UNIVERSITÀ DEGLI STUDI MILANO - BICOCCA

Esame Sistemi Complessi: Modelli e simulazione

Progetto STN

Spreading Tweet News

Andrea Guzzo - 761818
Manuel Zanaboni 816105
Vittorio Maggio - 817034
Lidia Alecci - 852501

Indice

Sommario

Una persona passa mediamente 5 anni e 4 mesi della propria vita sui social network, secondo uno studio condotto dall'agenzia di marketing americana Mediakix [7]. Sui social network ci informiamo e discutiamo di tutto ormai, eppure non è sempre un'idea saggia credere a tutto quello che si legge su internet, alcune notizie sono infatti pilotate per raggiungere in minor tempo più persone possibili. Tale obiettivo è raggiunto impiegando i bot. Quest'ultimi, come qualunque cosa, possono essere sfruttati per motivi nobili (informare di fatti reali che necessitano di raggiungere la popolazione il più in fretta possibile) o per diffondere fake news. In tale documento verrà discussa la posizione migliore dei bot all'interno di una rete per ottimizzare il numero di persone raggiunte rispetto al tempo impiegato per effettuarne la diffusione.

Introduzione

Oggigiorno i social network rivestono un ruolo chiave nella vita di tutti i giorni, essi infatti influenzano tutti gli aspetti della nostra vita: dal marketing allo shopping, passando per le interazioni con le celebrità ed aziende fino a sostituire, in molte occasioni, le fonti di informazioni primarie come quotidiani o notiziari.
Un report del 2018 effettuato dal Pew Internet research center [10] ha verificato che della popolazione italiana il 64% usa i social media per informarsi su fatti di cronaca e l'81% lo usa per informarsi e ricercare pareri su servizi o brand.
Tuttavia, soprattutto negli ultimi anni, i social hanno rivestito un ruolo chiave anche nel mondo politico, sia a livello di opinioni politiche che di possibilità di influenza sul voto degli elettori; sono presenti numerosi studi che hanno indagato tali fenomeni.
"Activism in the social media age" [9] ha preso in esame il 53% della popolazione americana che nell'arco di un anno ha utilizzato i social per ragioni politiche o sociali. Inoltre, da altri sondaggi è emerso che il 69% della popolazione pensa che i social siano importanti per informare i politici dei problemi che i cittadini si trovano ad affrontare; e il 58% ritiene che i social media influenzano le decisioni politiche.
Su quest'ultimo punto un'altra importante ricerca "Automating power: Social bot interference in global politics" [29] scende più nello specifico indagando come i Bot sono stati importanti nel veicolare pensieri politici e quindi quanto siano potenzialmente necessari e utili per influenzare le masse. Tale ricerca prende in esame le elezioni di vari paesi e,sulla base di altre ricerche effettuate nel corso degli anni, definisce a quale scopo sono stati impiegati i political bot e da chi probabilmente sono stati attivati (se dallo stato stesso o da aziende esterne).
Lo studio effettuato, le cui specifiche sono descritte nel presente documento, ha l'obiettivo di quantificare l'importanza di selezionare, all'interno di una determinata rete, gli utenti che sono nella posizione migliore di influenzare le masse e che quindi trasformandoli in bot permettono di massimizzare la propagazione di una determinata notizia o pensiero.

1 Richiami alla teoria

In questo capitolo verranno descritti alcuni concetti teorici necessari alla comprensione del progetto e che sono stati utilizzati per modellare il problema. È importante sottolineare come la ricerca scientifica e la progettazione multi-agente su progetti in ambito Social Network Analysis sia relativamente scarsa in quanto esistono differenti punti di vista per affrontare e modellare questo tipo di problemi. Molti lavori in ambito accademico studiano la situazione attraverso considerazioni teoriche sull'analisi a posteriori delle reti sociali, risultando in moltissime metriche e parametri che spesso si dimostrano essere discordanti.

1.1 Grafo

Un grafo è una struttura relazionale formata da un numero finito di V di vertici (o nodi) e un numero finito E di segmenti (archi o spigoli) che collegano tra di loro uno o più nodi. Una definizione più formale è la seguente: "Un grafo è una relazione n-aria su un insieme finito S definita dai sottoinsiemi di S con n elementi che soddisfano una proprietà $P(1, ..., n)$". Con ordine del grafo viene inteso il numero di nodi esistenti, mentre gli archi di un grafo sono definiti appunto da una relazione (ad esempio binaria) e i nodi che la compongono sono detti estremi dell'arco incidente tra i vertici (o nodi). Il numero degli archi determina inoltre la dimensione del grafo. [2]
Un grafo orientato G invece è una coppia (V, E) dove V (insieme dei vertici) è un insieme finito ed E è una relazione binaria di V. Se (u, v) è un arco di un grafo $G = (V, E)$ diciamo che il vertice v è adiacente al vertice u. Dato un grafo G orientato, il grado uscente (out-degree) di un vertice è il numero di archi che escono dal vertice; il grado entrante (in-degree) è il numero di archi che entrano nel vertice. Un cammino (path) da un vertice v_0 ad un vertice v_n è una lista ordinata di archi $P = (v_0, v_1), (v_1, v_2), \ldots, (v_{n-1}, v_n)$, e n corrisponde alla lunghezza di questo cammino.
Un grafo orientato G si dice completo quando ogni coppia di vertici è collegata da una coppia simmetrica di archi. La definizione è analoga al caso in cui il grafo non sia orientato, con la differenza che, in quest'ultimo, ogni coppia di archi opposti situata tra due nodi è sostituita da un solo arco non orientato. Il numero di archi in un grafo non orientato completo è pari a $n(n-1)/2$ dove n è il numero di vertici del grafo. Se si escludono i cappi (self-loops), allora un grafo orientato completo è composto da $n(n-1)$ archi.
La densità della rete può essere espressa come il rapporto tra il numero di archi esistenti e il numero di archi possibili (come ad esempio un grafo di densità al 50% sarà composto da un numero di archi che è pari alla metà del totale degli archi possibili). [1]

1.2 Agenti Intelligenti

Secondo la definizione di Russel e Norving [12], un Agente intelligente (o meglio un Learning agents) è un "qualsiasi cosa possa essere vista come un sistema che percepisce il suo ambiente attraverso dei sensori e agisce su di esso mediante attuatori". Questa frase definisce un generico agente coinvolto in qualche attività all'interno dei confini di un ambiente, esso può percepire l'attuale astrazione dell'ambiente usando dei sensori e influenzare direttamente lo stato successivo dell'ambiente o di se stesso usando degli attuatori. Le parole "stato corrente" o "stato successivo" non sono casuali, poiché l'interazione tra l'agente e l'ambiente è controllata dal flusso del tempo. La scelta dell'azione dell'agente non solo dipende dalla percezione corrente dell'ambiente, ma potrebbe anche dipendere dalla sequenza

di percezioni fino a quell'istante. Pertanto, il comportamento dell'agente è definito da una funzione di azione che consente di mappare una sequenza di percezioni ad una determinata azione.
Un agente che fa sempre l'azione corretta, o quella che ci si aspetta, è chiamato agente razionale. La valutazione della funzione di azione non indica quale sia l'azione giusta da compiere in un dato istante infatti, per dare all'agente la capacità di comprendere la bontà delle proprie azioni, è importante coinvolgere una misura di performance, un feedback dato all'agente generato tramite l'analisi della sequenza di percezioni passate rispetto alle probabili mosse future. Ciò che è razionale in un dato istante dipende quindi dalla misura della performance, dalla conoscenza dell'ambiente pregressa, dalle azioni disponibili in quel determinato istante e dalla sequenza di percezioni fino al momento della valutazione. Esistono molte definizioni di agenti e numerose classificazioni. Di suguito viene presentata una classificazione efficace che caratterizza un agente in base alla sua complessità.

1.2.1 Agente reattivo semplice

Questi agenti scelgono le azioni sulla base della percezione corrente ignorando tutta la storia percettiva precedente. L'intelligenza dell'agente è quindi ridotta solo alla situazione attuale dell'ambiente, seguendo la regola "se succede una condizione, allora esegui quell'azione". Questo meccanismo dell'agente funziona solamente quando l'ambiente è completamente osservabile, altrimenti si perde una o più mappature di condizione-azione con conseguente impossibilità di selezionare l'azione migliore.

1.2.2 Agenti reattivi basati su modello

Il modo più efficace di gestire l'osservabilità parziale, per un agente, è tenere traccia della parte del mondo che non può vedere nell'istante corrente. Questo significa che l'agente deve memorizzare una sorta di stato interno che dipende dalla storia delle percezioni e che quindi riflette almeno una parte degli aspetti non osservabili dello stato corrente. La struttura interna descrive la parte dell'ambiente che non può essere percepita e viene anche definita come "modello del mondo". Questo agente quindi non solo è limitato alla piena osservabilità, ma può anche gestire ambienti parzialmente osservabili.

1.2.3 Agenti basati su obiettivi

Conoscere lo stato corrente dell'ambiente non sempre basta a decidere che cosa fare, oltre alla descrizione dello stato l'agente ha quindi bisogno di qualche tipo di informazione riguardante il suo obiettivo (goal) che descriva situazioni desiderabili come ad esempio raggiungere la destinazione richiesta del passeggero nel caso di un taxi a guida autonoma. Questo processo e questa espansione nella comprensione dell'agente porta ad un diverso processo decisionale in quanto questo agente sa quali sono le azioni che lo aiuteranno a raggiungere lo stato dell'obiettivo e come può comportarsi per raggiungerlo rispetto all'ambiente.

1.2.4 Agenti basati sull'utilità

Nella maggior parte degli ambienti gli obiettivi, da soli, non bastano a generare un comportamento di alta qualità. Gli obiettivi forniscono solamente una distinzione binaria tra stati "contenti" e "scontenti", laddove una misura di prestazione più generale dovrebbe permette di confrontare stati del mondo differenti e misure precise qualora l'agente riuscisse a raggiungerli. Uno stato del mondo

preferibile rispetto all'obiettivo dell'agente ha quindi una maggiore utilità e a tal proposito si utilizza una funzione di utilità che consente di assegnare ad uno stato (o ad una sequenza di stati) un numero reale che quantifica il grado di contentezza a esso associato (de facto, la misura delle prestazioni che un altro agente o un uomo nel mondo reale ha già eseguito in precedenza per valutare quello stato o quella configurazione di stati in base ad una conoscenza a priori). Pertanto un agente razionale basato sull'utilità sceglie l'azione che massimizza l'utilità prevista delle azioni scelte.

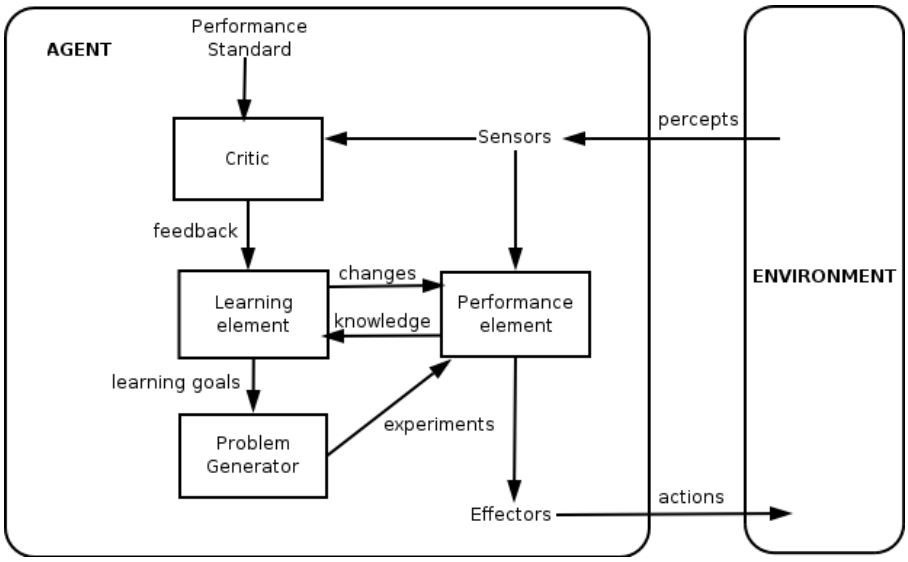

Figura 1: Schema generale di un agente e dell'ambiente

1.3 Sistemi multi-agente

Un sistema multi-agente è un sistema computerizzato (inteso anche come sistema distribuito e aperto) composto dalla multipla interazione di differenti agenti intelligenti. Questo tipo di sistemi sono in grado di risolvere problemi che sono difficili o impossibili per un solo ed unico agente o sistema monolitico. Gli agenti all'interno di questo sistema possono agire in modo indipendente o cooperare tra di loro, inoltre la cooperazione tra questi agenti prevede che essi si scambino dei messaggi tra di loro seguendo dei protocolli o che siano in grado di comunicare rispetto differenti livelli e scopi come: l'obiettivo, l'ambiente, la storia, ecc... I due aspetti principali dei sistemi multi-agente sono:

- **Autonomia**: ovvero la possibilità di un agente di agire liberamente all'interno dell'ambiente rispettando sempre un protocollo e in base all'obiettivo dell'agente.
- **Eterogeneità**: nei sistemi multi-agente i protocolli devono specificare il significato dei messaggi in quanto gli agenti interagiscono sulla base dei significati delle loro comunicazioni.

All'interno di questo paradigma esiste un altro concetto importante ovvero la coordinazione. Esso è un concetto chiave per lo studio di attività complesse in un sistema dinamico e si occupa di gestire le dipendenze attraverso le varie attività compiute, o che devono essere ancora avvenire, dagli agenti. Anche le interazioni possono avvenire in modo differente, ovvero in modo diretto o indiretto.

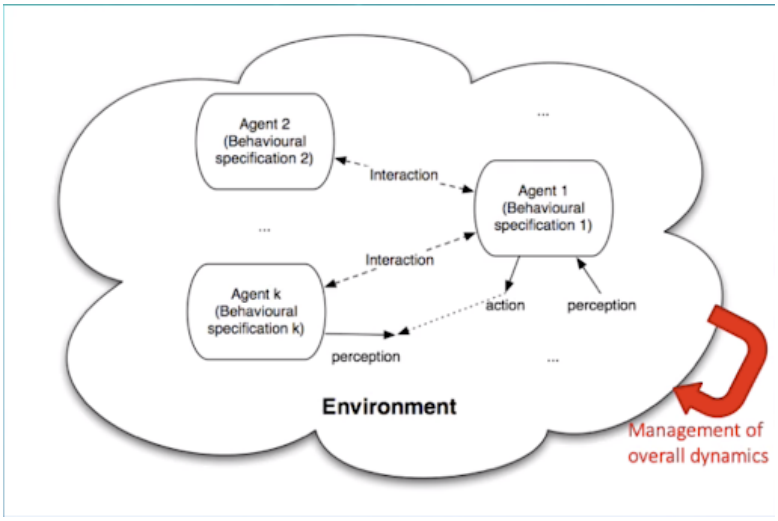

Figura 2: Modello di riferimento di un sistema multi agente all'interno di un ambiente

1.4 Rete sociale ad agenti

Una rete sociale [6] è una struttura sociale costruita da un gruppo di attori (come ad esempio individui o organizzazioni), una serie di legami diadici (ovvero gruppi che condividono degli interessi comuni e comunicano tra di loro) e altre interazioni sociali tra gli attori. In particolare la prospettiva delle reti sociali fornisce un insieme di metodi per analizzare la struttura di entità sociali e una varietà di conseguenti teorie che possono venire spiegate attraverso la costruzione di modelli osservati all'interno di queste strutture. Alcuni esempi di questi modelli locali o globali all'interno di queste reti sono l'individuazione di entità influenti oppure l'esame delle dinamiche di evoluzione e adattamento della rete stessa. L'ambito di ricerca sull'analisi delle reti sociali è certamente interdisciplinare in quanto coinvolge aspetti sociali, psicologici, statistici, informatici, matematici e sicuramente è strettamente legato alla ricerca e allo studio dei complex networks e della network science.
Una rete sociale simulata è appunto una rappresentazione della realtà che presenta determinate caratteristiche. Spesso queste simulazioni però non si relazionano bene con diversi aspetti sociologici delle reali reti sociali in quanto le simulazioni vengono eseguite solamente con reti regolari, casuali, a piccolo mondo e dall'attaccamento preferenziale che non riescono a catturare le sfumature sociologiche degli aspetti sociali. Hamill e Gilbert [4] hanno quindi definito delle reti sociali basate sul concetto di agente in modo tale da inserire alcuni aspetti chiave e rilevanti precedentemente non considerati.
Esistono molte reti differenti appartenenti al mondo delle reti sociali, in particolare le reti sociali personali (o ego-centriche) hanno determinate caratteristiche specifiche:

- Hanno una dimensione limitata e il loro limite dipende dal tipo di relazione che si sta studiando
- Variazione degli individui secondo una distribuzione con coda a destra con un alto grado di connessione per le relazioni molto forti
- Altro raggruppamento, ovvero i membri della rete personale di un individuo dovrebbero tendere a conoscersi per riflettere l'omofilia

- Cambiano nel tempo

Ma più in generale un modello di una rete sociale dovrebbe avere:

- Una bassa densità di rete nel suo interesse: ovvero solo pochi dei reali e potenziali collegamenti all'interno di una rete dovrebbero esistere
- Assortatività positiva del grado di connettività: le persone che hanno una grande rete sociale e conoscono molte altre persone hanno più possibilità di conoscere altre persone che hanno anch'esse un elevato numero di connessioni
- Comunità e gruppi di persone saranno molto connesse tra loro stessi, ma avranno poche connessioni con le persone al di fuori della struttura sociale
- È possibile raggiungere le altre persone all'interno della rete con un piccolo numero di passi (concetto di distanza all'interno della rete).

