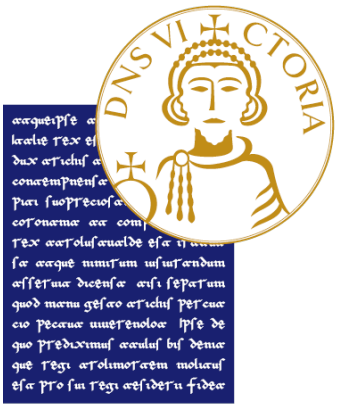
UNIVERSITÀ DEGLI STUDI DEL SANNIO



**Dipartimento di Ingegneria**

*Corso di laurea magistrale in Ingegneria Informatica*

# Valutazione di tecniche basate su grafi per l'analisi di fake news e metriche di propagazione

Relatori

Prof. Antonio Pecchia

Prof. Francesco Vasca

Candidato

Francesco Pio Briuolo 399000544

Anno Accademico 2022/2023

# Abstract

Al giorno d'oggi le persone fanno affidamento sui social media per consultare e condividere informazioni, soprattutto grazie alla rapidità e all'ampiezza con cui le notizie possono diffondersi attraverso questi canali, piuttosto che sui media tradizionali. Sebbene questi mezzi di comunicazione denominati "new media" abbiano rivoluzionato l'accesso e la condivisione delle informazioni, hanno anche facilitato la propagazione di contenuti falsi o fuorvianti. Questo fenomeno prende il nome di *fake news* e porta dietro di sè potenziali conseguenze dannose per la società.

L'obiettivo di questa tesi di ricerca è fornire uno studio approfondito della letteratura e di alcune tecniche specifiche per analizzare la *propagazione* delle *fake news*, al fine di supportare l'attività di *detection* delle stesse.

Di seguito è riportata l'organizzazione del lavoro. Il capitolo I delinea il problema della diffusione delle fake news, evidenziando come la natura virale dei social media amplifichi la portata delle informazioni false. Si offre poi una panoramica sugli strumenti più validi utilizzati per affrontare questo problema, ovvero l'Intelligenza Artificiale e, più nello specifico, il *Machine Learning* e le *reti neurali*. In questo capitolo vengono anche presentati gli studi condotti sui lavori presenti in letteratura, evidenziando l'importanza

di adottare tecniche basate su grafi per analizzare la propagazione delle fake news, soprattutto in ottica di rilevamento delle stesse. Un altro aspetto approfondito in questo capitolo riguarda l'analisi dei dataset più utilizzati, evidenziando le criticità che caratterizzano l'argomento. Nei successivi due capitoli vengono presentate le metodologie utilizzate per realizzare i lavori prodotti in questa tesi. Il capitolo II è dedicato alla realizzazione di un grafo bipartito al fine di estrarre informazioni utili per la rilevazione delle fake news utilizzando dei dati estratti, tramite web scraping, direttamente dal sito di fact-checking PolitiFact. Per fare ciò è stata utilizzata la *teoria dei grafi*. II capitolo III, invece, introduce lo studio di una rete di propagazione gerarchica per il rilevamento delle fake news, condotto a partire da un lavoro già esistente [18] in cui vengono utilizzati algoritmi di Machine Learning per la rilevazione delle fake news. In questo caso, il dataset di riferimento è FakeNewsNet [17] e l'obiettivo è quello di utilizzare le caratteristiche della rete delineate dal lavoro originale per effettuare previsioni sulle notizie nel social network X. Il contributo della tesi, in questo senso, è quello di creare una rete neurale Feedforward per ottenere dei risultati aggiuntivi e confrontarli con quelli ottenuti dagli altri algoritmi. Nel capitolo IV vengono presentati e discussi i risultati ottenuti tramite i lavori sperimentali condotti. Infine, nel capitolo V vengono presentate le sfide ancora aperte di fronte ai problemi della fake news detection e della loro diffusione, oltre a dei possibili sviluppi futuri che potrebbero sfruttare lo studio presentato per implementare soluzioni interessanti.

