

- **Numero di nodi**
- **Numero di archi**
- **Densità:** attributo i cui possibili valori sono compresi tra 0 e 1; misura la probabilità che una qualsiasi coppia di nodi sia adiacente. Si tratta, dunque, di un indice che misura mediamente quanto sono legati, tra loro, i nodi del grafo.
- **Densità pesata:** attributo analogo al precedente in cui, però, entrano in gioco i pesi degli archi.
- **Raggio:** corrisponde alla minima eccentricità dei nodi, dove l'eccentricità è definita come la lunghezza massima dello *shortest path* tra un nodo e i restanti nodi della rete.
- **Diametro:** corrisponde all'eccentricità massima tra i nodi.
- **Periferia:** insieme di nodi tali per cui l'eccentricità è pari al diametro.
- **Average Clustering Coefficient:** il *Clustering Coefficient* di un nodo fornisce una misura di quanto i suoi vicini tendano a formare una *clique*. Facendo una media riferita all'intero grafo, si ottiene una proporzione di quanto i nodi tendono ad essere connessi tra loro.
- **Connessione:** attributo booleano che restituirà "True" se il grafo è connesso, "False" se esso presenta componenti connesse.

2.1. METRICHE GENERALI

Quanto al *Raggio* e al *Diametro*, va specificato che la lunghezza di un percorso, avendo a che fare con un grafo pesato, è data dalla somma dei pesi degli archi che lo compongono e, nel caso specifico della social network in questione, la somma dei pesi corrisponde al numero di interazioni tra i personaggi.

Per il calcolo delle proprietà sono stati utilizzati opportuni metodi, dal nome spesso evocativo, della libreria *NetworkX*.

L'output ottenuto è mostrato nella figura 2.1.

```
Numero di Nodi: 796
Numero di Archi: 2823
Densità: 0.009
Densità Pesata: 0.103
Raggio: 5
Diametro: 9
Periferia: ['Lucifer-Long', 'Hugh-Hungerford', 'Murch', 'Gerold-Grafton', 'Gormon-Tyrell', 'Torwold-Browntooth', 'Manfrey-Martell', 'Simon-Toyne']
Average Clustering Coeff: 0.486
Is connected: True
```

Figura 2.1: Risultati delle metriche generali relative alla rete

Procedendo ad un commento di alcuni dei valori risultanti, si nota una densità decisamente bassa, il che può lasciare presagire una disparità nella distribuzione dei gradi tra i vari nodi. Oltre a ciò, il basso rapporto tra numero di archi del grafo e il numero di archi del medesimo grafo completo ($n(n-1)/2 = 316410$) lascia intuire il perché di una densità così esigua.

Va leggermente meglio considerando anche i pesi degli archi, in tal caso, infatti, la densità pesata è pari al 10,3%.

Proseguendo, nella lista relativa alla *Periferia* sono presenti effettivamente personaggi minori, come ad esempio mercenari, spesso non presenti nella serie TV bensì esclusivamente nei libri, da cui la serie è tratta.

Infine, l'*Average Clustering Coefficient* pari a 0.486 indica che, mediamente, quasi la metà delle figure che sono in relazione con un certo personaggio sono anche in relazione tra loro. Tale dato potrà avere ripercussioni nell'analisi delle strutture, nel Capitolo 4.

2.2. DISTRIBUZIONE DEI PESI

2.2 Distribuzione dei pesi

Successivamente, è stata condotta un'analisi grafica relativa alla distribuzione dei pesi presenti sugli archi del grafo, ossia del numero di interazioni tra i personaggi.

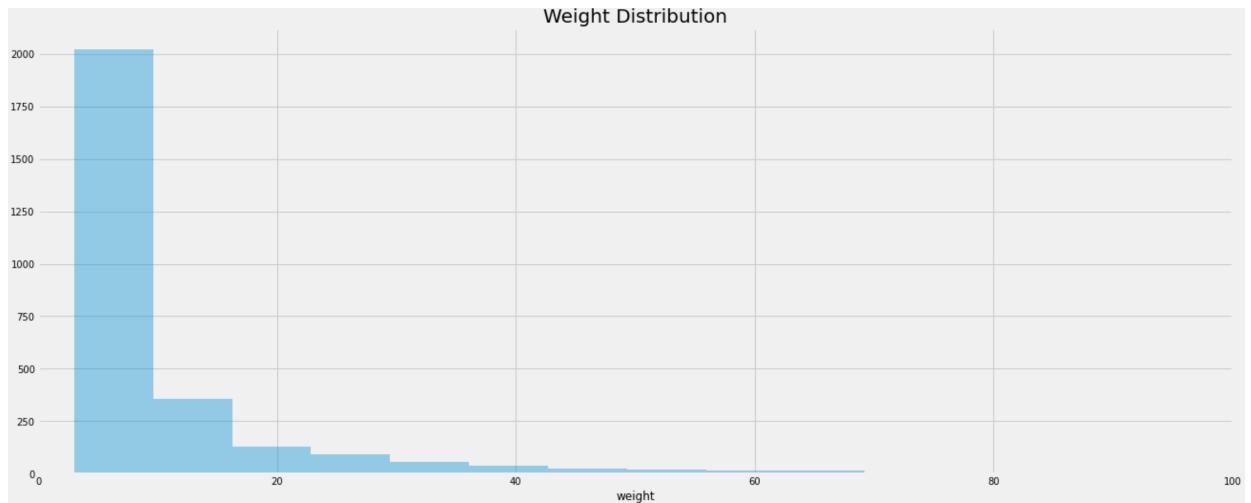


Figura 2.2: Distribuzione dei pesi degli archi

Dal grafico riportato in figura 2.2, si osserva come la quasi totalità dei pesi nel grafo sia inferiore a 10; ad ogni modo, quella che appare è una distribuzione con un andamento esponenziale in cui si registra molte volte un numero ristretto di interazioni e poche volte un numero di interazioni elevato (da 20 in su).

2.3 Visualizzazioni e Layout

In conclusione dell'analisi descrittiva si è provveduto a rappresentare graficamente la rete, esplorando le diverse possibilità di layout offerte da *NetworkX*.

Inizialmente, il grafo è stato riorganizzato secondo lo Spring Layout per poi essere visualizzato per mezzo della libreria *matplotlib*. Il risultato è mostrato in figura 2.3.

Come si può notare, la gran parte dei nodi è concentrata nella regione centrale della rete, a formare un nucleo abbastanza popolato e caotico. Osservando bene, in alto a destra rispetto questo nucleo centrale, è possibile scorgere una sorta di appendice, un insieme di nodi più denso del resto della frontiera e che, di conseguenza, potrebbe rappresentare una comunità relativamente isolata dal centro della rete. A parte questa particolarità, comunque, non emergono, a vista d'occhio, pattern significativi o indizi di gruppi ben distinti. La questione delle comunità e delle strutture, ad ogni modo, verrà ripresa ed analizzata nei capitoli successivi.

La seconda visualizzazione adottata per il grafo si basa sul cosiddetto Spiral Layout ed è mostrata nella figura 2.4. Tale disposizione, dal punto di vista della significatività, viene sicuramente a perdere valore ma è interessante sotto l'aspetto prettamente grafico.