All'interno di una rete sociale ad agenti possiamo assistere quindi a differenti e molteplici modellazioni differenti, solitamente in rappresentazioni di questo tipo gli agenti corrispondono agli utenti, alle persone interne alla rete e quindi vengono modellati affinchè abbiano delle caratteristiche, delle peculiarità e cerchino di relazionarsi con gli altri agenti attorno a loro ad esempio all'interno di comunità.
Ci sono moltissimi aspetti da considerare nella costruzione di una rete sociale con qualsiasi tecnica di applicazione così come altrettante informazioni e considerazioni possono essere fatte a priori una volta costruito il modello e analizzata la rete simulata.
Definiamo quindi alcuni concetti fondamentali che caratterizzano qualsiasi tipo di rete.

1.4.1 Comunità

Una comunità è un insieme di individui che condividono uno stesso ambiente, sia esso fisico e/o tecnologico, formando un gruppo riconoscibile unito da vincoli organizzativi, linguistici, religiosi, economici e da interessi comuni. [19] Il concetto di comunità è cambiato nel corso della storia umana e si è evoluto al pari passo delle grandi invenzioni tecnologiche come Internet. All'interno del web troviamo il concetto di Comunità Virtuale [20] diffusa universalmente dal libro The Virtual Community di Hogward Rheingold [11]. Una comunità virtuale o comunità online è, nell'accezione del termine, un insieme di persone interessate ad un determinato argomento, o con un approccio comune alla vita di relazione, che corrispondono tra loro attraverso una rete telematica, come internet, una rete di telefonia, un social network, costruendo una rete sociale con caratteristiche peculiari. È importante sottolineare che all'interno di queste community esiste un concetto di identità proprio all'interno della piattaforma che può o non può corrispondere ad un'identità reale ed un concetto di linguaggio specifico utilizzato all'interno della piattaforma che può portare alla generazione di slang particolari all'interno della stessa. È infine importante considerare il "senso di comunità" [27] che unisce i membri all'interno di una stessa comunità che introduce concetti come l'essere membro, l'influenza all'interno del gruppo, l'integrazione e la condivisione di connessioni emotive. Possiamo quindi evidenziare come la definizione stessa di comunità comporta moltissime accezioni e definizioni differenti a seconda del contesto, della materia o del punto di vista considerato. All'interno di questo studio siamo principalmente interessati al concetto di comunità virtuale e del senso di comunità esistente all'interno dei social network, in particolare a Twitter.

1.4.2 Densità della rete

All'interno di una rete, e considerando la social network analysis, la densità viene definita come: tutte le possibili "connessioni" esistenti tra i nodi della rete (partecipanti o utenti). È quindi il numero di connessioni che ogni partecipante ha, diviso per il totale delle possibili connessioni che lo stesso utente potrebbe avere. [3]

1.4.3 Lunghezza del cammino (Distanza)

In matematica e nella teoria dei grafi la distanza tra due vertici in un grafo è espressa come il numero di archi in un percorso più breve (chiamato anche geodetico) che li collega. [22] Tra due vertici inoltre possono esistere più percorsi minimi differenti. Se non esiste però un percorso che collega due vertici del grafo, ovvero se appartengono a due componenti non collegate tri di loro, allora convenzionalmente la loro distanza è infinita. Nel caso di un grafo diretto, la distanza $d(u, v)$ tra due vertici u e v è definita come la lunghezza del percorso diretto più breve da u a v costituito da archi, purché esista almeno uno di questi percorsi.

1.4.4 Connessione

In matematica e informatica, la connettività è uno dei concetti di base della teoria dei grafi ed è una misura importante per la resilienza della rete stessa: corrisponde al numero minimo di elementi (nodi o archi) che devono essere rimossi per separare dei nodi in sottografi isolati. [21]
In un grafo indiretto G, due vertici u e v sono detti collegati se G contiene un percorso u a v, altrimenti sono chiamati scollegati. Quando i due vertici sono collegati da un percorso di $lunghezza = 1$, cioè da un singolo bordo, i vertici sono chiamati adiacenti.
Un grafo si dice connesso se ogni coppia di vertici del grafo è connessa, questo comporta che esista un percorso per ogni coppia di vertici, quindi un grafo non diretto che non è collegato viene detto disconnesso. Un grafo non diretto G è quindi scollegato se esistono due vertici in G in modo tale che nessun percorso in G abbia questi vertici come punti finali. Un grafo con un solo vertice è collegato. Un grafo senza archi con due o più vertici è disconnesso.
Infine, un grafo diretto è chiamato debolmente connesso se la sostituzione di tutti i suoi archi diretti con archi non diretti produce un grafo connesso (non diretto). È collegato unilateralmente (detto anche semiconnesso) se contiene un percorso diretto da u a v o un percorso diretto da v a u per ogni coppia di vertici (u, v). Mentre è fortemente collegato, o semplicemente forte, se contiene un percorso diretto da u a v e un percorso diretto da v a u per ogni coppia di vertici (u, v).

1.4.5 Invarianza di scala

Un concetto fondamentale all'interno delle reti sociali e anche della rete del World Wide Web è proprio la definizione di rete ad invarianza di scala. Una rete ad invarianza di scala si definisce tale quando si considera un grafo che ha un numero di connessioni tra vertici ed archi sotto forma di un "esponenziale negativo" e quindi invariante rispetto ai cambiamenti di scala. Più nel dettaglio il numero di due tipi di nodi, ad esempio uno con 10 connessioni e uno con 15 hanno tra di loro una proporzione che è $\exp(-1(Nb - Na))$ dove Nb ed Na sono il numero di nodi del denominatore

e numeratore rispettivamente mentre a è un parametro del tipo di rete considerato. Questa legge è detta legge di potenza, di cui a è il parametro. [26]
Empiricamente questa definizione comporta che ci siano degli hub molto densi di nodi e vertici (una comunità o un gruppo di persone) e che quando un nodo deve cercare dei nuovi collegamenti esso lo faccia prima verso un particolare nodo che fa parte di un hub e che ha già molti collegamenti portando il grafo nel suo interesse ad una crescita esponenziale con l'aumentare del numero di collegamenti della rete.
La presenza degli hub e di questo comportamento è alla base della "Teoria del piccolo mondo" anche chiamata dei "6 gradi di separazione" in quanto ogni rete complessa in natura è tale che due qualunque nodi possono essere collegati da un percorso costituito da un numero relativamente piccolo di collegamenti matematicamente determinabile. [28]

1.5 Misure di valutazione

Come abbiamo visto precedentemente, nelle reti sociali è importante definire delle misure per spiegare e confrontare un modello o una simulazione rispetto ad un caso reale. A tal proposito sono state introdotte numerosissime misure e metriche considerabili per valutare la bontà di una rete sociale, esse derivano in grande maggioranza dalla teoria dei grafi in quanto una rete sociale non è nient'altro che un particolare grafo rappresentativo di una realtà o di un sistema.

1.5.1 Grado e ordine di un grafo

L'ordine di un grafo [25] è $|V|$ (il numero dei vertici). La dimensione di un grafo è $|E|$. Il numero di archi incidenti in un vertice $v \in V$ (cioè il numero di archi che si connettono ad esso) prende il nome di grado del vertice v, dove un arco che si connette al vertice ad entrambe le estremità (un cappio) è contato due volte. Si considerano il "grado massimo" e il "grado minimo" di G come, rispettivamente, il grado del vertice di G con il maggior numero di archi incidenti e il grado del vertice di G che ha meno archi incidenti. Quando il grado massimo ed il grado minimo coincidono con un numero k, si è in presenza di un grafo k-regolare (o più semplicemente grafo regolare). Per un arco u, v, i teorici dei grafi usano solitamente la notazione più sintetica uv. Un grafo $G = (V, \varnothing)$ privo di archi è detto grafo nullo. Un caso estremo di grado nullo è quello del grafo $G = (\varnothing, \varnothing)$, per il quale anche l'insieme dei nodi è vuoto.
Un grafo è definito completo se due qualsiasi dei suoi vertici sono adiacenti (esiste un arco che li connette). La massima cardinalità di un sottografo completo del grafo si chiama densità del grafo.

1.5.2 Densità di un grafo

Sia definito il grafo $G = (N, A)$ come coppia dei due insiemi $N = 1, 2, 3, ..., n$ ed A sottinsieme del prodotto cartesiano $N\chi N$. N sarà l'insieme degli n nodi che compongono il suddetto grafo e A l'insieme degli archi.

- sia n la cardinalità di N (ovvero il numero dei nodi di un grafo)
- sia L la cardinalità di A (ovvero il numero degli archi dello stesso grafo)

Se le coppie di nodi si considerano ordinate il grafo è detto orientato o digrafo, altrimenti si dice non orientato o semplice. Il grafo è detto pesato se ad ogni arco è associato un valore che rappresenta un peso/costo.
La densità di un grafo semplice (Δ) o non orientato è definita come:

$$\Delta = 2L/n(n-1) \tag{1}$$

La densità di un grafo (Δ) orientato è definita come:

$$\Delta = L/n(n-1) \tag{2}$$

Nel caso di grafi pesati ad L occorre sostituire la sommatoria dei pesi di ciascun arco. La densità di un grafo assume valori compresi tra 0 ed 1 e pertanto si può ricollegare facilmente al concetto di probabilità. La densità di un grafo misura la probabilità che una qualsiasi coppia di nodi sia adiacente, mentre la connessione di un grafo dipende dalla distribuzione degli archi tra i nodi.
Un grafo sconnesso può avere densità maggiore di uno connesso a causa della concentrazione degli archi tra un ristretto numero di nodi.

1.5.3 Centralità

Nella teoria dei grafi e nell'analisi delle reti, gli indicatori di centralità [24] identificano i vertici più importanti all'interno di un grafo. Le applicazioni includono l'identificazione della persona o delle persone più influenti in un social network, i nodi infrastrutturali chiave in Internet o nelle reti urbane e i super diffusori di malattie.
Ci sono moltissime misure di centralità che possono essere utilizzate all'interno di un grafo per studiare le caratteristiche e le peculiarità, un'importante misura frequentemente utilizzata è la Betweenness. [23]

1.6 Modelli epidemiologici compartimentali

Con l'avvento del COVID-19 i modelli epidemiologici si sono rivelati fondamentali per comprendere e combattere le situazioni di emergenza che si sono venute a creare nei vari stati. I primi modelli matematici, anche quelli più utilizzati, che permettono di descrivere i processi di diffusione epidemici arrivano dai lavori di Kermack e McKendrick (1932) e rappresentano ancora oggi uno degli approcci fondamentali per studiare molteplici processi di diffusione dalle malattie, dai virus nel mondo naturale, passando per i virus informatici fino all'analisi di opinioni e contenuti sui social network.
Questi modelli vengono anche chiamati modelli epidemiologici compartimentali [18] perchè alla base del loro ragionamento c'è l'assegnazione di una particolare fetta di popolazione a determinati comportamenti in base a delle caratteristiche epidemiologiche che gli individui hanno. Man mano che l'epidemia evolve e si adatta le persone assumeranno caratteristiche diverse (contrarranno la malattia, guariranno, ...) e quindi cambieranno compartimento allo scorrere del tempo. Lo studio dell'evoluzione delle persone in base alle loro caratteristiche e all'appartenenza ad un compartimento è alla base della costruzione di questi modelli.
Il modello SIR è composto da tre stadi primari che le persone possono assumere rispetto ad una malattia:

- (S) suscettibile, ovvero l'individuo è vulnerabile all'infezione, ma non è ancora stato contagiato
- (I) infetto, ovvero l'individuo è stato contagiato e può diffondere la malattia
- (R) rimosso, l'individuo è guarito e ha recuperato lo stato di salute, oppure è deceduto

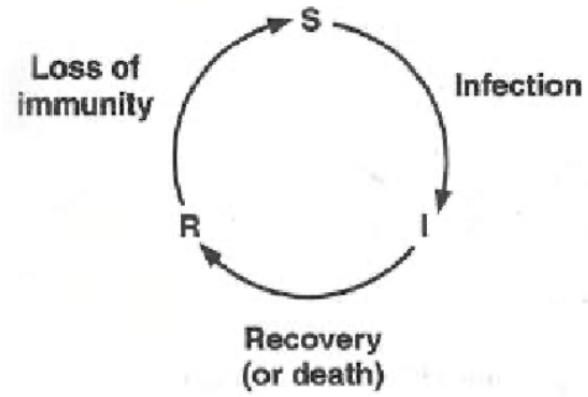

Figura 3: Schema di riferimento per il modello SIR

Oltre alla caratterizzazione delle persone esistenti all'interno di uno scenario, il modello SIR considera una distribuzione di probabilità da parte delle persone suscettibili di essere infettate che dipende strettamente dal grado di infettività di una malattia. Nei primi lavori in questo campo per semplificare le studio delle dinamiche, lo studio della probabilità di diffusione venne influenzata dalla numerosità delle popolazioni di infetti e suscettibili e dalla variazione dei parametri di controllo: il grado di infettività e il tasso di recupero.
In tale modello l'infezione segue un andamento di crescita logistica, quando l'epidemia inizia si ha una piccola porzione di elementi infetti. Al passo successivo spesso si evidenzia una fase di lenta crescita che corrisponde anche al momento migliore per bloccare la diffusione della malattia nonostante la difficoltà nella distinzione dell'epidemia e da casi sparsi non correlati tra di loro. Se la diffusione procede nel tempo si entra nella fase esplosiva della crescita logistica e diventa molto difficile riuscire a fermarla. L'ultima fase è quella di esaurimento in quanto la popolazione dei suscettibili diventa troppo piccola affinché l'epidemia continui a diffondersi.

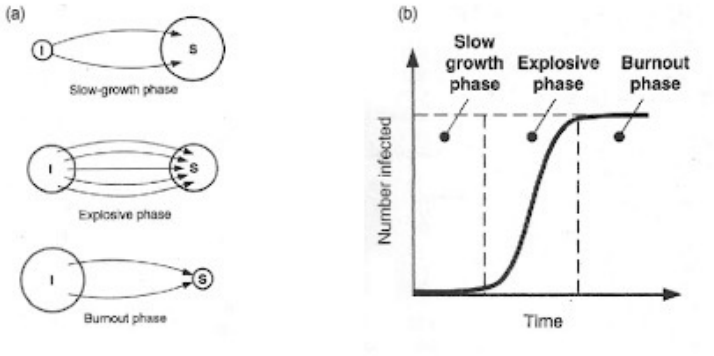

Figura 4: Tassi dei nuovi infetti in dipendenza dalle dimensioni delle popolazioni di suscettibili ed infetti in cui i tassi di crescita sono massimizzati per la fase esplosiva della crescita logistica (a). In (b) il diagramma della crescita logistica con il suo caratteristico andamento della curva ad "S" che mostra una fase di bassa crescita (slow-growth), una fase esplosiva (explosive phase) e una fase di esaurimento (burnout phase).

Il modello a crescita logistica segue quindi nell'arco delle tre fasi un andamento caratterizzato dalla forma ad "S" e la sua diffusione dipende soprattutto dalla capacità di infettare gli individui. La contagiosità (chiamata anche tasso di crescita) è regolata all'interno del modello dal tasso di riproduzione, ovvero il numero medio di nuovi infetti generati da ogni singolo individuo malato. La condizione del tasso di riproduzione affinché la malattia entri nella fase esplosiva è che il tasso sia superiore a 1, qualora fosse inferiore gli infetti verrebbero rimossi più velocemente rispetto al numero dei nuovi infetti. Il tasso di riproduzione rappresenta quindi il concetto di soglia critica dell'epidemia. Questa prima modellazione e spiegazione dei modelli SIR si basavano su reti semplici e casuali tuttavia nel mondo della scienza delle reti troviamo moltissime tipologie di reti differenti che modellano particolari condizioni i cui risultati sono incerti e interessanti da analizzare, ecco perché i modelli SIR e derivati hanno ancora oggi una grande applicazione in quanto consentono di adattarsi bene a moltissime reti differenti pur mantenendo un'elevata capacità espressiva facilmente calcolabile. È importante considerare che nel corso del tempo ci sono state diverse evoluzioni del modello per costruire nuovi compartimenti utili a studiare differenti punti di vista di una situazione epidemiologica (come ad esempio il fatto che una persona non riesca mai a guarire del tutto e continui ad essere contagiosa anche a distanza di molto tempo) o per modellare un determinato problema come ad esempio SEIR (Suscettibile Esposto Infetto Rimosso), SEIS (le persone rimangono sempre nello stato di Suscettibilità anche dopo l'infezione e quindi nel tempo) e altre differenti variazioni e modellazioni.

2 Stato dell'arte

In prima battuta, siamo andati a cercare nella letteratura studi simili, in modo da avere una panoramica su quanto era già stato fatto e quanto necessita o di approfondimento o di totale nuova scoperta. In particolare siamo rimasti colpiti dai seguenti studi.

A Survey of Twitter Rumor Spreading Simulations[14] uno studio che discute varie ricerche di viral marketing e social network effettuate sfruttando le reti di Twitter. In tale paper sono comparati diciotto studi e per ognuno sono indicati e confrontati: tipo del target studiato, metodo impiegato e riproducibilità.