# Indice

# Capitolo 1

# Introduzione e Stato dell'arte

L'informazione riveste un ruolo cruciale nella società moderna, influenzando ogni aspetto della vita quotidiana, dalle decisioni personali alle dinamiche globali. Essa è fondamentale per consentire alle persone di prendere decisioni consapevoli e l'accesso a dati accurati e tempestivi può fare la differenza in situazioni di emergenza, come durante pandemie o disastri naturali. Le metodologie che caratterizzano la diffusione delle notizie hanno subito trasformazioni radicali nel corso dei secoli e questa evoluzione ha segnato profondamente il modo in cui le persone interagiscono e si relazionano tra loro. È proprio nell'era digitale che si è osservata la proliferazione dei canali di accesso all'informazione, grazie all'avvento di Internet e dei social media, come YouTube, Facebook, X (in origine Twitter). Tutto ciò ha consentito, rispetto ai media tradizionali come i giornali o la televisione, di amplificare la portata e la velocità con cui le notizie vengono raccolte, distribuite e consumate. Ma ciò rappresenta un'arma a doppio taglio, dal momento che le informazioni che circolano sul web, e in particolare sui social media, non sono

tutte veritiere ed è sempre più facile imbattersi nelle cosiddette *fake news*. Questo fenomeno esiste da tantissimo tempo, ma al giorno d'oggi la diffusione di informazioni false o fuorvianti al fine di ingannare, manipolare o influenzare l'opinione pubblica è in forte crescita, soprattutto da quando i social media rappresentano il mezzo principale per la diffusione delle notizie. Ci si trova dunque in un ambiente in cui le notizie, vere o false, possono diventare virali in pochi istanti. Proprio per la facilità e il basso costo che richiede divulgare informazioni sui social media, molte persone o organizzazioni sfruttano strumenti generativi per la produzione di notizie false per ottenere dei guadagni, ad esempio in termini finanziari o politici [27]. In questo contesto, dunque, le fake news hanno assunto un ruolo preoccupante, minando la fiducia nell'informazione e compromettendo la capacità degli utenti di discernere la verità. È stato dimostrato che le fake news influenzano il modo in cui le persone interpretano e rispondono alle notizie reali [19]. Pertanto, se le notizie ricevute dai social media sono veritiere, esse rappresentano un grande vantaggio, ma al contrario, se queste sono false, possono avere conseguenze dannose sulla società, sulla politica, sull'economia e persino sulla salute pubblica. L'entità del danno legato alla diffusione delle fake news è incalcolabile, motivo per cui il rilevamento tempestivo e accurato delle fake news è diventato cruciale per contrastare questa epidemia di disinformazione. Tuttavia, identificare e neutralizzare le informazioni false sui social media è una sfida complessa, che richiede approcci innovativi e strumenti avanzati. La rilevazione delle fake news presenta caratteristiche uniche, che rendono inefficaci o inadatti gli algoritmi per la detection usati nei media tradizionali. Analizzare come le notizie si diffondono attraverso reti sociali e altri canali può fornire preziose

informazioni sulla loro origine, la loro diffusione e il loro impatto.

Per comprendere meglio la gravità del fenomeno e dei potenziali danni che le fake news possono arrecare alla società, è utile comprendere in che modo una notizia falsa divulgata sui social media possa generare conseguenze disastrose nella vita reale. Uno dei casi più famosi è sicuramente quello del "*Pizzagate*", relativo a uno scandalo emerso durante le elezioni presidenziali degli Stati Uniti del 2016. Questa teoria sosteneva, senza alcuna prova, che una rete di pedofilia legata al Partito Democratico operasse da una pizzeria chiamata Comet Ping Pong a Washington, D.C. La teoria del complotto ha avuto origine da email del presidente della campagna elettorale di Hillary Clinton, John Podesta, che sono state hackerate e pubblicate da WikiLeaks. Gli adepti di questa teoria hanno interpretato erroneamente alcune parole nei messaggi come riferimenti a un traffico sessuale di bambini. Il proprietario James Alefantis e il personale del ristorante hanno anche ricevuto minacce di morte sui propri account social. Per diverse settimane, inoltre, sono stati costretti a nascondere le foto dei propri figli che venivano utilizzati per supportare questa falsa teoria. Addirittura alcune band musicali che erano solite esibirsi nella pizzeria rientrano tra le vittime: la cantante di una band ha dovuto cancellare il suo account Twitter, una band è stata costretta a disattivare i commenti ai propri video caricati su YouTube. La situazione ha raggiunto un punto critico quando, il 4 dicembre 2016, un uomo di nome Edgar Maddison Welch è entrato armato nella pizzeria Comet Ping Pong e ha sparato un colpo, cercando di "indagare" personalmente sulla presunta attività criminale. Fortunatamente, nessuno è rimasto ferito, e Welch è stato arrestato e condannato [3]. Si potrebbero citare moltissimi altri esem-

pi, soprattutto durante la pandemia di COVID-19, durante la quale è stata registrata una crescita smisurata della disseminazione di fake news. Basti pensare che, in Italia, sul sito del ministero della salute è stato pubblicato un archivio di tutte le notizie che sono state diffuse riguardo questo argomento, classificandole come vere o false, fornendo prove a supporto della scelta. Una delle più famose è sicuramente quella secondo cui "fare gargarismi con la candeggina, oppure assumere etanolo o metanolo, protegge dall'infezione da nuovo coronavirus": oltre a classificare la notizia come falsa, sul sito viene specificato che queste pratiche non solo non proteggono dal virus, ma sono anche estremamente pericolose per la salute [12].