2.3. VISUALIZZAZIONI E LAYOUT

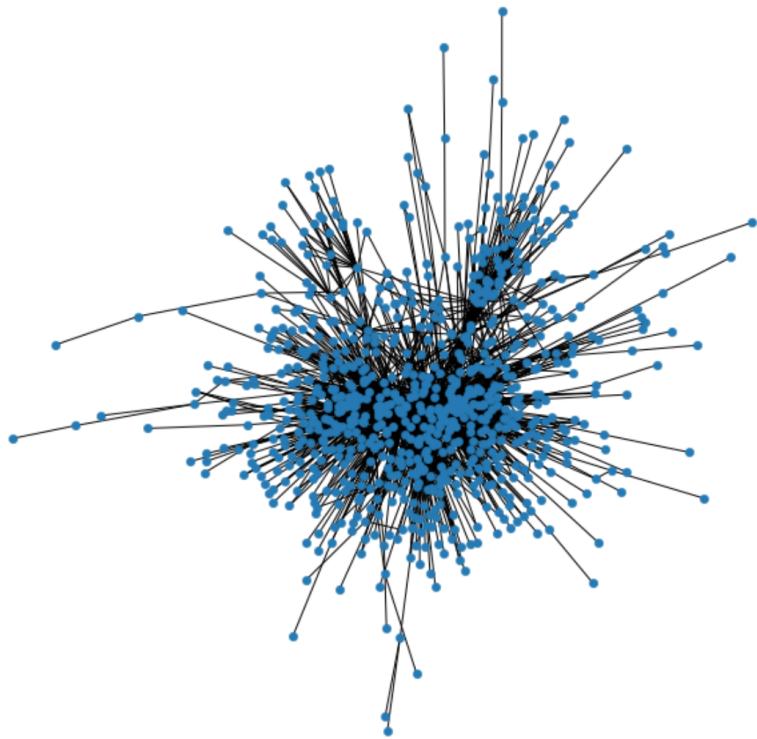


Figura 2.3: Spring Layout

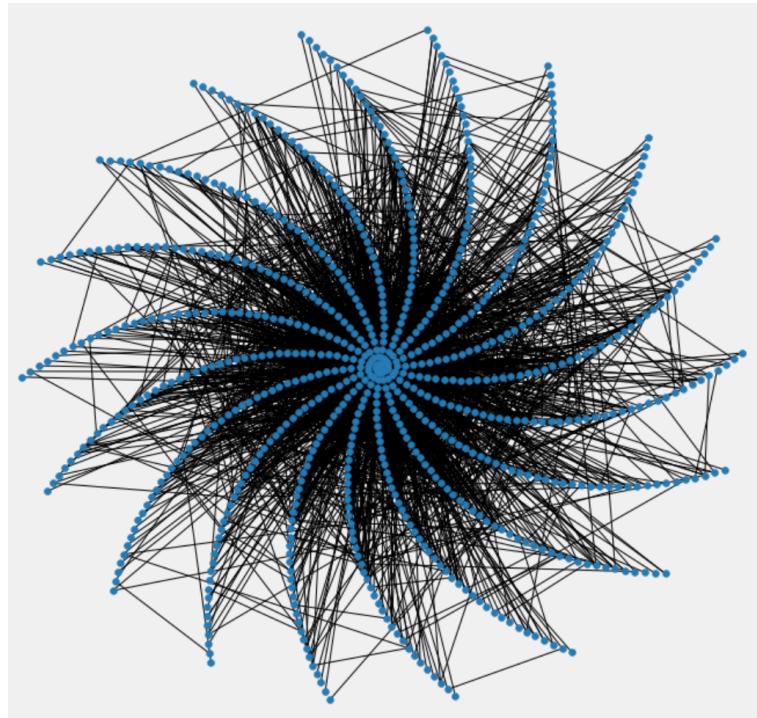


Figura 2.4: Spiral Layout

2.3. VISUALIZZAZIONI E LAYOUT

Infine, è stato utilizzato il Kamada Kaway Layout, il quale dispone i nodi secondo l'omonima funzione-costo che calcola la lunghezza dei percorsi.

Tale disposizione, mostrata in figura 2.5, è tutto sommato simile allo Spring Layout; di contro, ha richiesto un tempo di elaborazione decisamente maggiore.

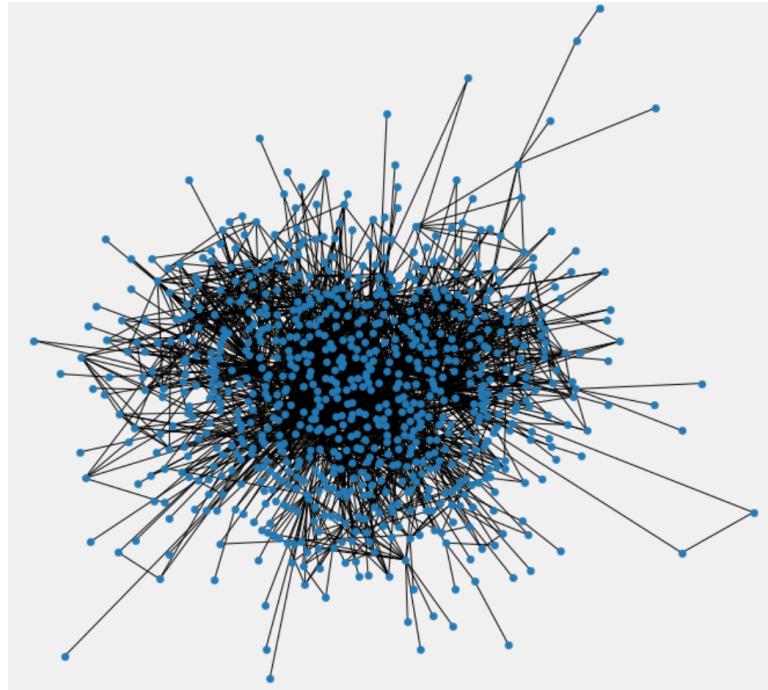


Figura 2.5: Kamada Kaway Layout

Capitolo 3

Analisi delle centralità

Una delle prime analisi che si effettuano su una social network consiste nell'andare a misurare il potere o l'influenza dei componenti di una rete, sulla base delle loro connessioni. Non esistono metriche univoche, però, per stabilire l'importanza di un nodo ed ecco perchè nel tempo sono state sviluppate diverse misure di centralità. Di seguito vedremo le più studiate e analizzate per la social network analysis.

3.1 Degree Centrality

La prima misura di centralità che è stata calcolata è la **Degree Centrality** con la quale si calcolano il numero di legami che un certo nodo ha con gli altri della rete. Tale misura evidenzia come un nodo risulti essere più centrale tanto più è alto il suo grado. Questa misura però non garantisce che il nodo con la degree centrality più alta sia il più importante perchè si potrebbe ricadere nel caso in cui si hanno tanti legami ma di poca importanza. Tale misura viene calcolata grazie alla funzione `degree_centrality()` messa a disposizione dalla libreria NetworkX. In figura 3.1 sono riportati i dieci personaggi della serie GoT che hanno ottenuto lo score più alto (valore normalizzato tra 0 e 1).

Best 10 Degree Centrality:

1. Tyrion-Lannister: 0.15345911949685534
2. Jon-Snow: 0.14339622641509434
3. Jaime-Lannister: 0.1270440251572327
4. Cersei-Lannister: 0.1220125786163522
5. Stannis-Baratheon: 0.11194968553459118
6. Arya-Stark: 0.10566037735849056
7. Catelyn-Stark: 0.09433962264150943
8. Sansa-Stark: 0.09433962264150943
9. Robb-Stark: 0.0930817610062893
10. Eddard-Stark: 0.0930817610062893

Figura 3.1: Personaggi con degree centrality più alta

I risultati ottenuti sono coerenti con le aspettative in quanto i primi dieci sono essenzialmente i personaggi principali della serie. Al primo posto si trova Tyrion Lannister seguito dal protagonista Jon Snow. Seguono i fratelli di Tyrion cioè Jaime e Cersei ed infine Stannis Baratheon e tutta la casata Stark.