Predicting Information Spreading in Twitter[30] uno studio condotto dalla Microsoft il cui obiettivo è predire i futuri retweet, ciò diviene possibile allenando un modello probabilistico. Questo paper risultava particolarmente interessante in quanto il modello utilizzato, oltre ad essere stato allenato usando i dati su Twitter, presenta una buona flessibilità e potenziale applicabilità in altri contesti e in altre reti.

Rumor Diffusion and Convergence during the 3.11 Earthquake: A Twitter Case Study[15] caso studio molto interessante in cui i rumor sulla causa del terremoto avvenuto in Giappone l'11 Marzo 2011 si sono diffusi velocemente, ma altrettanto velocemente sono stati messi a tacere da un tweet proveniente da una fonte ufficiale (account della pubblica amministrazione). In tale modello, il riconoscimento dei rumors è avvenuto tramite la ricerca di una serie di keyword, le cui modalità di identificazione sono ben specificate all'interno del paper. Una delle cose più interessanti di tale studio è l'applicazione di un modello basato su SIR con 3 possibili stati per ogni utente:

- Ground state (G): utenti che non sono ancora entrati in contatto con il rumor;
- Excited state (E): utenti che credono il rumor sia vero;
- Final state (F): utenti che sanno già che il rumor sia falso.

Epidemiological Modeling of News and Rumors on Twitter[5] tale studio si prefigge come obiettivo il riconoscimento e comprensione dei pattern comunicativi all'interno delle reti di Twitter. Effettuando anche un confronto tra i modelli SIS e il SEIZ, arrivando a concludere che per lo studio effettuato la modellazione SEIZ è stata più accurata, in particolare per catturare la diffusione di informazioni (sia di notizie che di rumors).

Modeling Social Influence in Social Networks with SOIL, a Python Agent-Based Social Simulator[8] descrive l'applicazione del modello Agent-based Social Simulation (ABSS) con Soil, spiegando brevemente il funzionamento affiancato da alcuni esempi pratici. In particolare ne vengono esaltati la facilità di estensione dei comportamenti degli agenti e la potenziale integrazione di algoritmi di machine learning.

Soil: An Agent-Based Social Simulator in Python for Modelling and Simulation of Social Networks[13] paper ufficiale di Soil che insieme alla documentazione e agli esempi integrati nella libreria ci ha permesso di personalizzare i comportamenti degli agenti e di configurare al meglio il nostro modello. In tale paper nella prima parte vengono confrontati varie librerie di diversi linguaggi. Dopo tali confronti viene spiegata la struttura di Soil con qualche richiamo alla teoria.

3 Soil

Soil è un modulo Python ABSS[1]. Ciò vuol dire che permette di sfruttare uno scheletro di base per lanciare simulazioni di modelli sociali tramite la personalizzazione degli agenti.
È stato scelto Soil in quanto è un progetto open source, costantemente in crescita e aggiornato, cross-platform e permette di utilizzare standard di moduli già rodati e ampiamente utilizzati e conosciuti.

3.1 Struttura di Soil

La logica di funzionamento di Soil si articola in due componenti principali: la definizione degli agenti e la configurazione della simulazione.

- **Agenti** - la descrizione di un agente avviene mediante la definizione di una classe dedita all'implementazione del comportamento dell'agente;
- **Simulazione** - configurazione che include vari aspetti della simulazione come numero e tipo di agenti, topologia della rete, nome della simulazione, eventuali parametri d'ambiente etc...

La figura 5 mostra la struttura schematizzata del funzionamento di Soil. Per la definizione della *topologia* della rete viene utilizzata un'istanza di un grafo della libreria `networkx`. In seguito alla definizione di tale topologia, Soil esegue l'associazione tra ogni nodo e il suo corrispettivo agente, quest'ultima azione è controllata dall'ambiente.

[1] Agent-Based Social Simulation

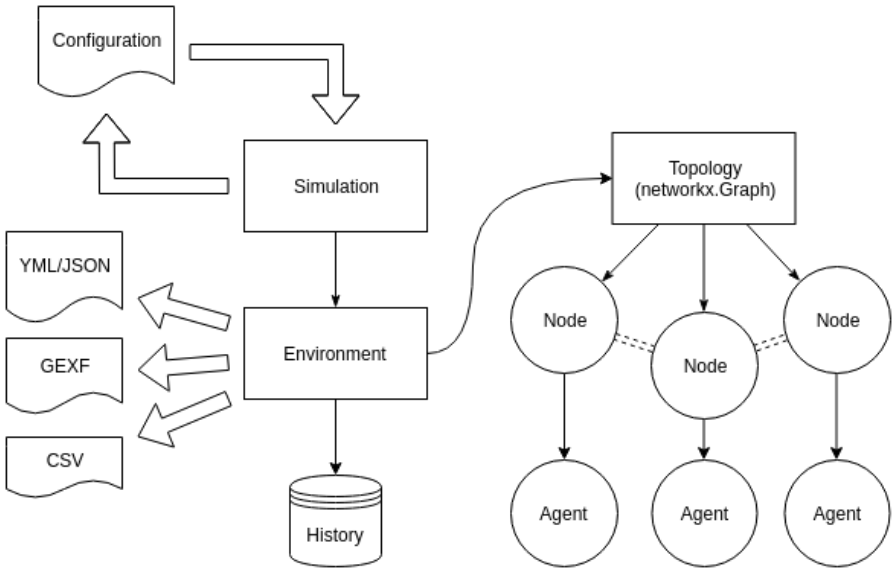

Figura 5: Struttura di Soil

La configurazione di ogni singola simulazione può avvenire a livello di codice o tramite un file di configurazione (con estensione JSON[2] o YAML). L'utilizzo di file di configurazione permette una descrizione dichiarativa e riproducibile.
Riportiamo una rapida descrizione dei parametri configurabili di una simulazione:

- **name** - nome associato alla simulazione;
- **max_time** - numero di step;
- **num_trials** - numero di simulazioni da eseguire;
- **interval** - frequenza di aggiornamento degli stati degli agenti (secondi);
- **network_params** - topologia della rete. Soil rende possibile il caricamento di una rete esistente (tramite la lettura di un apposito file) oppure la creazione di una rete randomica tramite l'utilizzo dei `generatori` forniti dalla libreria networkx;
- **load_module** - nome del modulo python in cui sono definite le classi che implementano gli agenti;
- **network_agents** - permette di definire l'associazione tra nodi e agenti; É possibile associare direttamente il tipo di agente sul nodo specifico oppure definire il ratio di distribuzione di diversi tipi di agenti tramite il parametro `weight`;
- **environment_agents** - simile alla definizione precedenti, tuttavia questi agenti non vengono assegnati ad alcun nodo;

[2] JavaScript Object Notation

- **environment_params** - definizione di parametri dell'ambiente, accessibili da tutti gli agenti. Nell'ambiente viene memorizzato lo stato condiviso della simulazione.

I risultati di una simulazione sono memorizzati automaticamente tramite un database `sqlite`, inoltre è possibile specificare formati aggiuntivi di esportazione, tra i quali: YAML/JSON, GEXF[3] e CSV[4].

Secondo le nostre ricerche, la documentazione di Soil non risulta sempre corretta e aggiornata, tuttavia essendo un progetto open source è possibile ottenere informazioni relative al funzionamento di determinate utility in modo abbastanza agevole.

Nel capitolo seguente verrà spiegato il funzionamento di Soil per quanto riguarda l'uso che ne è stato fatto durante lo sviluppo di questo progetto.

4 Soluzione proposta

Dopo una attenta lettura ai paper, abbiamo impostato le nostre domande di ricerca, in modo da avere ben chiaro in mente l'obiettivo che volevamo prefiggerci di raggiungere con tale progetto. Le domande al quale tale documento cerca di dare risposta sono le seguenti:

- All'interno di una rete sociale, come massimizzare il numero di persone che visualizzano una notizia in un numero di step determinato?
- Dove è possibile posizionare i bot che generano i contenuti per massimizzare la diffusione ed eventualmente minimizzare il tempo di diffusione?
- É possibile distinguere chi è venuto in contatto con una notizia pubblicata da un determinato tipo di utente (bot o opinion leader) e che informazione possiamo ottenere da tali caratteristiche?

Abbiamo scelto e ricostruito la rete di partenza (grafo dei follower del presidente del Consiglio Giuseppe Conte) e su di essa sono state eseguite una serie di simulazioni con Soil. Abbiamo inoltre scelto di monitorare le infezioni (da considerarsi come diffusione di notizie) analizzando le differenze tra infetti ed esposti. Abbiamo posto attenzione anche al tipo di infezione: se diretta e indiretta, distinguendo anche da chi proviene tale infezione: da bot, da opinion leader o da un altro utente.
Tale progetto è implementato totalmente in Python e i tools e le librerie usate sono le seguenti:

- Twint: scraper per Twitter;
- Networkx: per gestire le reti e quindi i grafi;
- Gephi: per la visualizzazione dei grafi;
- Plotly: per visualizzare i grafici finali in modo da avere un riscontro visivo dei risultati;
- Soil: per implementare un modello di simulazione multi-agente;

[3] Graph Exchange XML Format
[4] Comma Separated Values

- Streamlit: per costruire un'interfaccia grafica in modo da modellare e personalizzare il modello in modo immediato e senza dover passare direttamente dal codice.
- Pandas: per gestire i dataframe.

Siamo quindi partiti da una serie di assunzioni e ipotesi per poi andare a validarle tramite il modello proposto. Tali assunzioni ci hanno guidato nell'impostare il modello, inserendo le probabilità che più ci sembravano vicine alla realtà. Le assunzioni prese in considerazione sono le seguenti:

- l'Opinion Leader ha influenza maggiore sulla rete rispetto ai Bot;
- ma i Bot hanno un'influenza più sparsa (ampia);
- la probabilità di infezione tra utenti è molto bassa;
- anche la probabilità che l'utente effettui qualche azione (retweet, commento...) dopo essere entrato in contatto con la notizia è bassa e quindi la probabilità di passare da esposto a infetto si alza di molto solo se molti dei suoi vicini hanno risposto a loro volta alla notizia.

4.1 Grafo

0-5	38353
6-10	41
11-20	21
21-50	34
51-100	13
101-250	10
251-500	12
500+	12

Tabella 1: Grafo 500-users

0-5	105493
6-10	80
11-20	37
21-50	54
51-100	35
101-250	30
251-500	21
500+	35

Tabella 2: Grafo 1000-users

0-5	127327
6-10	102
11-20	50
21-50	67
51-100	47
101-250	45
251-500	27
500+	46

Tabella 3: Grafo 1500-users

0-5	215437
6-10	122
11-20	64
21-50	89
51-100	58
101-250	57
251-500	37
500+	57

Tabella 4: Grafo 2000-users

4.2 Opinion Leader

Il social network scelto per effettuare la simulazione è stato Twitter, in particolare è stata scelta la rete sociale dell'attuale presidente del consiglio Giuseppe Conte (username: @GiuseppeConteIT[17]) che attualmente conta 752.814 follower. Per motivi computazionali non è stato possibile analizzare il grafo completo, quindi sono stati considerati:

- 500 followers e 2 livelli di profondità, per un totale di 38485 utenti
- 1000 followers e 2 livelli di profondità, per un totale di 105774 utenti
- 1500 followers e 2 livelli di profondità, per un totale di 127700 utenti
- 2000 followers e 2 livelli di profondità, per un totale di 215910 utenti

Per scaricare i dati necessari, Twitter mette a disposizione delle API ufficiali, ma troppo limitate in termini di richieste per unità di tempo (nella versione standard per ottenere la lista dei followers sono permesse 15 richieste ogni 15 minuti), in particolare avendo la necessità di scaricare informazioni riguardanti migliaia di utenti.
Si è così scelto di usare uno scraper per velocizzare il processo, in particolare è stato usato lo scraper TWINT[16] (Twint Intelligence Tool).

4.3 Bot

Per determinare la posizione dei Bot all'interno della rete sono state calcolate le principali misure di centralità dei nodi all'interno del grafo, basate su:

In-Degree
Numero di archi entranti di un nodo, misura quanto un utente è seguito.
Formalmente:

$$In - Degree(i) = \sum_{j} x_{ji}$$

dove:
x_{ji} è un arco diretto dal nodo j al nodo i

Betweenness
Numero di volte che un nodo funge da ponte lungo il percorso più breve tra 2 nodi; su una rete sociale corrisponde a quanto un utente si trova sui percorsi di comunicazione tra gli altri utenti.
Formalmente:

$$Betweenness(i) = \sum_{j \neq k} g_{jk}(i)/g_{jk}$$

dove:
g_{jk} è il numero di cammini minimi tra i nodi j e k
$g_{jk(i)}$ è il numero di cammini minimi tra i nodi j e k passanti per il nodo i

Autovettori
Misura l'importanza di un nodo, un nodo è considerato importante se puntato da altri nodi importanti.
Formalmente:

$$EigenvectorCentrality(i) = \frac{1}{\lambda} \sum_{k} a_{k,i} x_k$$

dove:
$A = (a_{i,j})$ è la matrice di adiacenza del grafo.

É stata inoltre, svolta una simulazione considerando una posizione randomica dei Bot.

4.4 Simulazioni

Durante le simulazioni è stato selezionato come Opinion Leader l'esatto nodo che, all'interno del grafo ottenuto grazie allo scraper, rappresenta il Presidente del Consiglio. I Bot all'interno delle simulazioni sono sempre 10 tuttavia differiscono per la loro posizione all'interno della rete. Poiché gli stessi nodi con in-degree maggiore sono gli stessi con betweenness maggiore allora la simulazione sarà unica. Si avranno quindi tre simulazioni:

- Simulation_BTW: simulazione nella quale i nodi selezionati come Bot sono posti nei nodi aventi betweenness maggiore.
- Simulation_Eigenvector: simulazione nella quale i nodi selezionati come Bot sono i 10 nodi considerati più influenti sulla base degli eigenvector score.
- Simulation_Random: simulazione nella quale i nodi selezionati come Bot sono presi in modo random.

4.5 Configurazione di SOIL

4.5.1 File Python

Nel file PY viene descritto il modulo python che verrà poi usato per avviare la simulazione seguendo le direttive e le caratteristiche descritte nel file YML. Per ogni agente viene quindi creata una classe la quale avrà degli stati e dei metodi che ne descrivono i comportamenti e interazioni tra i vari agenti. Nel nostro caso ci sono 3 tipologie di agenti:

- OpinionLeader: è l'utente dal quale è stato generato il grafo e da cui tramite scraper si sono ottenuti i suoi follower e poi i follower dei follower. Nel nostro caso è l'utente Conte.
- Bot: sono i bot che hanno interesse a far espandere una notizia pubblicando spesso contenuti, hanno quindi un potere di influenza mediamente alto. La selezione di questi bot viene effettuata nel file BotSelection.py, tale selezione può avvenire sulla base dei parametri descritti nella sezione 4.3.
- User: sono gli utenti comuni, possono trovarsi in tre stati:
  - not_exposed: l'utente non è entrato per niente in contatto con una determinata notizia o informazione, nè perchè è stato esposto nè guardando la home di tweeter o cercando manualmente (quest'ultima casistica è gestita dal campo prob_search_spread).
  - exposed: l'utente è entrato in contatto con la notizia ma non ha ancora effettuato nessuna azione,quindi non ha mostrato agli altri della sua rete di essere entrato in contatto con tale informazione.
  - infected: l'utente non solo è stato esposto ma ha anche palesato il suo essere entrato in contatto con la notizia effettuando un'azione e quindi potenzialmente è in una condizione in cui potrebbe esporre altri utenti.

All'inizio della simulazione i soli che sono considerati esposti sono l'opinion leader e i bot dai quali appunto si propaga l'informazione. Per una questione di maggior controllo dei risultati finali, e quindi per vedere chi e quanto ha contribuito alla propagazione, è stato introdotto un parametro type. Quest'ultimo ha valore 1 nel caso la propagazione è partita dall'opinion leader, 2 nel caso avviene dai bot; utenti infettati da altri utenti propagano comunque il tipo da cui sono stati infettati, e quindi ad esempio un utente infettato da un utente che era stato in principio infettato da un bot avrà tipo 2 e sarà quindi conteggiato tra quelli infettati dai bot.

4.5.2 File YML

Il file YML racchiude le specifiche delle simulazioni che saranno lanciate con soil.
Tra i parametri personalizzabili particolarmente utili sono:

- max_time: numero di step effettuati al massimo nella simulazione.
- num_trials: parametro tramite il quale si può decidere se rifare la stessa simulazione più volte indicando il numero di volte che sarà lanciata.
- network_params: parametro nel quale si seleziona il grafo su cui sarà effettuata la simulazione, può essere o creato in maniera random o preso da un grafo già creato indicando il file GEXF.
- states: parametro che permette di personalizzare il singolo nodo tramite il suo id. Ciò ci ha permesso di selezionare la posizione sempre fissa di Conte e di verificare come le simulazioni cambiassero in base alla marcatura di alcuni nodi come Bot sulla base di dei parametri specifici, indicati nella sezione 4.3.

4.6 Web app

Per ragioni di riproducibilità più agevole e per estensibilità è stato scelto di sviluppare una web app che permettesse l'esecuzione di tutte le fasi contemplate dall'app e anche di ottenere i risultati in forma di grafici.
Per lo sviluppo della webapp è stata usata la libreria streamlit. Streamlit è un framework open-source in Python. E' particolarmente utile per data scientists e sviluppatori che lavorano nel campo di machine learning.