3.1. DEGREE CENTRALITY

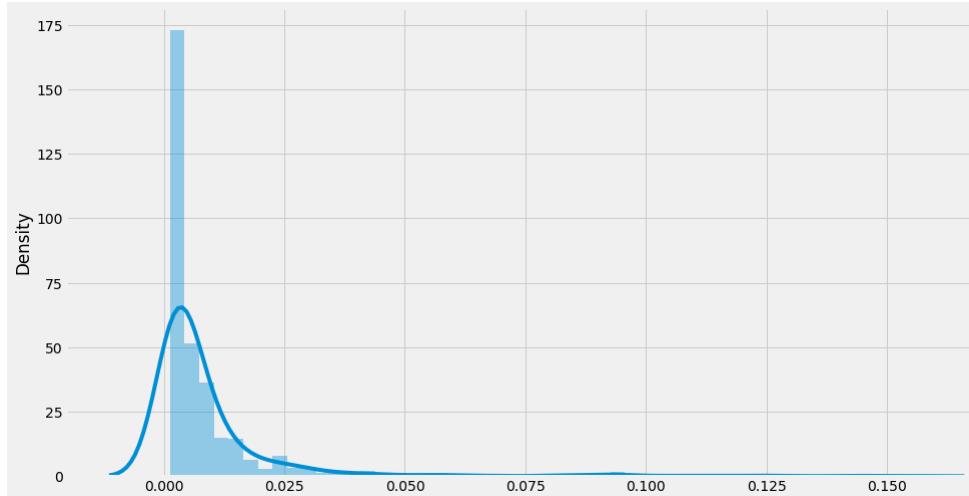


Figura 3.2: Densità dei nodi - Degree Centrality

Siccome il numero dei nodi della rete è elevato, riportare un bar plot con tutti i valori delle degree centrality avrebbe restituito un risultato illeggibile, quindi si è deciso di mostrare la distribuzione di tali valori, come si può notare in figura 3.2. Come si può osservare, la distribuzione segue un andamento esponenziale, come spesso accade per i fenomeni legati alle social network. Di seguito si descrivono in breve i personaggi con la degree centrality più alta in modo da capire il motivo della loro importanza.

Tyrion Lannister



Figura 3.3: *"Questo è quello che faccio, bevo e sò cose"*

3.1. DEGREE CENTRALITY

Tyrion Lannister è terzogenito di lord Tywin Lannister e fratello della regina Cersei Lannister e ser Jaime Lannister. Per colpa del suo aspetto viene deriso dalla quasi totalità della sua famiglia, tranne da suo fratello che lo rispetta. Proprio perchè si sente un emarginato nel corso della serie simpatizzerà per personaggi in condizioni simili alla sua, come Jon Snow. Non stupisce il fatto che sia il personaggio con degree centrality più alta proprio perchè durante la storia è vittima di una serie di peripezie che lo portano a relazionarsi con la maggior parte delle casate più importanti che lottano per il trono. La sua grande intelligenza e sottigliezza gli permettono, inoltre, di essere una persona piacente e quindi abile nel relazionarsi.

Jon Snow



Figura 3.4: "*L'inverno sta arrivando*"

Jon Snow è il figlio bastardo di Eddard Stark, re di Grande Inverno. Cresce insieme ai figli di Eddard ovvero Robb, Sansa, Arya, Bran e Rickon ma la verità sui suoi genitori viene nascosta a tutti, lui stesso compreso. Jon entra nei Guardiani della notte e ciò provocherà l'allontanamento da Grande Inverno. Più tardi nella storia scoprirà di essere figlio di Rhaegar Targaryan e ciò lo rende legittimo erede al trono di spade. Combatterà al fianco di Daenerys Targaryan per riuscire ad ottenere il trono. Il suo alto grado di degree centrality è dato dal fatto che è il protagonista indiscusso della serie.

Cersei Lannister

Cersei, figlia di Tywin Lannister e sorella di Tyrion, è uno dei personaggi chiave più discusi della serie. Le sue vicende incestuose con il fratello Jaime Lannister hanno portato ad una delle due guerre più importanti della storia che l'hanno vista diventare la prima donna a essere regina dei sette regni. Tutti questi eventi hanno portato Cersei ad avere un gran numero di relazioni con i personaggi della serie.

3.2. CLOSENESS CENTRALITY



Figura 3.5: "Un lannister paga sempre i suoi debiti"

Infine, in figura 3.6 viene riportato la heatmap delle degree centrality in cui i nodi centrali, essendo interessati da un maggior numero di connessioni, hanno un valore più alto.

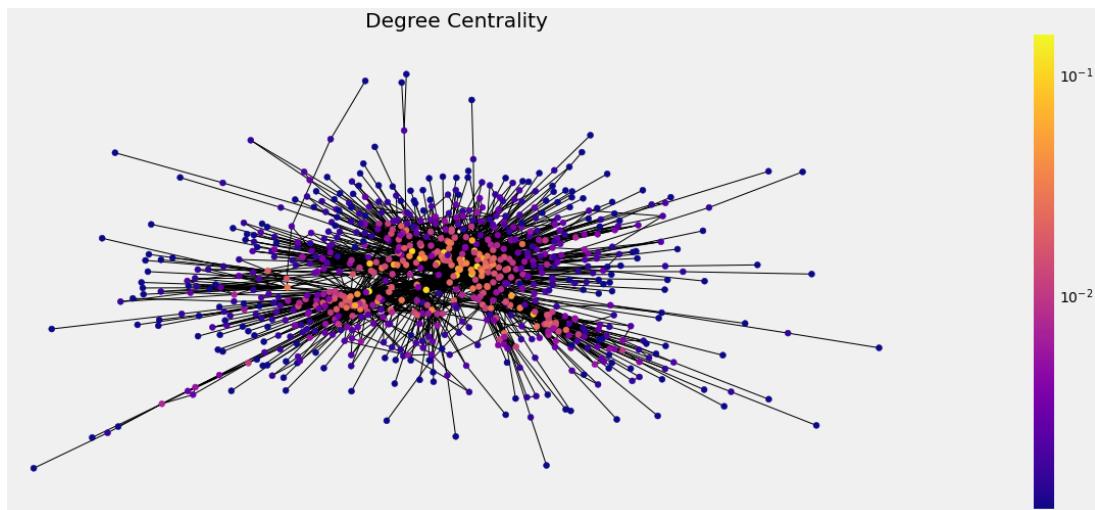


Figura 3.6: Spring Layout - Degree Centrality

3.2 Closeness Centrality

La **Closeness Centrality** è una misura che calcola la distanza di un nodo dagli altri nodi della rete. Un nodo con un alto valore di questa misura lo indica come un ottimo candidato per ricevere ed inviare informazioni agli altri componenti della rete. Anche in questo caso vengono riportati i dieci personaggi con lo score più significativo, come si può notare dalla figura 3.7, calcolato con la funzione di NetworkX `closeness_centrality()`.

3.2. CLOSENESS CENTRALITY

Best 10 Closeness Centrality (Distance):

1. Jaime-Lannister: 0.09587554269175108
2. Robert-Baratheon: 0.09244186046511628
3. Stannis-Baratheon: 0.0911801812134419
4. Theon-Greyjoy: 0.09111747851002866
5. Jory-Cassel: 0.09075342465753425
6. Tywin-Lannister: 0.09044368600682594
7. Tyrion-Lannister: 0.08984065996157758
8. Cersei-Lannister: 0.08981021238138275
9. Brienne-of-Tarth: 0.08939615427864613
10. Jon-Snow: 0.08895602551191675

Figura 3.7: Personaggi con closeness centrality più alta

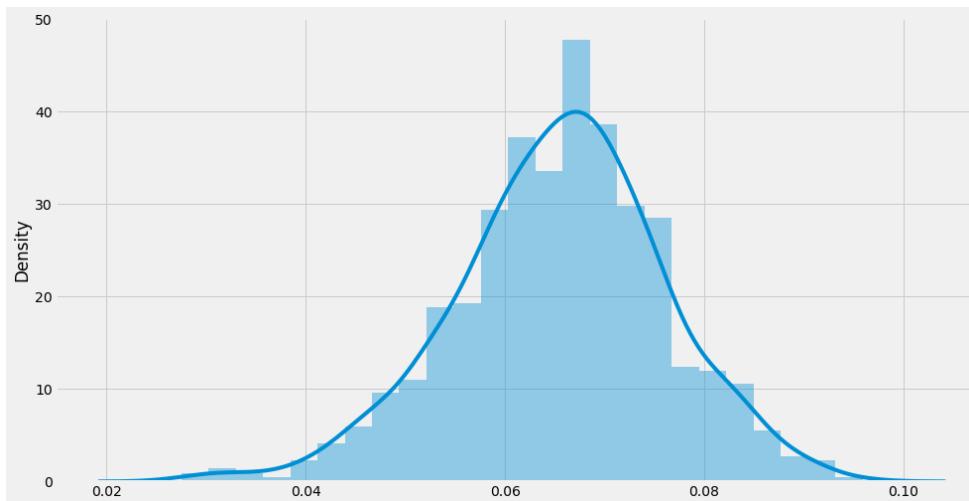


Figura 3.8: Densità dei nodi - Closeness Centrality

Come si può osservare dalla distribuzione dei valori della closeness centrality in figura 3.8, in generale essi sono molto bassi e simili tra di loro. Da questo si può dedurre che questo tipo di rete ha dei nodi che sono molto distanti tra di loro e quindi non è facile individuare un nodo capace di trasmettere informazione velocemente all'interno della rete. dal punto di vista della serie TV questo trova un riscontro visto che praticamente tutte le casate sono in guerra tra di loro e le alleanze spesso avvengono solo per secondi fini.

In Figura 3.8 si riporta il grafico dell'heatmap della closeness centrality che conferma come la quasi totalità dei nodi abbia lo stesso valore molto basso.

3.3. EIGENVECTOR CENTRALITY

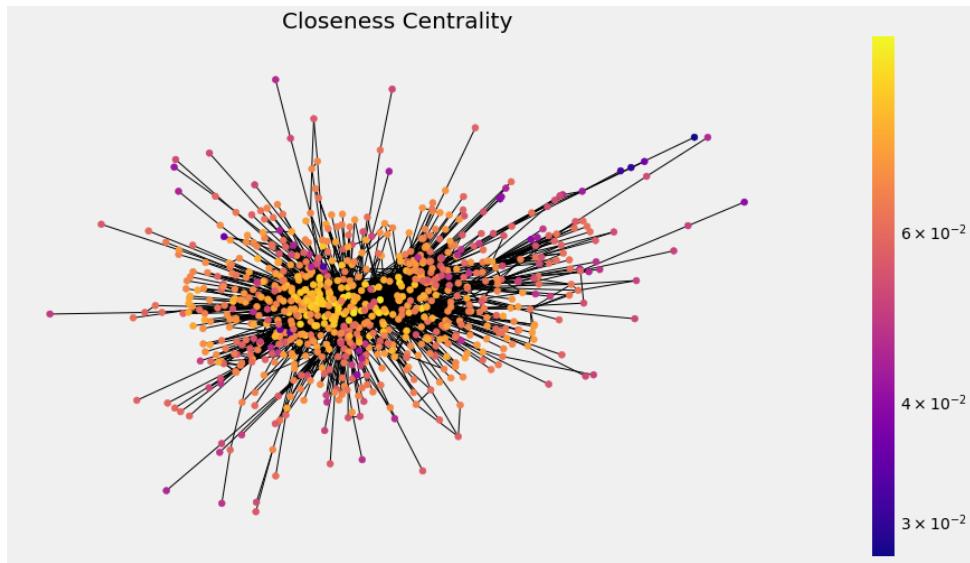


Figura 3.8: Spring Layout - Closeness Centrality

3.3 Eigenvector Centrality

La terza metrica che è stata calcolata è la **Eigenvector Centrality**. Questa centralità misura quanto un nodo è autorevole e influente all'interno della rete ovvero la sua importanza rispetto a quelli che gli sono vicini. I risultati ottenuti, mostrati in figura 3.9, attraverso l'utilizzo della funzione `eigenvector_centrality()` sono i seguenti.

Best 10 Eigenvector Centrality (Weight):

1. Tyrion-Lannister:0.38051325945323644
2. Cersei-Lannister:0.3587382421102516
3. Joffrey-Baratheon:0.34299188369358463
4. Robert-Baratheon:0.2826979007992425
5. Eddard-Stark:0.2817609476067616
6. Sansa-Stark:0.27429530921109546
7. Jaime-Lannister:0.21482369342058674
8. Robb-Stark:0.1739899049036914
9. Jon-Snow:0.16905984194643536
10. Catelyn-Stark:0.16406022784991475

Figura 3.9: Personaggi con eigenvector centrality più alta

3.3. EIGENVECTOR CENTRALITY

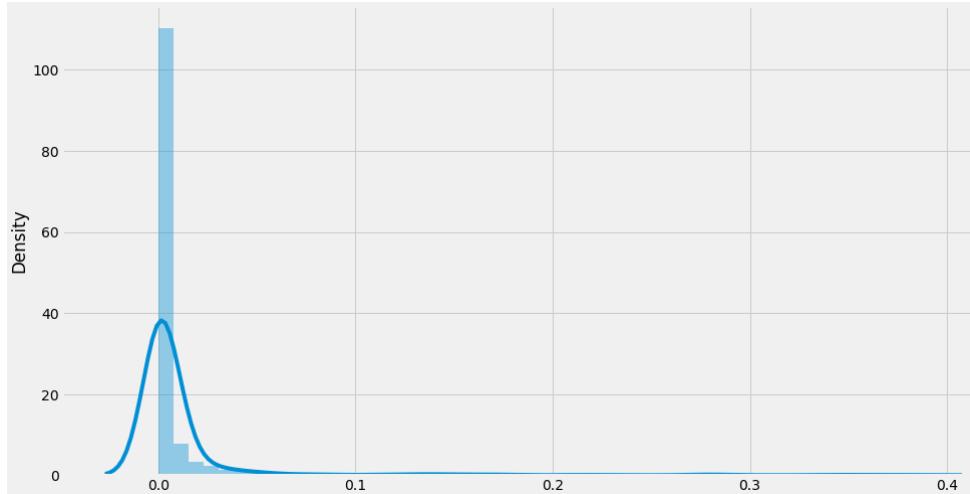


Figura 3.10: Densità dei nodi - Eigenvector Centrality

In questo caso i risultati sono stati ottenuti considerando anche l'attributo weight del dataset ovvero il peso dell'arco che indica l'importanza della relazione tra il nodo sorgente e quello di destinazione. Ai primi posti troviamo i personaggi che, come visto nella breve spiegazione riportata nella sezione 3.2, effettivamente sono autorevoli e influenti per via del ruolo che ricoprono (Cersei Lannister è stata regina, Joffrey Baratheon è stato re) sia per l'impatto che hanno nelle vicende della storia (Tyrion Lannister ha ricoperto ruoli importanti al servizio di diversi protagonisti della serie).

In figura 3.11 viene mostrato la heatmap della eigenvector centrality.