Di seguito si illustrano le varie fasi dell'app.

4.6.1 Fasi dell'app

Download dati da Twitter
In questa fase abbiamo utilizzato lo scraper TWINT per ottenere i followers di Conte e, a loro volta, i followers dei followers di Conte. Il risultato ottenuto è un file .csv per ogni utente di Twitter, all'interno del quale sono segnati gli user dei followers.
Creazione del grafo
Da questi file .csv ottenuti dalla fase precedente, si è creato il grafo orientato (secondo la relazione di "Follow") tramite la libreria networkx.
Posizionamento Bot e simulazione SOIL
Oltre a configurare e far eseguire le simulazioni di soil, così come spiegato nella sezione 4.5, in tale fase sono state effettuate le principali misure di centralità dei nodi nel grafo in modo da determinare la posizione dei Bot, ed effettuate le simulazioni appropriate.
Visualizzazione grafo risultante
In questa fase vengono mostrati i risultati delle simulazioni in forma di grafo. I nodi del grafo sono stati colorati differentemente a seconda dello stato dell'agente corrispondente (non esposto, esposto, infetto), inoltre, viene riportato anche il tipo di nodo che ha causato la diffusione/contagio.

Statistiche sulla diffusione
A questo punto abbiamo provveduto a calcolare le statistiche ottenute dai risultati, andando a calcolare le percentuali dei non esposti, esposti, infetti, per ogni fase e da chi sono stati infettati o esposti (bot, user, opinion leader).

4.6.2 Uso di streamlit

L'app è strutturata con un pannello di configurazione a sinistra.

Figura 6: Screen dell'app

Da tale pannello è possibile:

- Scegliere i parametri per il twitter scraper:
  - Scegliere lo username dell'account su cui avviare lo scraper.
  - Salvare il csv ottenuto.
  - Selezionare il path nel quale salvare il csv contenente i follower dell'account scelto.
  - Specificare a che livello scendere nel grafo dei follower (1 o 2).
- Parametri per generare il grafo:
  - Selezionare il path da cui prendere le informazioni (csv).
  - Selezionare la cartella da cui prendere il secondo livello del grafo (se presente).
  - Scegliere il nome del grafo.
  - Selezionare se salvare il grafo come diretto o indiretto.
  - Numero di follower da considerare.

- Parametri per la selezione dei Bot:
  - Path del grafo (.gexf).
  - Selezione del metodo con cui calcolare i bot.
  - Selezionare il numero dei bot da inserire nella rete.
- Parametri per la simulazione di SOIL:
  - Path del file yml a partire dal quale configurare la simulazione.
  - Scegliere il nome della simulazione.
  - Selezionare il path della cartella principale dedicata alle simulazioni.
  - Scegliere il numero massimo di iterazioni della simulazione.
  - Scegliere il numero di volte che la simulazione sarà lanciata con quei parametri.
  - Selezionare il file gexf contenente le caratteristiche del grafo.
- Parametri per la creazione dei grafici e dei risultati:
  - Selezione del file del grafo di partenza (.gexf).
  - File csv contenente l'output della simulazione di SOIL.
  - Nome della simulazione, sulla base della tipologia di selezione dei bot (random, btw o eigenvector).
  - Numero di step effettuati durante la simulazione sul grafo.
  - Visualizzare o no graficamente il grafo.
- Parametri per calcolare le statistiche finali:
  - Nome della simulazione, sulla base della tipologia di selezione dei bot (random, btw o eigenvector).
  - Numero di step effettuati durante la simulazione sul grafo.

5 Risultati ottenuti

Di seguito sono riportati i risultati ottenuti con le diverse configurazioni.

5.1 Random

5.1.1 Grafo da 500 followers

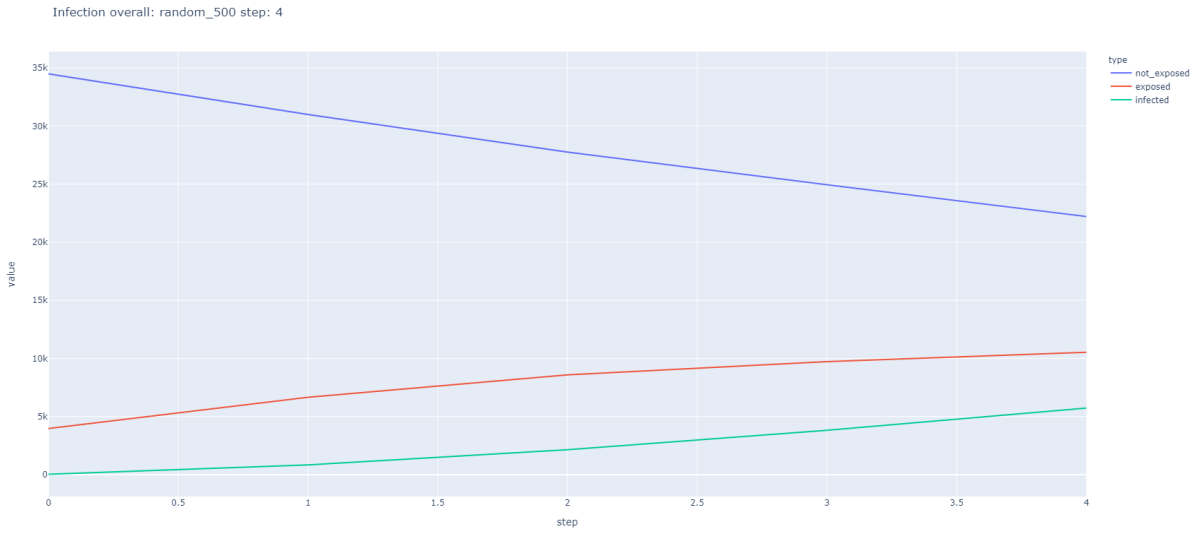

Figura 7: Infection overall

	Step 0	Step 1	Step 2	Step 3	Step 4
Not exposed	34487	30989	27757	24945	22214
Exposed	3964	6659	8586	9721	10538
Infected	34	837	2142	3819	5733

Exposed:

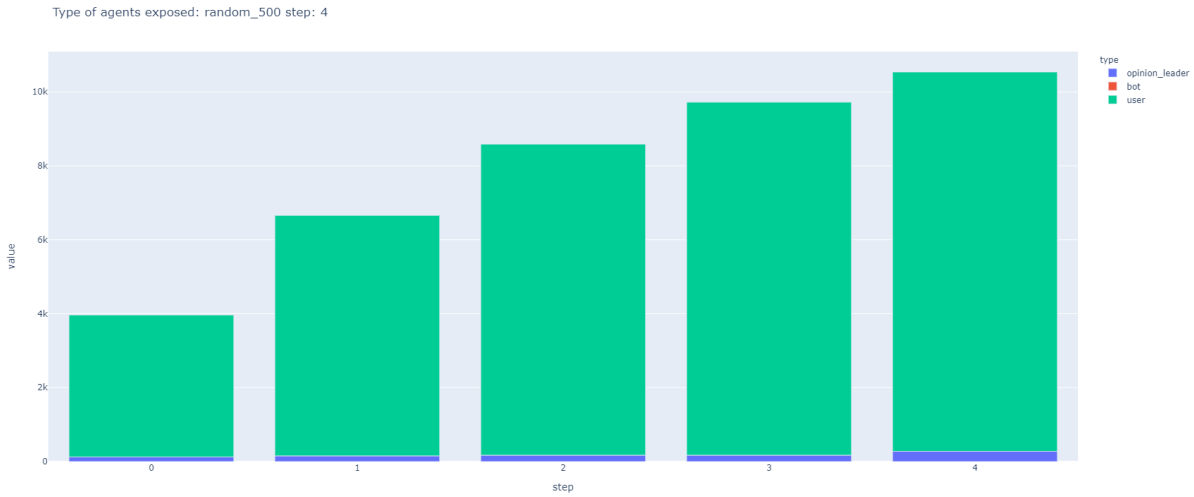

Figura 8: Exposed

	Step 0	Step 1	Step 2	Step 3	Step 4
Opinion Leader	278	456	600	791	957
Bot	1	1	1	0	5
User	3828	6512	8235	9311	9744

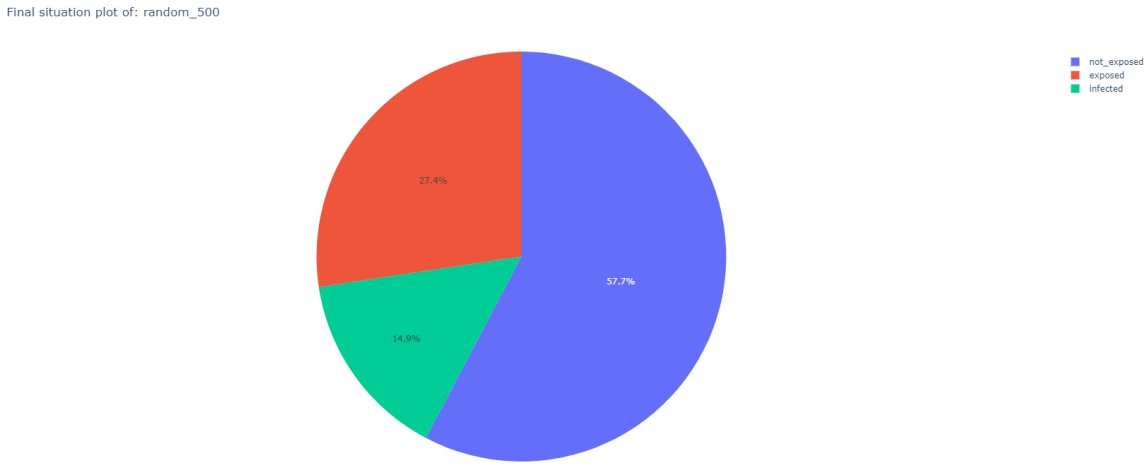

Figura 9: Final situation

Infetti diretti: 5686;
Infetti indiretti 48.

5.1.2 Grafo da 1000 followers

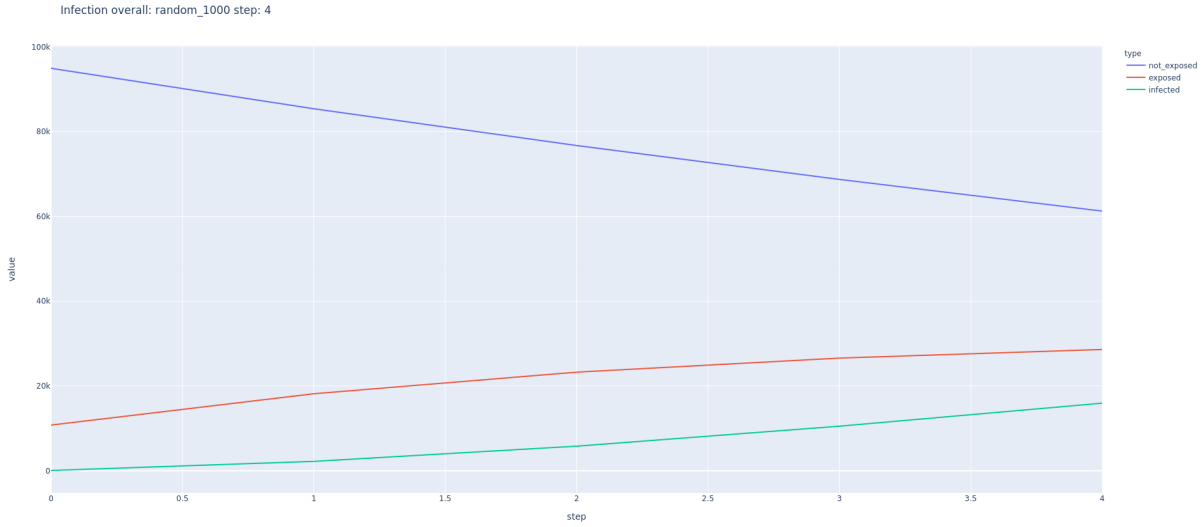

Figura 10: Infection overall

	Step 0	Step 1	Step 2	Step 3	Step 4
Not exposed	94933	85375	76715	68726	61240
Exposed	10783	18178	23246	26555	28607
Infected	58	2221	5813	10493	15927

Exposed:

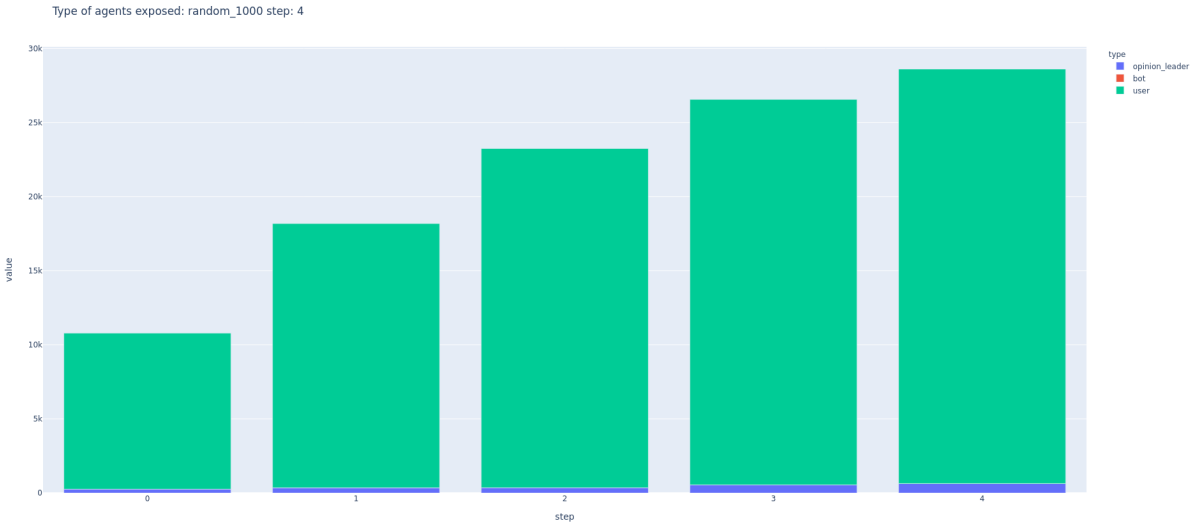

Figura 11: Exposed

	Step 0	Step 1	Step 2	Step 3	Step 4
Opinion Leader	259	352	355	547	631
Bot	0	0	0	0	0
User	10524	17826	22891	26008	27976

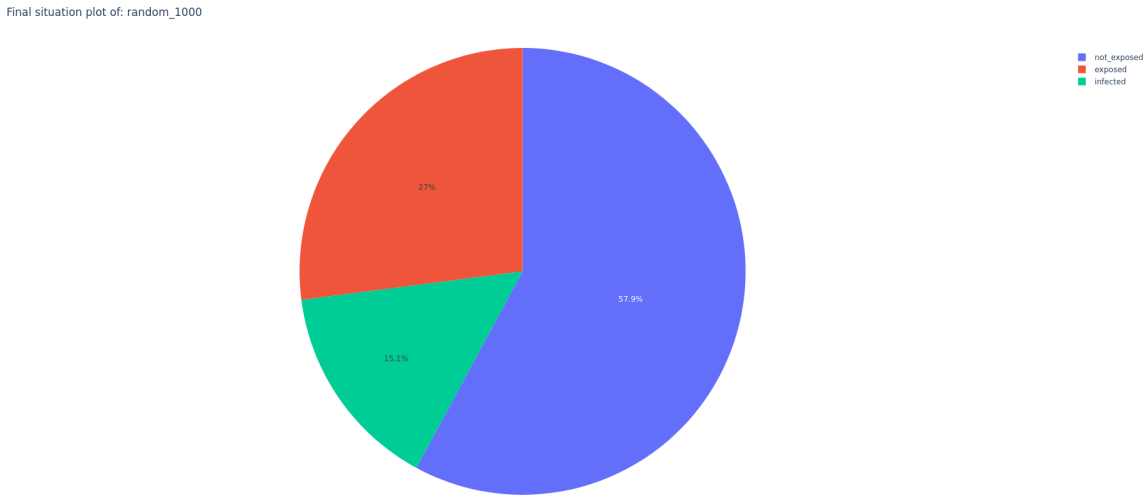

Figura 12: Final situation

Infetti diretti: 15656;
Infetti indiretti: 271.