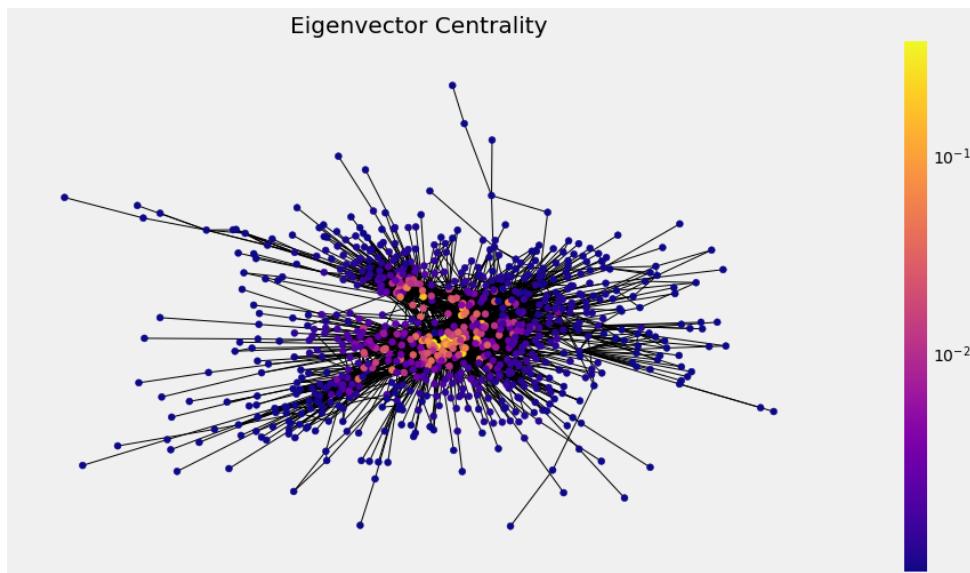


Figura 3.11: Spring layout - Eigenvector Centrality

3.4. BETWEENNESS CENTRALITY

3.4 Betweenness Centrality

Infine, ultima metrica analizzata, la **Betweenness Centrality** che misura la strategicità di un nodo della rete tra due aree importanti della stessa. Un nodo con betweenness centrality alta è un individuo che agisce da bridge tra due o più comunità della rete. I risultati, riportati in figura 3.12 e in figura 3.13, sono stati calcolati attraverso la funzione di NetworkX `betweenness_centrality()`. Come per la precedente metrica è stato utilizzato il parametro `weight` in modo tale che gli shortest path per ogni coppia di nodi vengano calcolati tenendo conto dei pesi degli archi.

Best 10 Betweenness Centrality (Weight):

1. Jon-Snow:0.1321196445389688
2. Theon-Greyjoy:0.12326572797391663
3. Jaime-Lannister:0.11677630973743316
4. Daenerys-Targaryen:0.09419228255405236
5. Stannis-Baratheon:0.09291440189591416
6. Robert-Baratheon:0.09252285861002142
7. Tyrion-Lannister:0.09162375232999247
8. Cersei-Lannister:0.07734001906118228
9. Tywin-Lannister:0.06358359223028201
10. Robb-Stark:0.06295787760760911

Figura 3.12: Personaggi con betweenness centrality più alta



Figura 3.13: Densità dei nodi - Betweenness Centrality

Dai risultati i valori di betweenness sono molto bassi e questo indica che in particolare nella rete non ci sono nodi che fungono da punto chiave di comunicazione tra diverse comunità.

Infine, viene riportata la rappresentazione 3.14 attraverso lo spring layout.

3.4. BETWEENNESS CENTRALITY

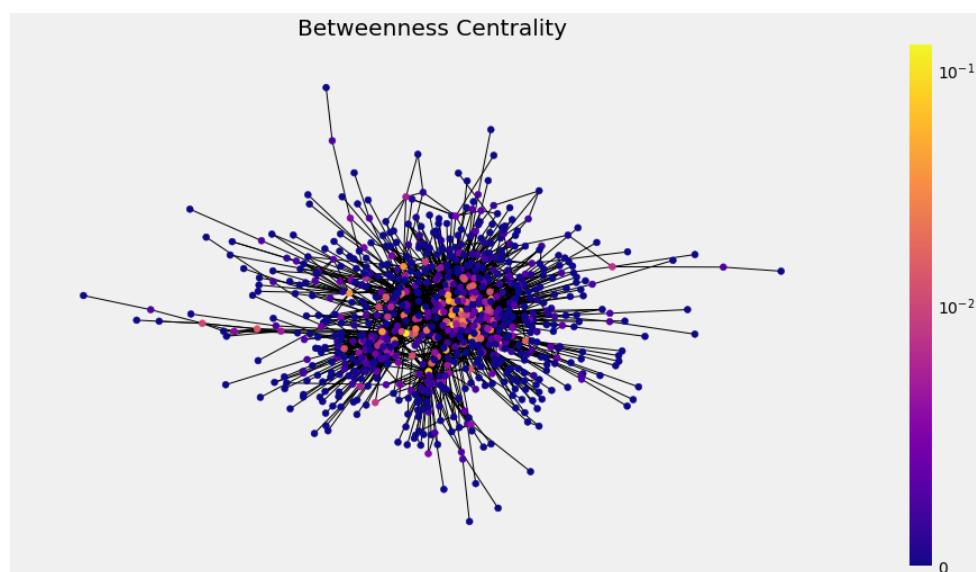


Figura 3.14: Spring layout - Betweenness Centrality

Capitolo 4

Analisi delle strutture

In questa fase di lavoro ci si è concentrati sullo studio e l'interpretazione dei sottografi della rete che danno luogo a strutture note come triadi e clique, sinonimi di coesione e robustezza. In secondo luogo, è stata applicato il concetto di ego network ad uno dei personaggi più influenti di Game of Thrones, così da avere un'idea del suo "vicinato" e del suo campo d'azione.

4.1 Triadi

Una **triade** è un insieme di tre nodi e si dice chiusa se ciascun nodo è connesso, tramite un arco, ai due rimanenti. La triade chiusa è sinonimo di stabilità per quel che riguarda la diffusione delle informazioni: se infatti, ad un certo punto, dovesse interrompersi una delle tre possibili connessioni, l'informazione potrebbe comunque giungere a tutti e tre i nodi.

Per calcolare il numero di triadi presenti nel grafo in questione è stato necessario effettuare un pre-filtraggio dello stesso; nello specifico, sono state considerate solo le triadi la cui somma dei pesi fosse maggiore o uguale di 100. Una volta aver imposto tali condizioni, il calcolo è stato eseguito attraverso un ciclo sui sottografi prodotti dal metodo ***combinations()***. I parametri passati a tale metodo sono stati i nodi del grafo filtrato e l'intero 3, cioè la dimensione di ciascuna combinazione da trovare; dopodiché si è verificato che i sottografi risultanti avessero un numero di archi pari a 3 e, dunque, fossero effettivamente una triade chiusa.

Le triadi trovate sono state 779. Tale dato, insieme ad un resoconto dei personaggi che formano le prime 10 triadi con peso maggiore, è riportato in figura 4.1. La triade più forte formata da Joffrey, Tyrion e Cersei è giustificata dal fatto che Cersei, nella serie, è la madre di Joffrey e Tyrion è il fratello di Cersei. Oltre a questo, per un certo periodo, Joffrey è stato proclamato re e Tyrion divenne primo cavaliere ovvero la figura più vicina al re in assoluto. Sulla base delle maggiori 10 triadi, inoltre, spiccano due personaggi più frequenti, Robert-Baratheon e Eddard-Stark, i quali compaiono 5 volte. Essi, dunque, sono candidati ad essere dei buoni diffusori di informazioni.

4.2. CLIQUE

```

Triads:779
Top 10 Triads (weight based):

1. 'Joffrey-Baratheon', 'Tyrion-Lannister', 'Cersei-Lannister'(601)
2. 'Joffrey-Baratheon', 'Tyrion-Lannister', 'Sansa-Stark'(559)
3. 'Robert-Baratheon', 'Eddard-Stark', 'Cersei-Lannister'(559)
4. 'Joffrey-Baratheon', 'Sansa-Stark', 'Cersei-Lannister'(490)
5. 'Jeor-Mormont', 'Samwell-Tarly', 'Jon-Snow'(453)
6. 'Robert-Baratheon', 'Eddard-Stark', 'Renly-Baratheon'(452)
7. 'Tyrion-Lannister', 'Jaime-Lannister', 'Cersei-Lannister'(452)
8. 'Robert-Baratheon', 'Catelyn-Stark', 'Eddard-Stark'(449)
9. 'Robert-Baratheon', 'Eddard-Stark', 'Varys'(442)
10. 'Joffrey-Baratheon', 'Robert-Baratheon', 'Eddard-Stark'(439)

```

Figura 4.1: Analisi delle triadi. In alto è riportato il numero totale di triadi; segue una lista con le top 10 triadi in cui, tra parentesi, è riportato il peso complessivo.