5.1.3 Grafo da 1500 followers

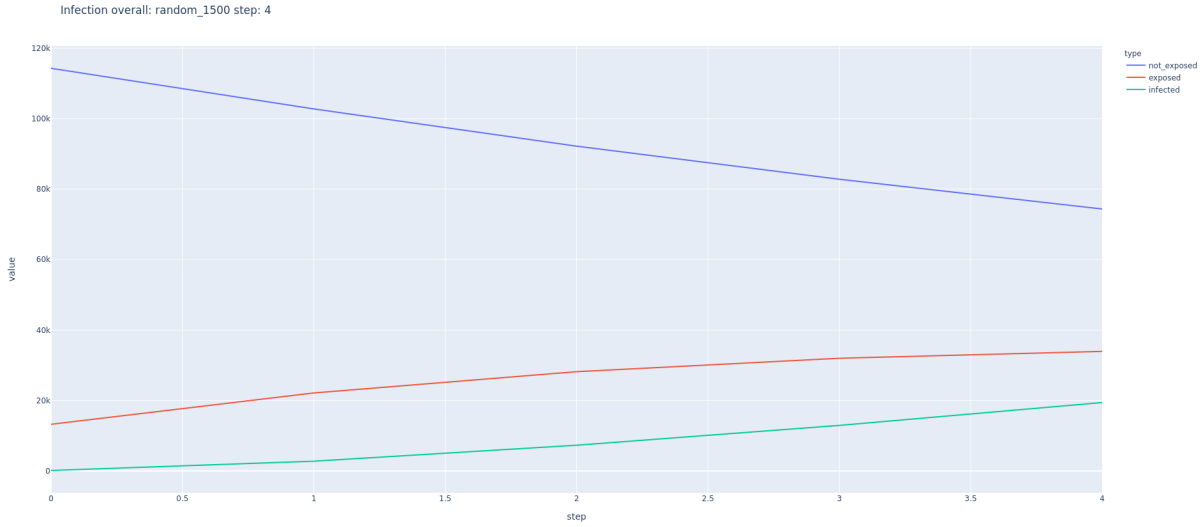

Figura 13: Infection overall

	Step 0	Step 1	Step 2	Step 3	Step 4
Not exposed	114324	102770	92206	82798	74386
Exposed	13261	22151	28184	31974	33914
Infected	115	2779	7310	12928	19400

Exposed:

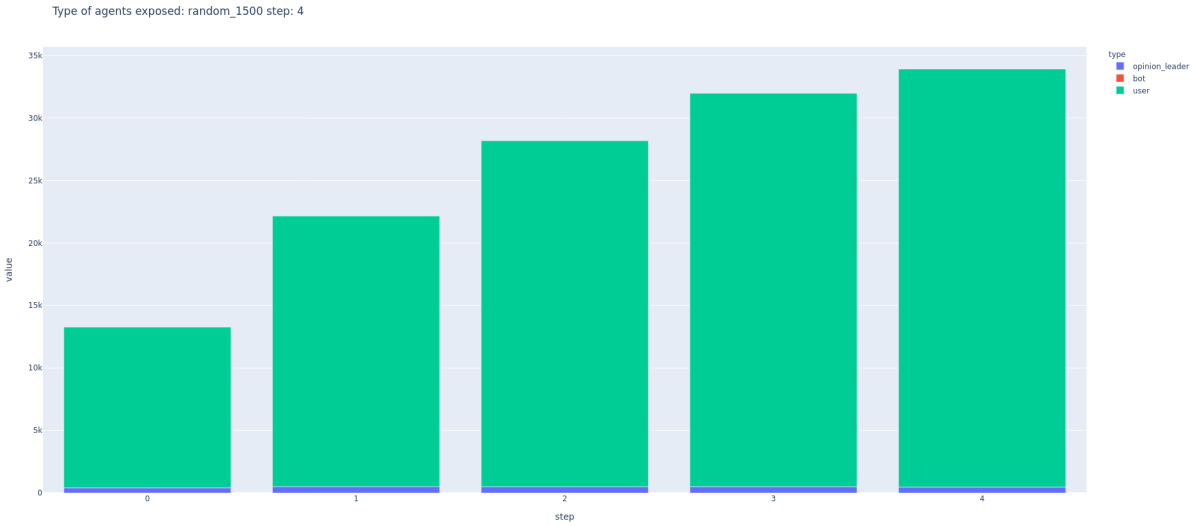

Figura 14: Exposed

	Step 0	Step 1	Step 2	Step 3	Step 4
Opinion Leader	418	516	502	510	471
Bot	0	0	0	0	0
User	12843	21635	27682	31464	33443

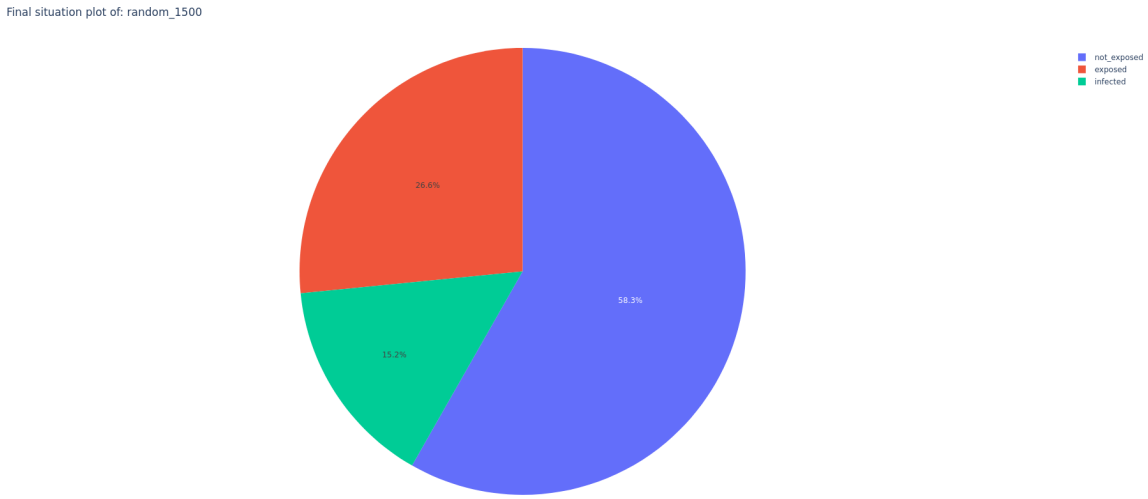

Figura 15: Final situation

Infetti diretti: 19319;
Infetti indiretti: 81.

5.1.4 Grafo da 2000 followers

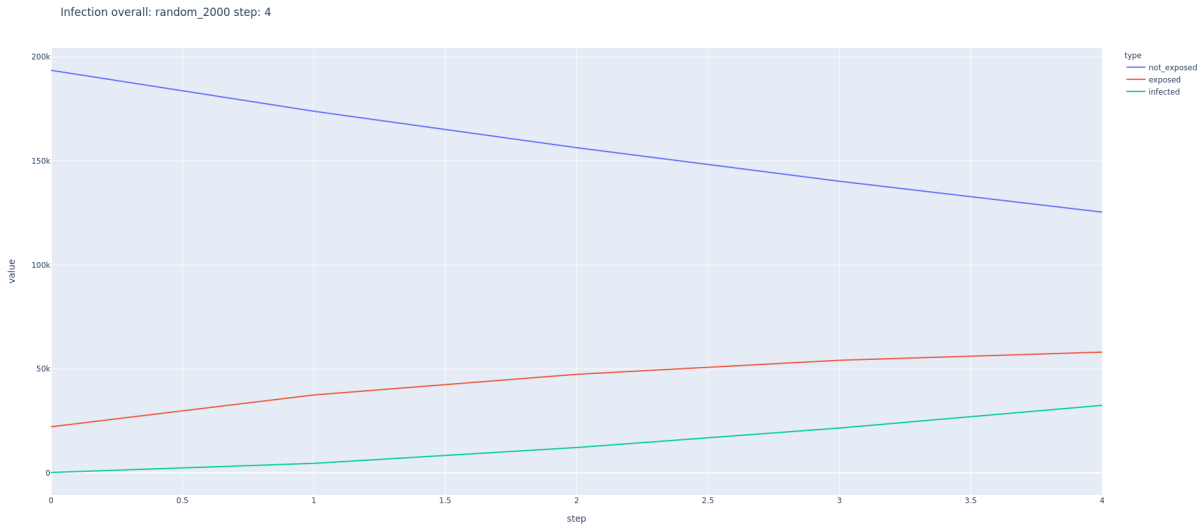

Figura 16: Infection overall

	Step 0	Step 1	Step 2	Step 3	Step 4
Not exposed	193592	173891	156381	140232	125366
Exposed	22155	37424	47383	54109	58054
Infected	163	4595	12146	21569	32490

Exposed:

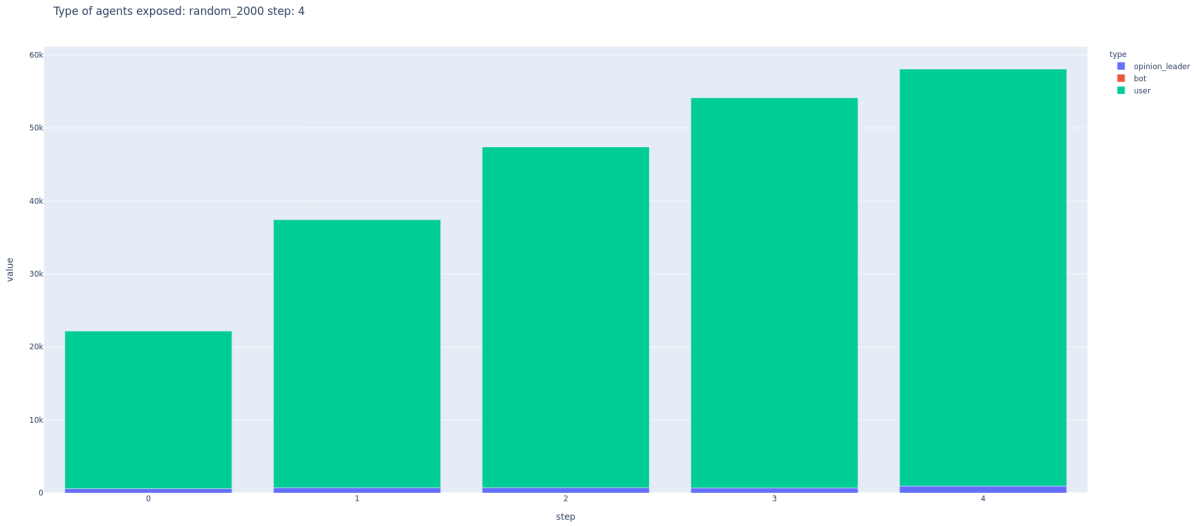

Figura 17: Exposed

	Step 0	Step 1	Step 2	Step 3	Step 4
Opinion Leader	600	726	731	707	964
Bot	0	0	0	0	0
User	21555	36698	46652	53402	57090

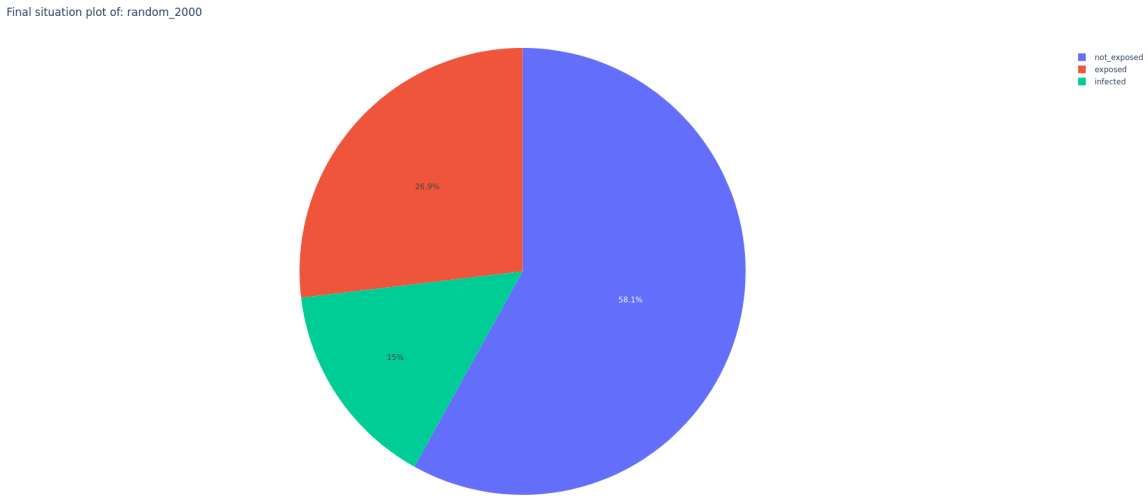

Figura 18: Final situation

Infetti diretti: 32216;
Infetti indiretti: 274.

Selezionando i Bot in maniera randomica questi hanno contribuito in maniera nulla alla propagazione della news all'interno del grafo, dove al termine della simulazione la percentuale di utenti non esposti supera significativamente quella degli utenti esposti o infetti. Questi risultati si sono ottenuti per tutte le grandezze del grafo.

5.2 Betweenness e In-Degree

In tutti i grafi considerati i nodi aventi maggiore Betweenness sono risultati essere coincidenti con quelli aventi maggiore In-Degree.

5.2.1 Grafo da 500 followers

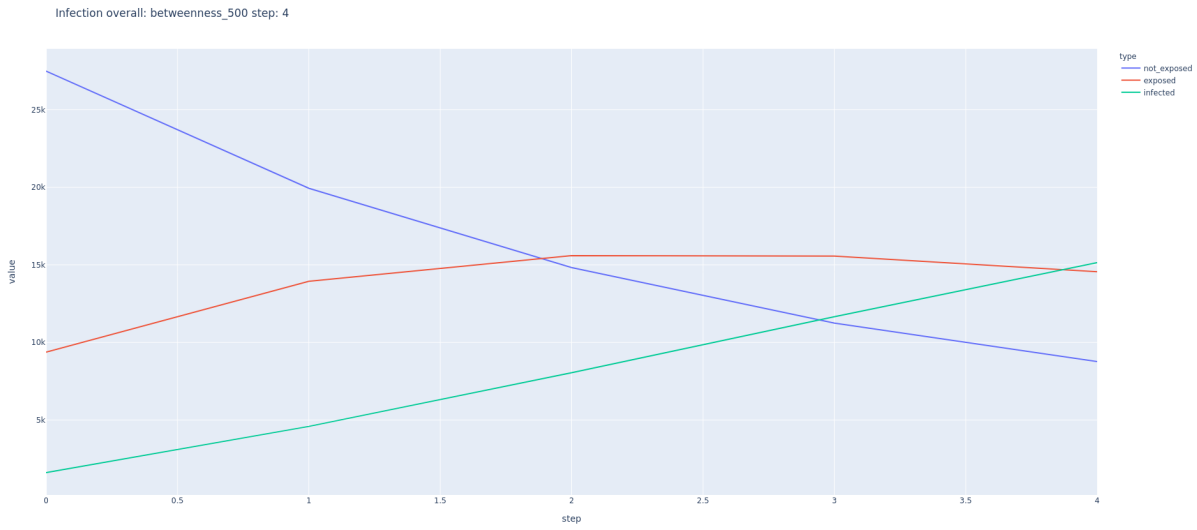

Figura 19: Infection overall

	Step 0	Step 1	Step 2	Step 3	Step 4
Not exposed	27493	19944	14833	11251	8769
Exposed	9380	13949	15600	15571	14564
Infected	1612	4592	8052	11663	15152

Exposed:

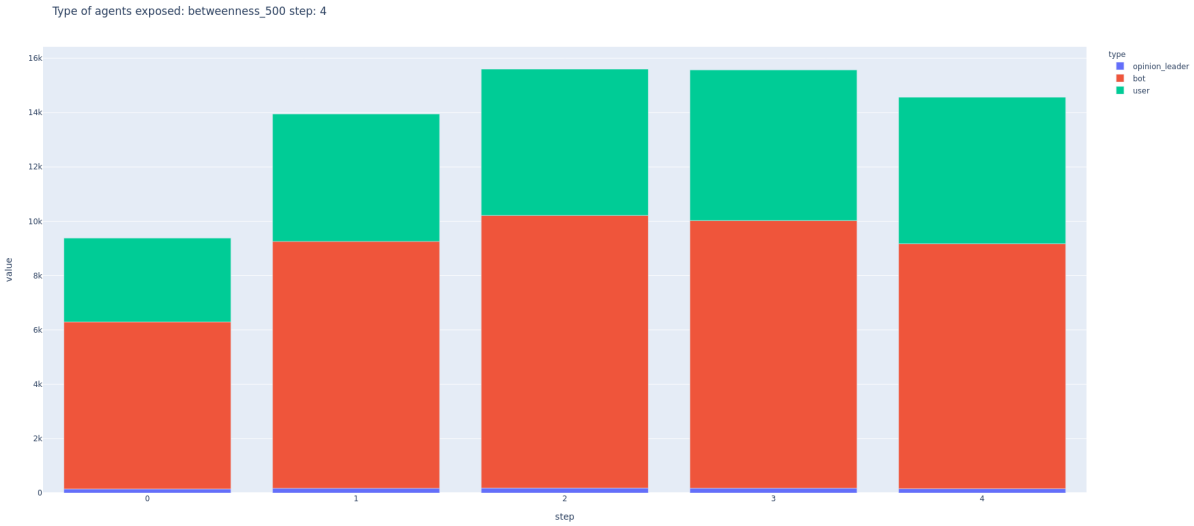

Figura 20: Exposed

	Step 0	Step 1	Step 2	Step 3	Step 4
Opinion Leader	144	172	180	174	156
Bot	6152	9086	10035	9852	9015
User	3084	4691	5385	5545	5393

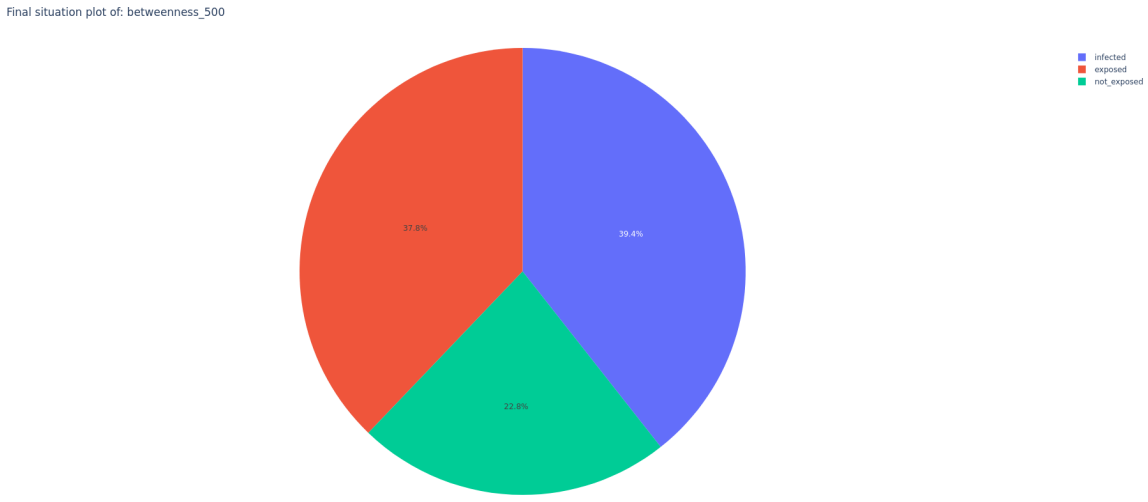

Figura 21: Final situation

Infetti diretti: 15143;
Infetti indiretti 9.