4.2 Clique

Estendendo il concetto di triade chiusa a più nodi, si parla di **clique**. Una clique, dunque, è un generico insieme di nodi totalmente connesso e, di fatto, può essere visto come l'insieme di più triadi tra loro sovrapposte.

Le cliques sono la massima espressione di coesione tra nodi ed, in quanto tali, la loro individuazione all'interno di una rete può risultare molto importante.

Utilizzando il metodo `find_cliques()` sul grafo relativo alla rete di rapporti tra i personaggi di Game of Thrones, sono risultate 681 cliques comprendenti almeno 4 nodi.

Dopodiché, è stata visualizzata la clique massima, la quale conta un numero di nodi pari a 14; il sottografo corrispondente è mostrato in figura 4.2. E' possibile notare che la clique in questione è ripartita essenzialmente tra gli appartenenti a tre famiglie: i Baratheon, i Lannister e gli Stark.

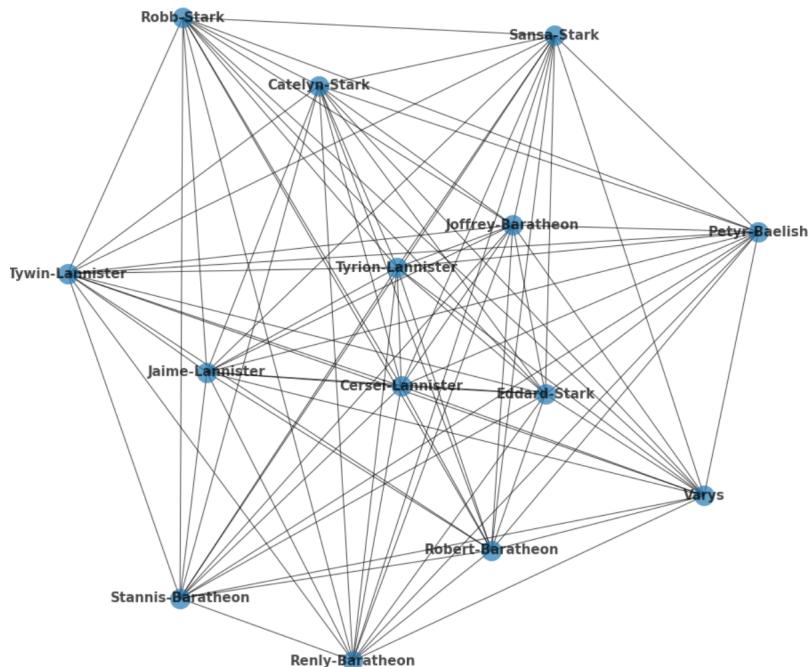


Figura 4.2: Clique costituita da 14 nodi

4.3. K-CORE

La clique, come ci si poteva aspettare, è ripartita essenzialmente tra le tre casate protagoniste della serie le quali, a causa delle continue sotto-trame e stratagemmi (matrimoni funzionali, intrecci amorosi, cospirazioni) sono inevitabilmente collegate in modo completo. Le tre casate in questione sono: Lennister, Baratheon e Stark. Oltre a ciò, spicca il personaggio Petyr Baelish il quale, però, non ha un legame di parentela con le tre famiglie sopra citate ma è un personaggio manipolatore e viscido che per tutta la serie cerca di sedersi sul tanto agognato trono dei Sette Regni, relazionandosi quindi forzatamente con gli effettivi pretendenti al trono attraverso strategie doppio-giochiste.

4.3 K-core

Un’ulteriore struttura analizzata è stata **k-core**. K-core costituisce un rilassamento del concetto di clique in quanto calcola il sottografo massimale tale per cui ogni suo nodo ha grado maggiore o uguale a k .

Tale sottografo è stato calcolato per mezzo del metodo `k_core()` il quale, non ricevendo l’ordine k all’atto della chiamata, calcola di default il "main core" del grafo di partenza. Il risultato è mostrato nella figura 4.3. I nodi appartenenti al k-core sono 20; di fatto, nonostante una leggibilità non eccelsa, si tratta della clique massima cui sono stati aggiunti 6 nodi ulteriori. Nel k-core, infatti, non c’è più il vincolo di connessione diretta a ciascun altro nodo dunque il percorso tra due nodi qualsiasi può comprendere più di un arco.

In questo caso, i nodi in più che vengono mostrati rispetto alla precedente clique rappresentano personaggi che nel corso della storia hanno sì contribuito ad alcuni passaggi fondamentali per la corsa al trono ma poi o sono morti o sono rimasti un po’ distaccati rispetto agli eventi principali.

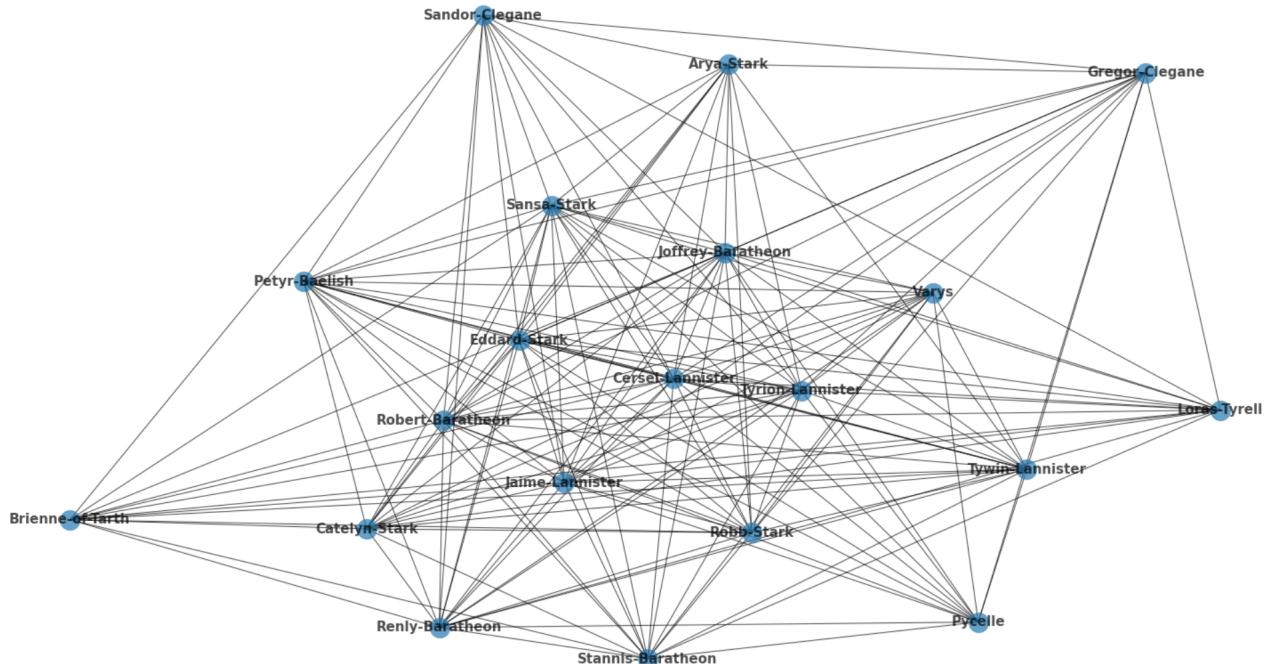


Figura 4.3: K-core

4.4. EGO NETWORK: TYRION-LENNISTER

4.4 Ego Network: Tyrion-Lennister

La successiva analisi effettuata ha visto come protagonista il personaggio Tyrion-Lennister, attorno al quale è stata costruita una **ego-network**.