5.2.2 Grafo da 1000 followers

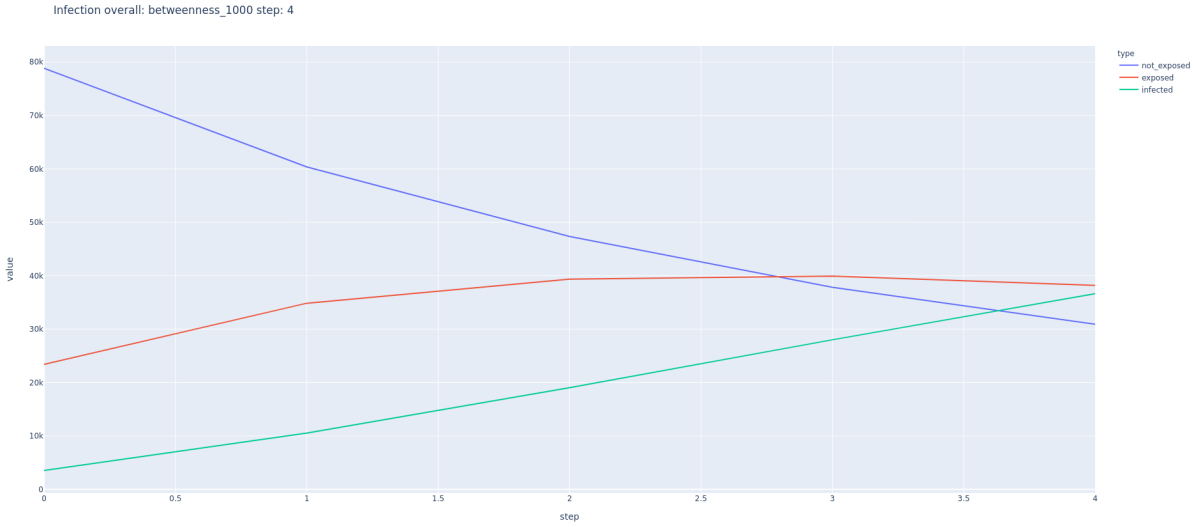

Figura 22: Infection overall

	Step 0	Step 1	Step 2	Step 3	Step 4
Not exposed	78842	60377	47361	37845	30926
Exposed	23402	34856	39375	39927	38198
Infected	3530	10541	19038	28002	36650

Exposed:

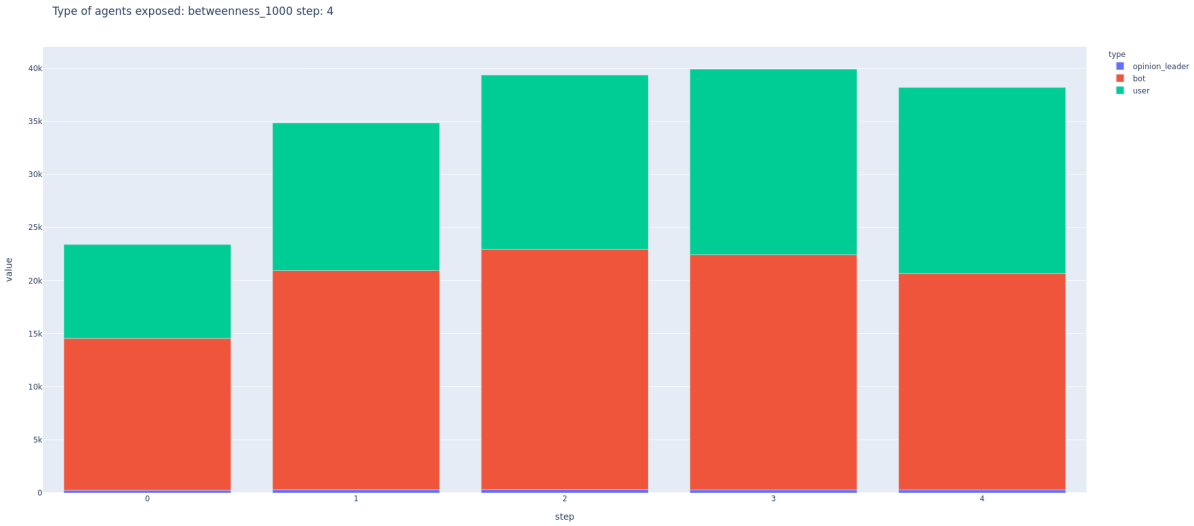

Figura 23: Exposed

	Step 0	Step 1	Step 2	Step 3	Step 4
Opinion Leader	250	311	326	303	300
Bot	14310	20642	22626	22142	20370
User	8842	13903	16423	17482	17528

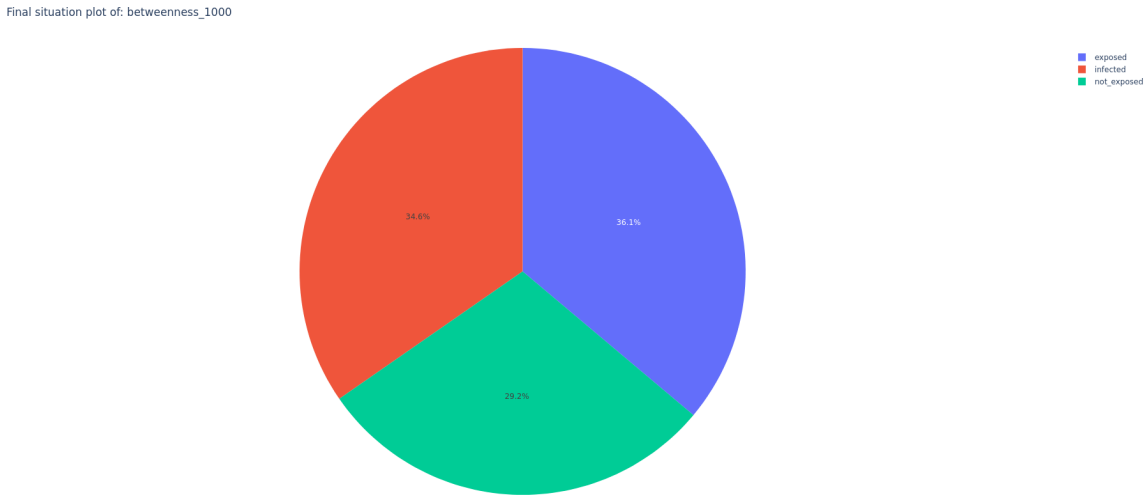

Figura 24: Final situation

Infetti diretti: 36597;
Infetti indiretti: 53.

5.2.3 Grafo da 1500 followers

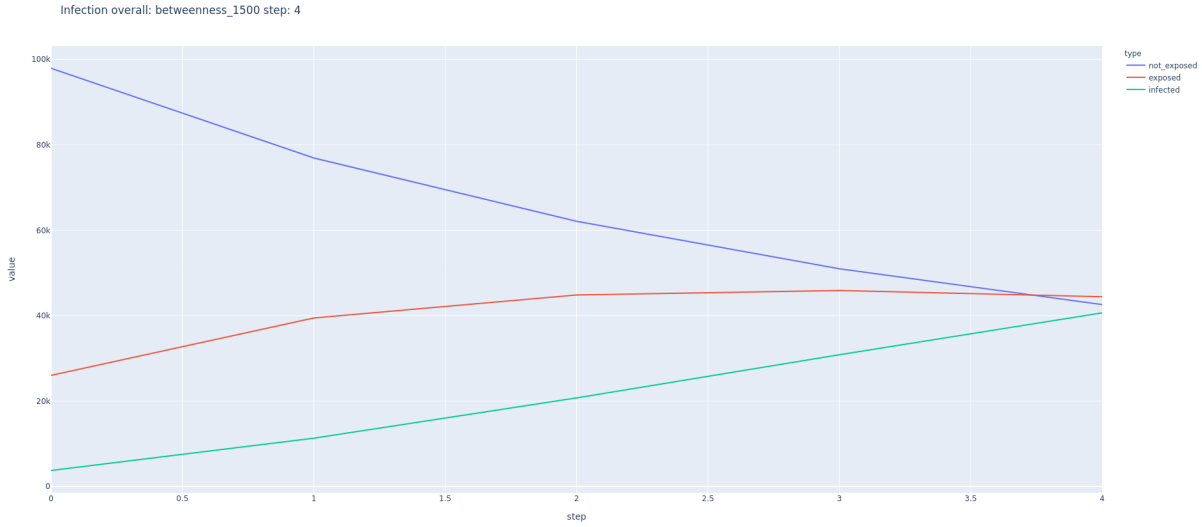

Figura 25: Infection overall

	Step 0	Step 1	Step 2	Step 3	Step 4
Not exposed	97952	76930	62122	50979	42608
Exposed	26024	39468	44830	45892	44446
Infected	3724	11302	20748	30829	40646

Exposed:

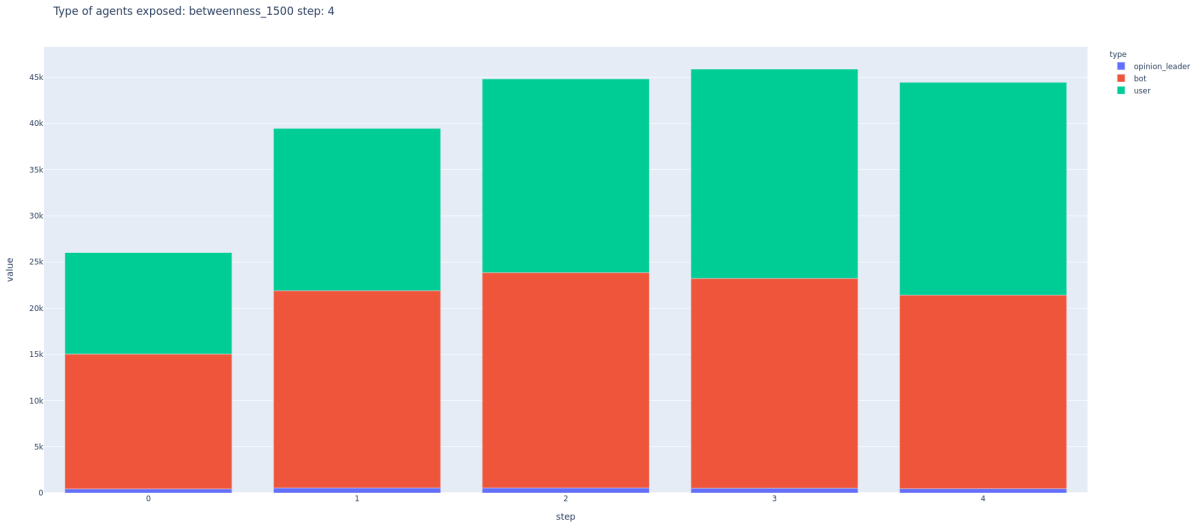

Figura 26: Exposed

	Step 0	Step 1	Step 2	Step 3	Step 4
Opinion Leader	433	545	545	512	469
Bot	14611	21355	23317	22720	20955
User	10980	17568	20968	22660	23022

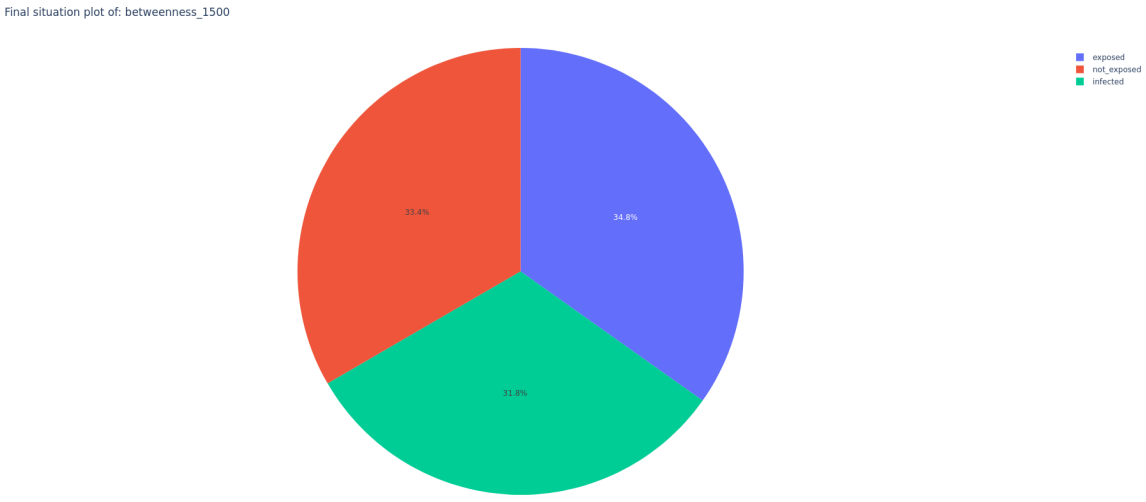

Figura 27: Final situation

Infetti diretti: 40559;
Infetti indiretti: 87.

Selezionando i Bot considerando la misura di centralità della Betweenness si nota che essi hanno una grande influenza all'interno della rete. La maggior parte dell'esposizione degli utenti è avvenuta tramite Bot e questo ha fatto sì che la percentuale di utenti infetti e esposti sia significativamente maggiore a quella degli utenti che non sono entrati in contatto con la news.
Questi risultati sono osservabili in tutte le dimensioni del grafo usate. L'eccessiva complessità computazionale richiesta nel calcolo della betweenness non ci ha permesso di effettuare simulazioni sul grafo dei 2000 followers.

5.3 Eigenvector

5.3.1 Grafo da 500 followers

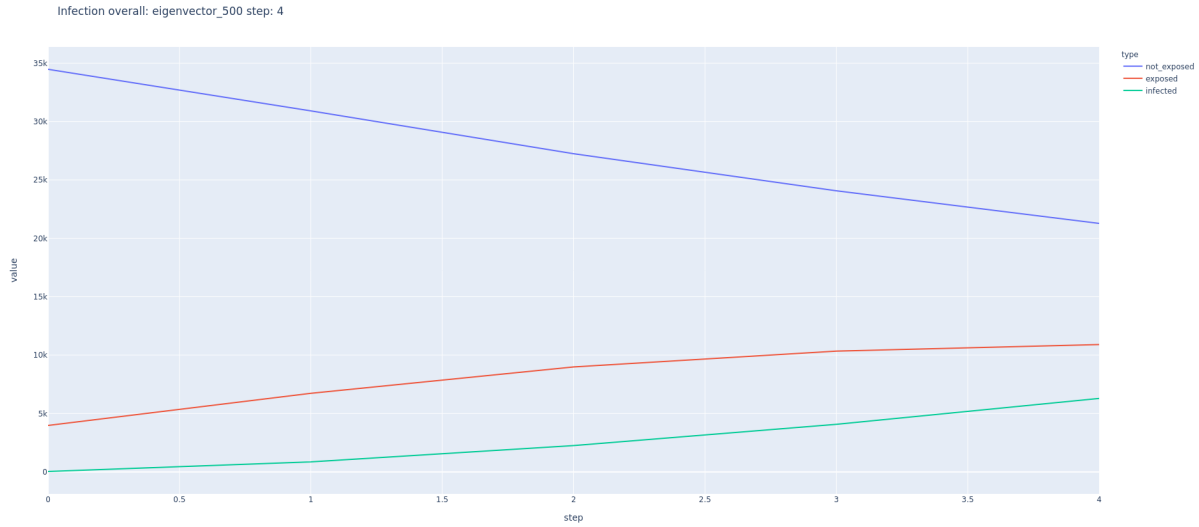

Figura 28: Infection overall

	Step 0	Step 1	Step 2	Step 3	Step 4
Not exposed	34472	30910	27239	24068	21281
Exposed	3980	6724	8989	10338	10908
Infected	33	851	2257	4079	6296

Exposed:

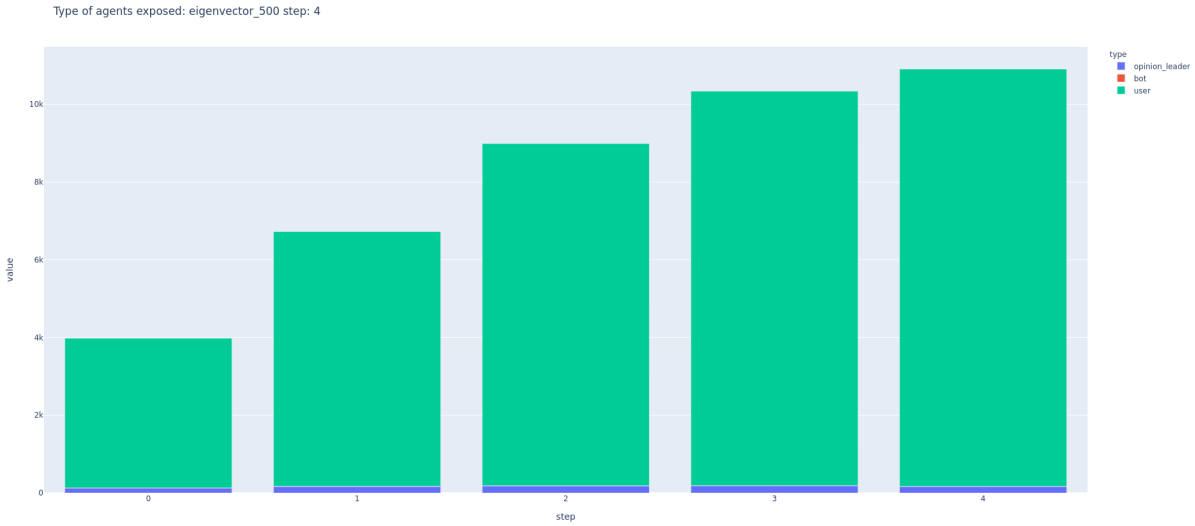

Figura 29: Exposed

	Step 0	Step 1	Step 2	Step 3	Step 4
Opinion Leader	123	165	179	183	165
Bot	10	10	9	8	8
User	3847	6549	8801	10147	10735

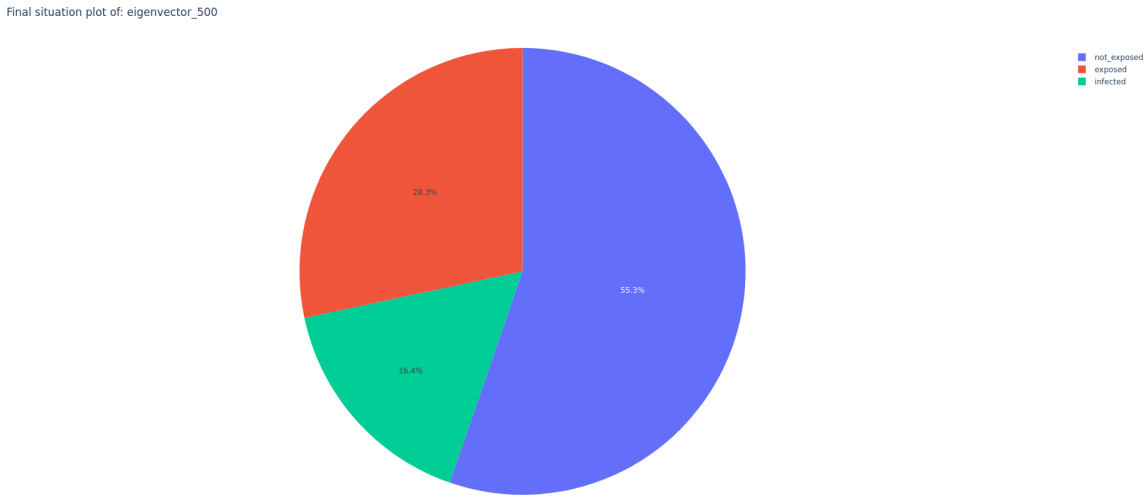

Figura 30: Final situation

Infetti diretti: 5807;
Infetti indiretti: 489.

5.3.2 Grafo da 1000 followers

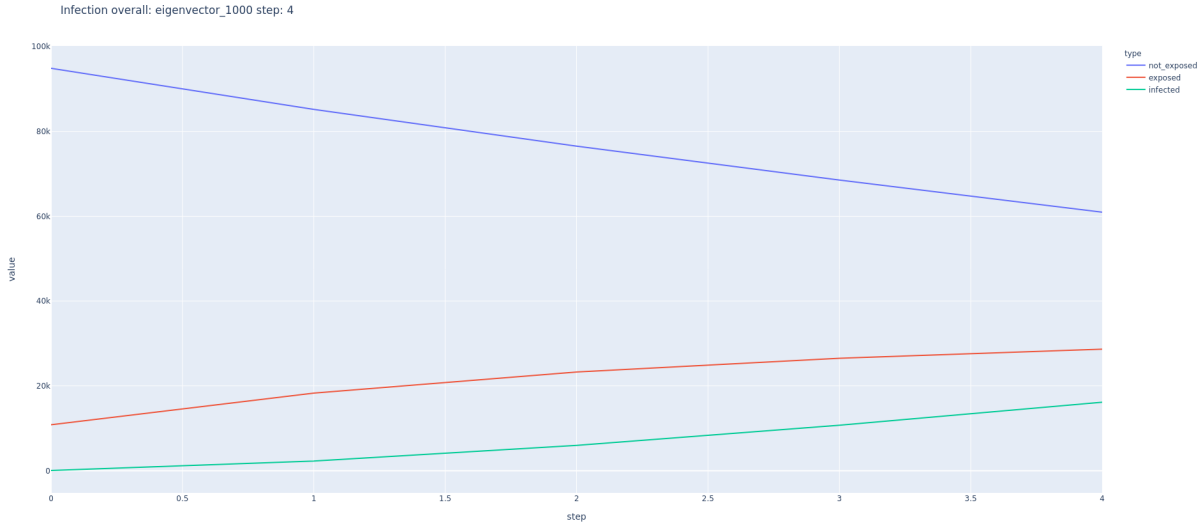

Figura 31: Infection overall

	Step 0	Step 1	Step 2	Step 3	Step 4
Not exposed	94882	85205	76507	68538	60957
Exposed	10841	18297	23274	26501	28671
Infected	51	2272	5993	10735	16146

Exposed:

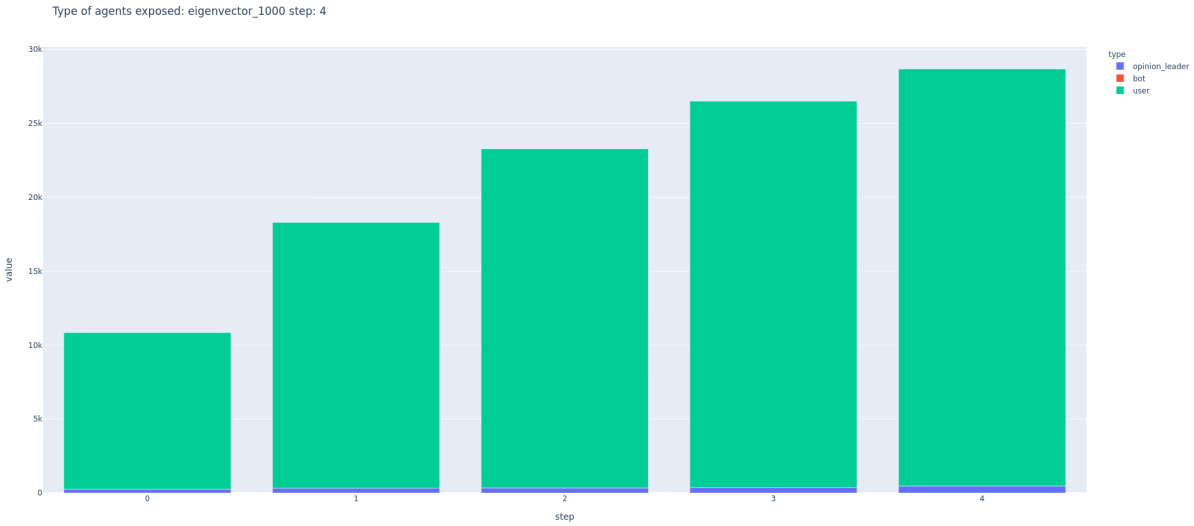

Figura 32: Exposed

	Step 0	Step 1	Step 2	Step 3	Step 4
Opinion Leader	259	321	346	363	473
Bot	2	5	4	4	7
User	10580	17971	22924	26134	28191

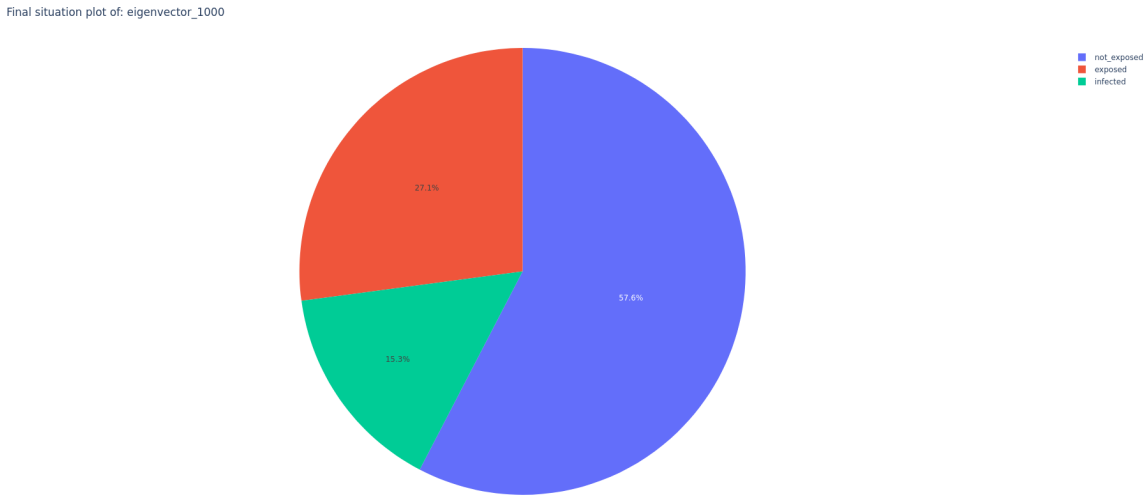

Figura 33: Final situation

Infetti diretti: 15911;
Infetti indiretti: 235.

5.3.3 Grafo da 1500 followers

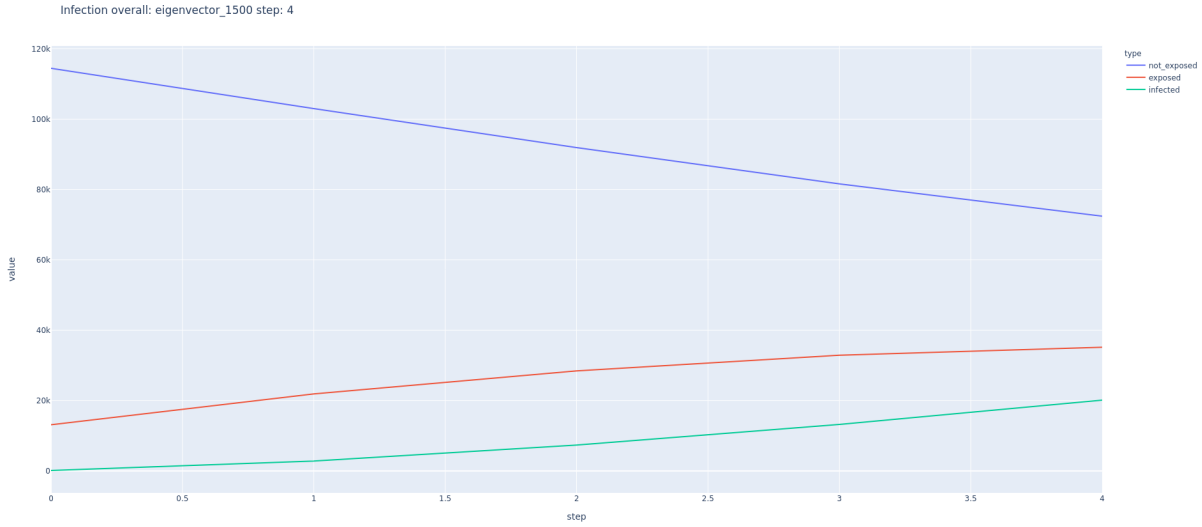

Figura 34: Infection overall

	Step 0	Step 1	Step 2	Step 3	Step 4
Not exposed	114474	103011	91937	81591	72417
Exposed	13113	21881	28436	32890	35178
Infected	113	2808	7327	13219	20105

Exposed:

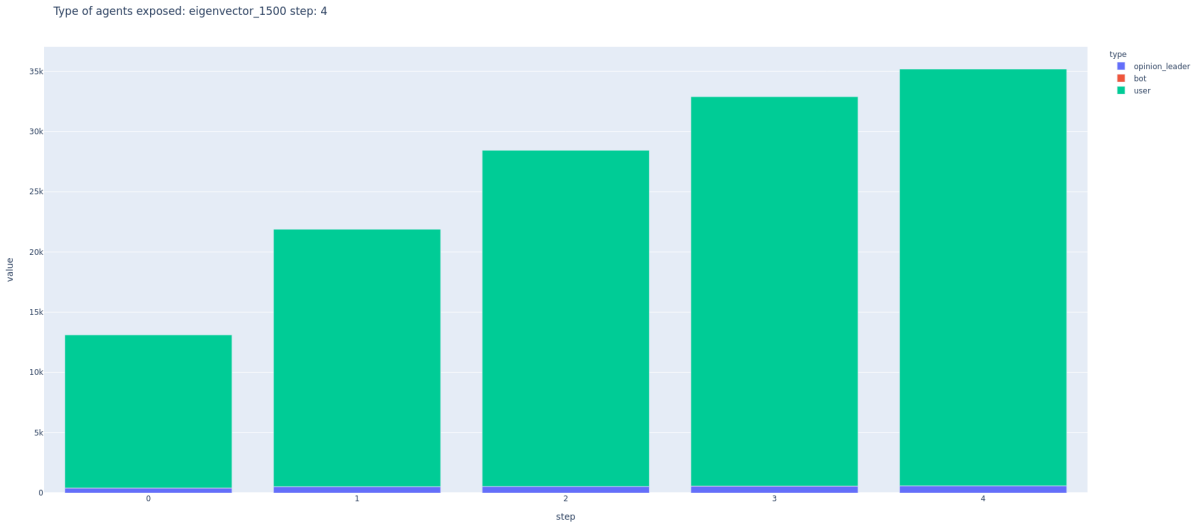

Figura 35: Exposed

	Step 0	Step 1	Step 2	Step 3	Step 4
Opinion Leader	415	526	542	571	592
Bot	4	3	7	8	7
User	12694	21352	27887	32311	34579

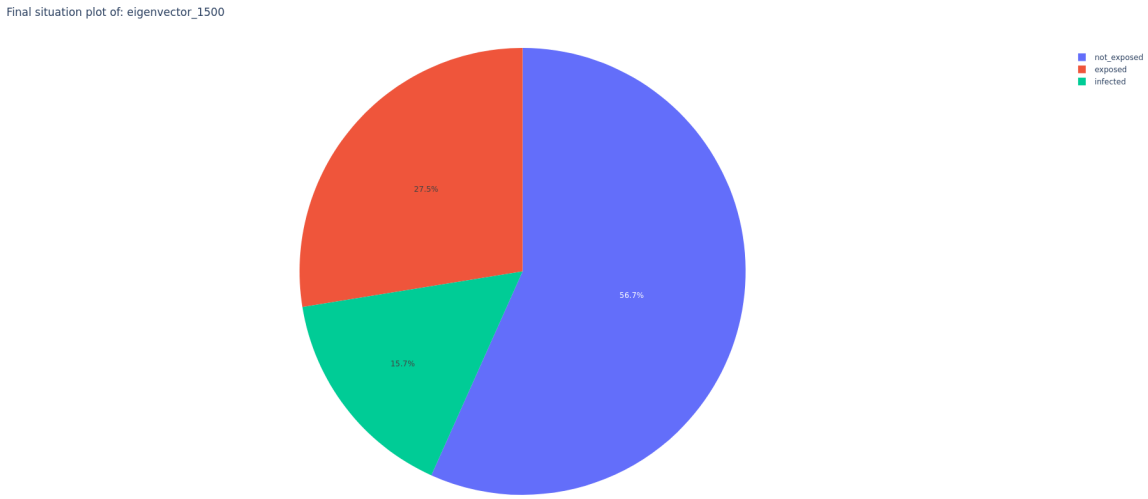

Figura 36: Final situation

Infetti diretti: 19136;
Infetti indiretti: 969.