Una ego-network è una sotto-rete centrata su un nodo specifico, particolarmente utile quando si intende studiare un elemento di cui non si conosce molto. Tale struttura è fortemente basata sul concetto di **omofilia**, cioè la propensione delle persone a circondarsi di propri simili. Vedendo i vicini di un nodo, infatti, si possono ricavare delle informazioni preziose riguardo, in questo caso, il centro della ego-network.

Per l'analisi di tale struttura è stato scelto il personaggio Tyrion-Lennister in quanto quest'ultimo gioca un ruolo importante nella storia ed è risultato al primo posto per quel che riguarda la Degree Centrality e l'Egeinvector Centrality, come discusso nel capitolo 3. L'ego-network di Tyrion-Lennister, ottenuta attraverso il metodo `ego_graph()` di NetworkX, è mostrata nella figura 4.4. Per non appesantire eccessivamente la visualizzazione, si è scelto di pre-filtrare il grafo prima del calcolo, lasciando solamente gli archi con peso pari almeno a 30.

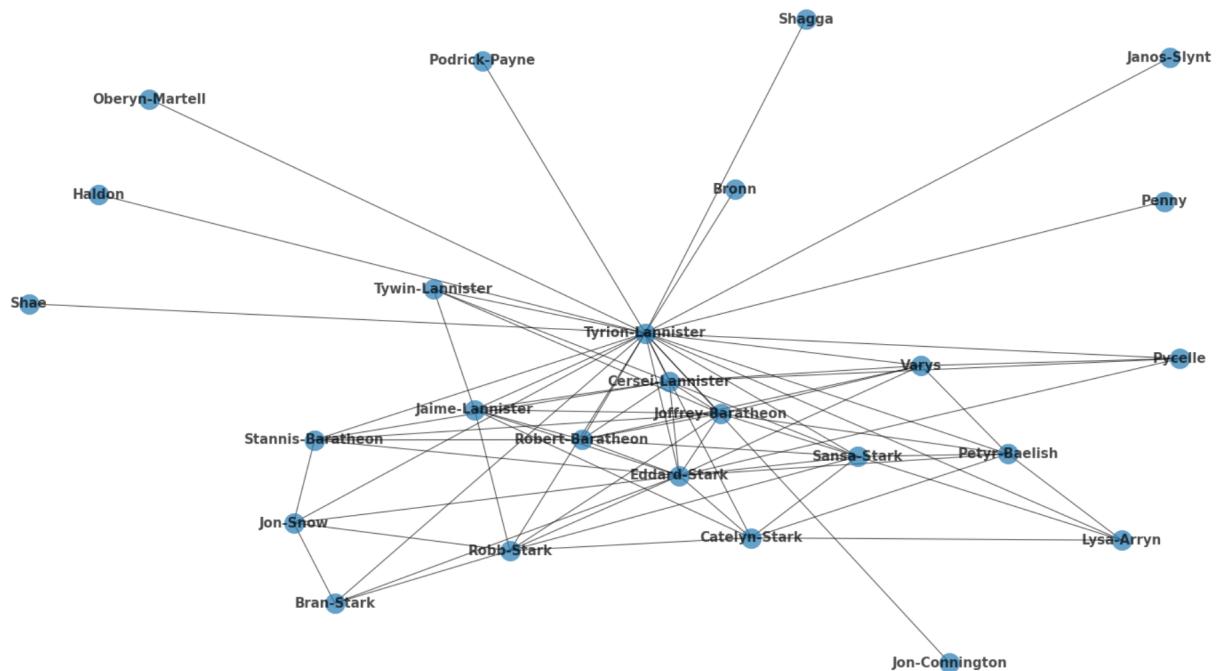


Figura 4.4: Ego Network relativa a Tyrion-Lennister

Osservando la ego-network, emerge una particolarità: è infatti possibile individuare due regioni con caratteristiche alquanto distinte, rispettivamente nella parte superiore ed inferiore.

Focalizzando l'attenzione sulla regione in alto, si nota come i vicini di Tyrion-Lennister non siano propensi ad essere connessi tra loro; per questo ulteriore sottografo il Clustering Coefficient tenderà a 0, dunque domina l'ego e le sembianze sono quelle di una rete a stella. Difatti, si notano esclusivamente delle triadi aperte che pongono Tyrion-Lennister a tutti gli effetti nel ruolo di un intermediatore.

Nella regione inferiore, invece, accade pressoché l'opposto, nel senso che i vicini del nodo

4.4. EGO NETWORK: TYRION-LENNISTER

"ego" presentano molte più connessioni reciproche, a formare una struttura più fitta nella quale il transito delle informazioni è sicuramente più efficiente.

Capitolo 5

Group Centrality

Come ultimo tipo di analisi sono stati calcolati i vari tipi di centralità per varie famiglie che compaiono all'interno della serie, per vedere quali fossero i gruppi più significativi. Le metriche di centralità di gruppo assumono lo stesso significato di quelle già viste nelle sezioni precedenti, tuttavia in questo caso gli algoritmi che li calcolano non considerano un singolo nodo, ma un insieme. Tali funzioni, appartenenti sempre alla libreria di *NetworkX*, sono:

1. **group_degree_centrality**: viene calcolata come il rapporto tra la somma del numero dei nodi non appartenenti al gruppo, ma connessi ad uno dei membri, e il numero totale di nodi che non appartengono al gruppo
2. **group_closeness_centrality**: viene calcolata come il rapporto fra il numero di nodi non appartenenti al gruppo e la somma delle distanze tra il gruppo e ogni nodo non appartenente al gruppo
3. **group_betweenness_centrality**: viene calcolata come il rapporto tra il numero di shortest path tra le coppie di nodi del grafo che passano per almeno uno dei nodi del gruppo e il totale di shortest path di tutte le possibili coppie del grafo

Per quanto riguarda le famiglie a cui si faranno riferimento per l'analisi in questione, per comprendere i vari partecipanti in Figura 5.1 è mostrato l'albero genealogico che definisce le varie relazioni tra i personaggi. In realtà, per facilitarne la lettura, in esso non vengono riportati tutti i membri della famiglia, ma solo i più importanti. Quindi, quando si sono considerati i vari gruppi per estrarre le misure di centralità sono state riportate delle persone che non compaiono in Figura 5.1.

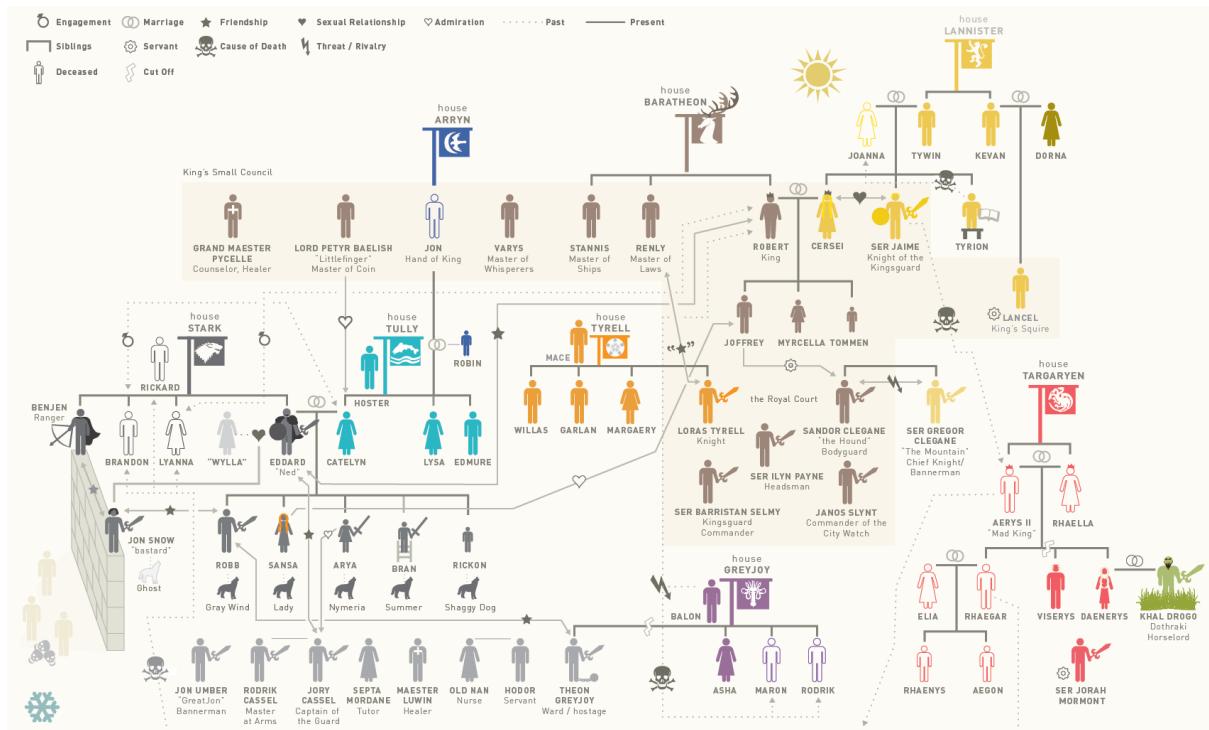


Figura 5.1: Albero genealogico GoT

Come si può osservare dalla figura i gruppi considerati sono stati otto, ovvero:

- **Lannister:** Twyin, Kevan, Tyrion, Lancel, Jaime, Cersei, Joanna, Daven, Genna e Martin
 - **Stark:** Brandon, Lyanna, Eddard, Jon Snow, Benjen, Robb, Sansa, Arya, Bran, Rickon e Catelyn
 - **Baratheon:** Stannis, Renly, Robert, Joffrey, Myrcella, Tommen e Shireen
 - **Tyrell:** Alla, Elinor, Garlan, Gormon, Leo, Loras, Mace, Margaery e Megga
 - **Tully:** Brynden, Edmure e Hoster
 - **Greyjoy:** Aeron, Asha, Balon, Euron, Theon, Quellon e Victarion
 - **Arryn:** Alys, Jon, Lysa, Robert
 - **Targaryen:** Aegon I, Aegon (figlio di Rhaegar), Aegon V, Aemon (Dragonknight), Aemon (Maester Aemon), Aeris I, Aeris II, Baelor I, Daenerys, Daeron I, Jaehaerys I, Maegor I, Rhaegar, Rhaella e Rhaenys

Sulla base delle famiglie appena riportate sono stati calcolate le metriche di centralità, le quali poi sono state riportate in tre diversi diagrammi a barre in modo da poter visualizzare meglio i risultati ottenuti.

Per quanto riguarda la **degree centrality**, i risultati ottenuti sono mostrati in Figura 5.2.

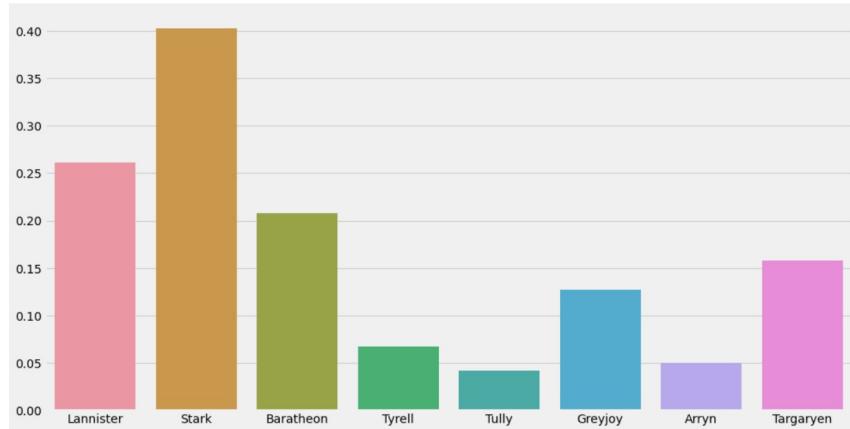


Figura 5.2: Group Degree Centrality

La famiglia Stark presenta un valore molto più alto di degree centrality rispetto le altre famiglie e questo probabilmente è dovuto al fatto che i personaggi appartenenti a questa famiglia ricoprono ruoli chiave nella storia (es. Jon Snow è il protagonista) e la maggior parte sono presenti dalla prima fino all'ultima stagione. Per quanto riguarda la **closeness centrality**, i risultati ottenuti sono mostrati in Figura 5.3.

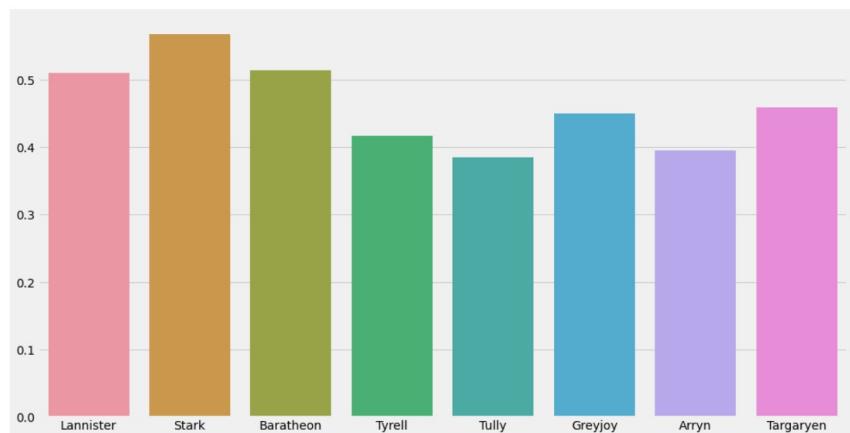


Figura 5.3: Group Closeness Centrality

In merito alla closeness centrality i risultati tra le famiglie non sono così distanti ma comunque primeggiano anche qua le casate Stark, Lannister e Baratheon.

Per quanto riguarda la **betweenness centrality**, i risultati ottenuti sono mostrati in Figura 5.4.

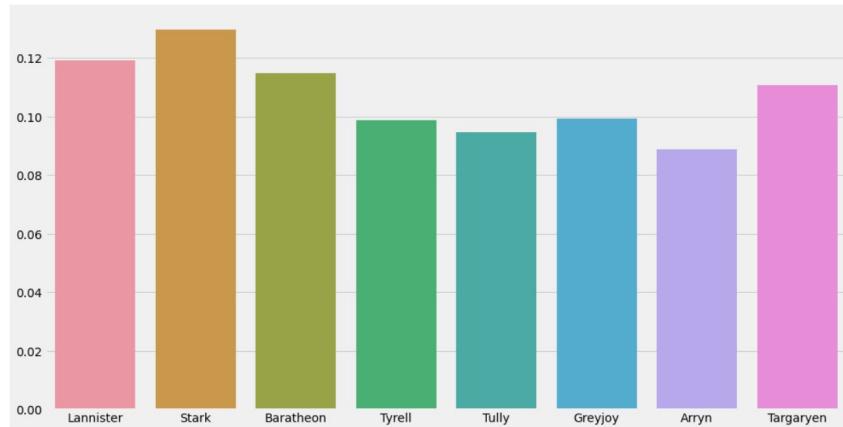


Figura 5.4: Group Betweenness Centrality

Riguardo la betweenness centrality ciò che salta all'occhio è il valore piuttosto elevato della casa Targaryen. I membri della famiglia per la maggior parte sono morti e quindi solo citati ai fini della narrazione. Questo score si può giustificare facendo riferimento alla figura di Daenerys Targaryen poichè all'inizio della serie si trova in una parte di mondo lontana da dove si svolgono le azioni ma più avanti ritornerà portando con se un enorme esercito e quindi una sua comunità. Questo dimostrerebbe quindi come Daenerys sia un nodo che mette in comunicazione due parti abbastanza lontane della rete.