5.3.4 Grafo da 2000 followers

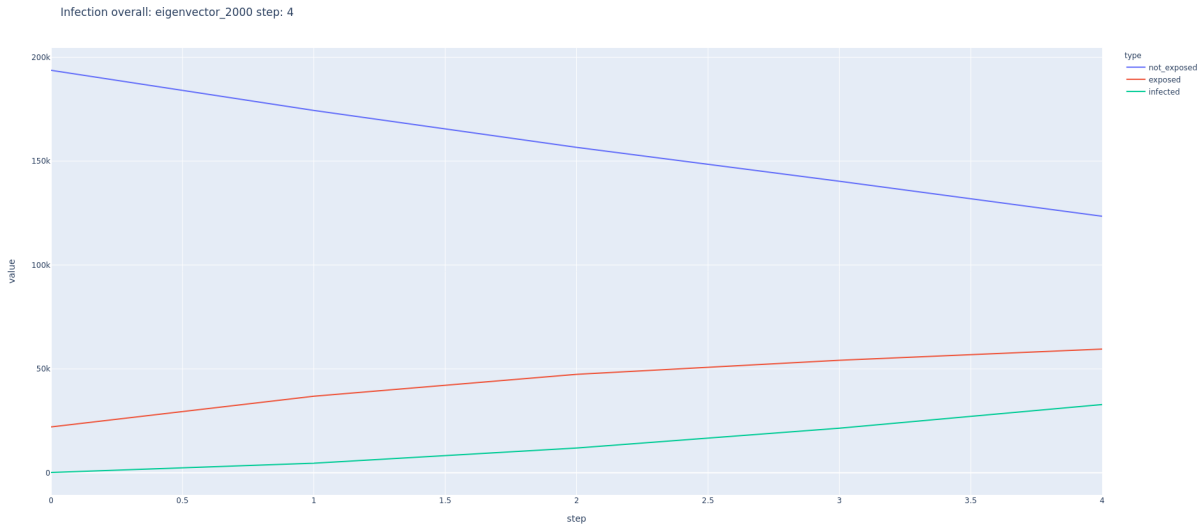

Figura 37: Infection overall

	Step 0	Step 1	Step 2	Step 3	Step 4
Not exposed	193700	174382	156612	140306	123490
Exposed	22060	36892	47361	54124	59550
Infected	150	4636	11937	21480	32870

Exposed:

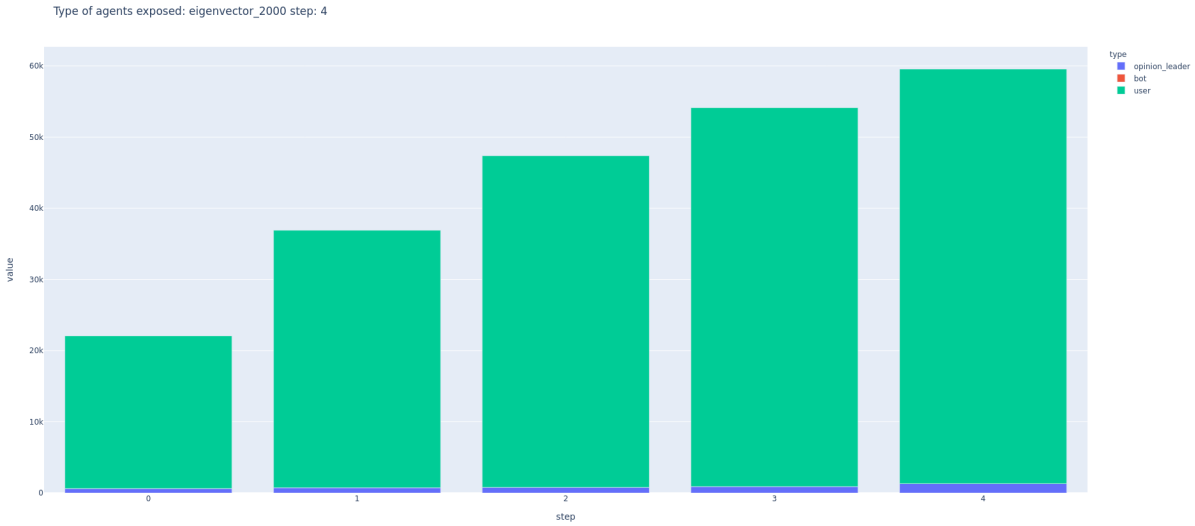

Figura 38: Exposed

	Step 0	Step 1	Step 2	Step 3	Step 4
Opinion Leader	618	736	792	898	1308
Bot	0	1	1	1	1
User	21442	36155	46568	53225	58241

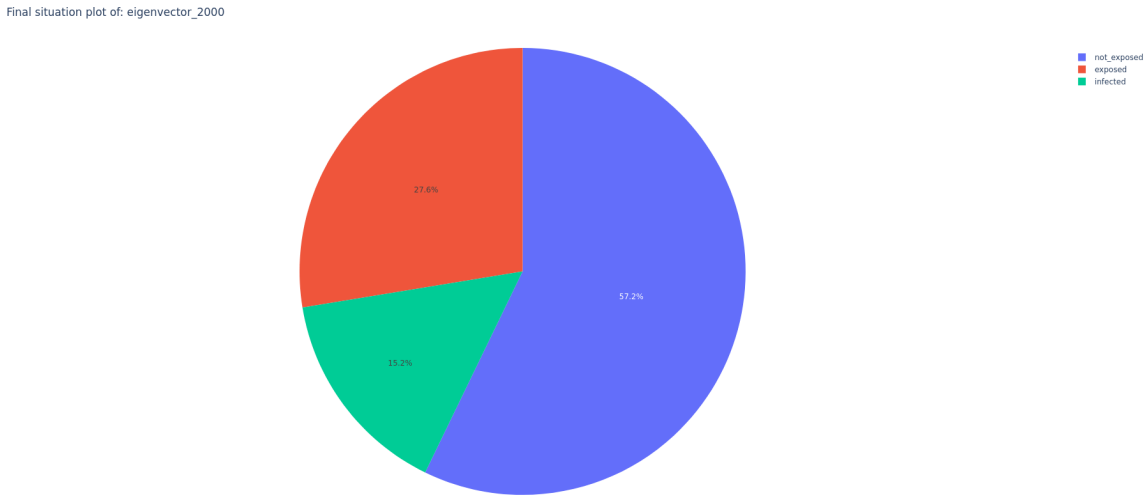

Figura 39: Final situation

Infetti diretti: 31995;
Infetti indiretti: 875.

Selezionando i Bot secondo la centralità basata su autovettori si hanno risultati simili a quelli ottenuti selezionando i Bot in maniera random, essi contribuiscono in modo minimo alla diffusione della news all'interno del grafo. Al termine della simulazione la percentuale di utenti non esposti supera quella degli utenti infetti o esposti.
Possiamo osservare gli stessi risultati in tutte le dimensioni del grafo considerate.

6 Conclusioni

La maggior diffusione della news all'interno del grafo si è ottenuta selezionando i Bot secondo la misura di centralità della Betweenness, è stata anche l'unica configurazione nella quale gli utenti infetti o esposti hanno superato il numero di utenti non esposti.
In tutte le simulazioni l'infezione è avvenuta per la maggior parte in maniera diretta (cioè dalla propagazione della notizia partita dall'opinion leader o da un bot).
Per quanto riguarda la fase di validazione non è stata possibile attuarla. A causa delle forti limitazioni nell'uso di Twitter dovute ad un restringimento delle policy di sicurezza e di accesso non siamo riusciti a validare il modello su casi reali. Le informazioni utilizzate all'interno delle analisi e la topologia della rete sono state ottenute effettuando uno scraping limitato proprio sui dati di twitter. Per la validazione del modello occorrerebbe avere a disposizione più dati e informazioni in modo da migliorare la fase di validazione e valutazione rispetto, ad esempio, a delle situazioni realistiche di diffusione e contaminazione di notizie che si sono verificate nel tempo.

7 Sviluppi futuri

Dopo aver costruito il simulatore e aver portato i risultati e le considerazioni derivanti dal processo di sperimentazione definiamo quelli che saranno gli sviluppi futuri del progetto.
A tal proposito vengono proposte due aree di miglioramento e di sviluppo: la prima legata all'applicazione e alla parte tecnica, mentre la seconda dedicata alla parte modellistica e progettuale.
Per quanto riguarda l'area applicativa:

- **Miglioramento della UX**: rendere migliorabile la user experience dell'applicazione web in modo da renderla facilmente fruibile e distribuibile ad una più larga utenza
- **Miglioramento delle performance**: è necessario incrementare le performance per quanto riguarda la velocità di computazione e di calcolo delle statistiche, modelli e grafici.
- **Deploy dell'applicazione**: rilascio dell'applicazione in un ambiente di produzione con sufficienti requisiti hardware per consentire e favorire numerosi simulazioni anche con più dimensionalità riducendo il limite infrastrutturale avuto durante la prima fase realizzativa
- **Aggiornamento del codice e della repository**: favorendo un approccio più OpenSource in modo che la soluzione sia più facilmente mantenibile e aggiornabile

- **Accesso ad hardware più performance**: in fase di valutazione per il lancio di simulazioni massive.

Per quanto riguarda invece la parte modellistica:

- **Arricchire la modellazione degli utenti**: Cercare di distinguere ulteriormente gli utenti in modo da definire delle categorie di utenza all'interno di una rete sociale in modo che siano configurabili e conseguentemente arricchire le categorie con attributi unici per utente come il genere, informazioni personali, ... oltre a rendere ancora più specifico l'inserimento dei bot all'interno della rete in modo tale da definire ad esempio la frequenza di pubblicazione, i tipi di pubblicazioni o altro... Allo stesso tempo anche gli Opinion Leader e le conseguenti sottoreti di cui fanno parte possono essere modellati in modo da rispecchiare ancora di più un caso reale.

- **Ottenere riscontro da dati reali**: A causa delle forti limitazioni nell'uso dei social network dovuto ad un restringimento dello politiche di sicurezza e privacy abbiamo riscontrato moltissime difficoltà nella valutazione e misurazione delle performance rispetto ad un caso reale. Avere a disposizione dei dati reali di alcune situazioni verificatesi in passato o ad una configurazione della rete più realistica ci consentirebbe di potenziare e migliorare la fase di valutazione e validazione dei risultati rispetto al nostro modello implementato.

- **Caratterizzazione dei contenuti**: All'interno del modello e del sistema multi-agente non abbiamo effettuato una distinzione del tipo di contenuti diffusi, la possibilità di caratterizzare i contenuti ad esempio con dei topics consentirebbe di definire e modellare comportamenti più simili alla realtà andando a studiare anche a livello contenutistico la diffusione dei post all'interno della rete.

- **Informazioni di contesto**: Rispetto ai contenuti e agli utenti, anche alcune informazioni di contesto potrebbero migliorare e rendere il sistema più simile alla realtà. Queste informazioni potrebbero essere ad esempio: tempi di pubblicazione delle news (a livello orario), informazioni geospaziali come luoghi o parti del mondo, tempi e facilità di accesso alla piattaforma dove gli utenti interagiscono, ecc...

- **Test con nuovi algoritmi**: implementare ulteriori algoritmi epidemiologici per la diffusione di notizie per identificare il migliore modello da realizzare all'interno del sistema.

- **Simulatore personalizzato**: costruzione di un simulatore personalizzato per questo specifico task in modo da potenziare e rendere più specifico l'impiego di un sistema multi agente in questo determinato contesto, ma allo stesso tempo più personalizzabile e ottimizzato

Riferimenti bibliografici

[1] A. L. Barabasi. *Network Science*. Cambidge university press, 2016.

[2] Cormen T. H. Leiserson C. E. Rivest R. L. & Stein C. *Introduction to algorithms second edition*. McGraw-Hil, 2001.

[3] Lipponen L. et al. de Laat M., Lally V. Investigating patterns of interaction in networked learning and computer-supported collaborative learning: A role for social network analysis. *Computer Supported Learning*, 2:87–103, 2007.

[4] Lynne Hamill & Nigel Gilbert. Simulating large social networks in agent-based models: A social circle model. *E:CO*, 12(4):XXX–XXX, 2010.

[5] Fang Jin, Edward R. Dougherty, Parang Saraf, Yang Cao, and Naren Ramakrishnan. Epidemiological modeling of news and rumors on twitter. In *Proceedings of the 7th Workshop on Social Network Mining and Analysis, SNAKDD 2013, Chicago, IL, USA, August 11, 2013*, pages 8:1–8:9. ACM, 2013. URL: https://www.researchgate.net/publication/262354540_Epidemiological_Modeling_of_News_and_Rumors_on_Twitter.

[6] Wasserman Stanley; Fauset Kathering. Social network analysis in the social and behavioral sciences. *Cambridge University Press*, pages 1–27, 1994.

[7] Mediakix. How much time is spent on social media lifetime, 2020. URL: https://mediakix.com/blog/how-much-time-is-spent-on-social-media-lifetime/.

[8] Eduardo Merino, Jesús M. Sánchez, David García, J. Fernando Sánchez-Rada, and Carlos Angel Iglesias. Modeling social influence in social networks with soil, a python agent-based social simulator. In *Advances in Practical Applications of Cyber-Physical Multi-Agent Systems: The PAAMS Collection - 15th International Conference, PAAMS 2017, Porto, Portugal, June 21-23, 2017, Proceedings*, volume 10349 of *Lecture Notes in Computer Science*, pages 337–341. Springer, 2017. URL: https://www.researchgate.net/publication/318144973_Modeling_Social_Influence_in_Social_Networks_with_SOIL_a_Python_Agent-Based_Social_Simulator.

[9] Pew Research. Activism in the social media age. 11 Luglio 2018, 2018. URL: https://www.pewresearch.org/internet/2018/07/11/public-attitudes-toward-political-engagement-on-social-media/.

[10] Pew Research. Facebook is the top social media site for news in western europe. 8 maggio 2018, 2018. URL: https://www.journalism.org/2018/05/14/many-western-europeans-get-news-via-social-media-but-in-some-countries-substantial-minori
pj_2018-05-14_western-europe_5-02/.

[11] Howard Rheingold. *The Virtual Community*. Homesteading on the Electronic Frontier, Addison-Wesley, 1993.

[12] Peter Norvig Russel Stuart J. *Artificial Intelligence: A Modern Approach*. Englewood Cliffs, N.J: Prentice Hall, 1995.

[13] Jesús M. Sánchez, Carlos A. Iglesias, and J. Fernando Sánchez-Rada. *Soil: An Agent-Based Social Simulator in Python for Modelling and Simulation of Social Networks*, volume 10349 of *LNAI*, pages 234–245. Springer Verlag, 2017. URL: https://www.researchgate.net/publication/318144945_Soil_An_Agent-Based_Social_Simulator_in_Python_for_Modelling_and_Simulation_of_Social_Networks.

[14] Emilio Serrano, Carlos Angel Iglesias, and Mercedes Garijo. A survey of twitter rumor spreading simulations. In *Computational Collective Intelligence - 7th International Conference, ICCCI 2015, Madrid, Spain, September 21-23, 2015. Proceedings, Part I*, volume 9329, pages 113–122, 2015. URL: https://www.researchgate.net/publication/300349587_A_Survey_of_Twitter_Rumor_Spreading_Simulations.

[15] Misako Takayasu, Kazuya Sato, Yukie Sano, Kenta Yamada, Wataru Miura, and Hideki Takayasu. Rumor diffusion and convergence during the 3.11 earthquake: a twitter case study. *PLoS one*, 10(4):e0121443, 2015. URL: https://www.researchgate.net/publication/273124011_Rumor_Diffusion_and_Convergence_during_the_311_Earthquake_A_Twitter_Case_Study.

[16] twintproject. Twint. URL: https://github.com/twintproject/twint.

[17] Twitter. Account ufficiale twitter di giuseppe conte. URL: https://twitter.com/GiuseppeConteIT.

[18] D.J Watts. *Six degrees: the science of a connected age*. Norton, 2003.

[19] Wikipedia. Comunità. URL: https://it.wikipedia.org/wiki/Comunit\unhbox\voidb@x\bgroup\let\unhbox\voidb@x\setbox\@tempboxa\hbox{a\global\mathchardef\accent@spacefactor\spacefactor}\let\begingroup\def{}\endgroup\relax\let\ignorespaces\relax\accent18a\egroup\spacefactor\accent@spacefactor?oldformat=true.

[20] Wikipedia. Comunità virtuale. URL: https://it.wikipedia.org/wiki/Comunit\unhbox\voidb@x\bgroup\let\unhbox\voidb@x\setbox\@tempboxa\hbox{a\global\mathchardef\accent@spacefactor\spacefactor}\let\begingroup\def{}\endgroup\relax\let\ignorespaces\relax\accent18a\egroup\spacefactor\accent@spacefactor_virtuale?oldformat=true.

[21] Wikipedia. Connettività in un grafo. URL: https://en.wikipedia.org/wiki/Connectivity_(graph_theory)?oldformat=true.

[22] Wikipedia. Distanza in un grafo. URL: https://en.wikipedia.org/wiki/Distance_(graph_theory)?oldformat=true.

[23] Wikipedia. Misure betweennes. URL: https://en.wikipedia.org/wiki/Betweenness_centrality?oldformat=true.

[24] Wikipedia. Misure di centralità di un grafo. URL: https://en.wikipedia.org/wiki/Centrality?oldformat=true.

[25] Wikipedia. Ordine di un grafo. URL: https://it.wikipedia.org/wiki/Grafo?oldformat=true.

[26] Wikipedia. Rete ad invarianza di scala. URL: https://it.wikipedia.org/wiki/Rete_a_invarianza_di_scala?oldformat=true.

[27] Wikipedia. Senso di comunità. URL: https://en.wikipedia.org/wiki/Sense_of_community?oldformat=true.

[28] Wikipedia. Teoria del mondo piccolo. URL: https://it.wikipedia.org/wiki/Teoria_del_mondo_piccolo?oldformat=true.

[29] Samuel C. Woolley. Automating power: Social bot interference in global politics. 10 Marzo 2016, 2016. URL: https://firstmonday.org/ojs/index.php/fm/article/view/6161.

[30] Tauhid R Zaman, Ralf Herbrich, Jurgen Van Gael, and David Stern. Predicting information spreading in twitter. In *Workshop on computational social science and the wisdom of crowds, nips*, volume 104, pages 17599–601, 2010. URL: https://www.researchgate.net/publication/228672351_Predicting_Information_Spreading_in_Twitter.