L'Intelligenza Artificiale (IA) rappresenta il più valido strumento per affrontare questa sfida. Innanzitutto, occorre fare chiarezza sul significato di questo termine. Per IA forte (strong AI o general AI) si intende una qualche macchina complessa che sia in grado di replicare ed emulare l'intelligenza umana, con le stesse capacità sensoriali e creative. Già a metà del secolo scorso l'idea era quella di realizzare una macchina che potesse apprendere da sola e che fosse in grado di pensare come un umano. La general AI, che ad oggi non esiste, è una forma avanzata di intelligenza artificiale che possiede la capacità di comprendere, apprendere e applicare conoscenze in una vasta gamma di compiti in modo autonomo e adattabile. Ciò che, invece, è possibile realizzare sono sistemi in grado di eseguire dei compiti specifici in tempi inferiori e con accuratezza migliore rispetto a come li eseguirebbe l'essere umano: si parla di tecniche di Machine Learning (ML), ovvero di apprendimento automatico. Il ML rientra nel campo dell'IA debole (weak AI o narrow AI) e non si pone l'obiettivo di emulare l'intelligenza umana,

ma sfrutta una base matematica per addestrare un modello e utilizzare poi quel modello per fare predizioni future. Il ML propone diversi algoritmi, ma una delle tecniche più rilevanti è quella delle reti neurali, ispirate alla struttura e al funzionamento del cervello umano. Esse sono composte da neuroni artificiali detti "perceptrons", che sono unità computazionali che emulano i neuroni biologici, i quali sono organizzati su più livelli. Si individuano un livello di input, che riceve in ingresso i dati da elaborare ed è composto da tanti neuroni quanti sono gli elementi che compongono il vettore su cui si intende effettuare la predizione. Ci sono poi i layers intermedi (o nascosti), che operano sui dati ricevuti dal layer precedente e che producono un output che rappresenta l'input per il layer successivo. Infine, c'è i layer di output che produce il risultato finale dell'elaborazione e la sua dimensione dipende dalla natura del problema. Il modo in cui si sceglie di interconnettere i neuroni dei livelli intermedi determina un aspetto strutturale della rete. Infatti, se tutti i neuroni di ogni livello sono connessi a tutti quelli del livello precedente, si parla di reti *fully connected*, mentre fissando un numero $M$ di connessioni che ogni neurone deve risolvere casualmente con i nodi del livello precedente si ottiene una *Random Neural Network*. Ci sono poi strutture più complesse, come i layers *pooled* in cui si usano tecniche di clustering per ridurre la dimensionalità dei dati, al fine di mitigare il costo computazionale della rete. L'elaborazione prodotta dal neurone dipende dalla cosiddetta *funzione di attivazione*, una vera e propria funzione matematica con cui viene realizzata la predizione. Le reti neurali possono essere semplici, con pochi strati (reti neurali Feedforward), o molto complesse, con molti strati nascosti (reti neurali profonde o deep). Il Deep Learning (DL) è dunque una branca

del ML, che fa uso di reti neurali composte da tanti livelli interni utilizzate per la risoluzione di problemi complessi. L'addestramento delle reti neurali profonde richiede grandi quantità di dati (per cui è legato al concetto di Big Data) e notevoli risorse computazionali per la loro gestione, motivo per cui tipicamente si ricorre all'l'utilizzo di hardware specializzato come GPU (Graphics Processing Unit) e cluster per il calcolo parallelo. L'effetto benefico del DL è evidente in termini di risultati, che possono superare quelli ottenuti con metodi di ML tradizionali. In generale, un algoritmo di ML viene utilizzato per scoprire qualcosa sui dati che si hanno a disposizione per poter poi fare delle previsioni su dati in futuro. L'output di un algoritmo di ML prende il nome di label (etichetta), che può essere un valore categorico (e in tal caso si parla di classificazione) o numerico (nel caso della regressione). In base alla presenza o all'assenza della label nel dataset utilizzato per l'addestramento, si parla di:

- Learning supervisionato: il modello viene addestrato utilizzando un dataset etichettato, cioè un insieme di esempi di dati che sono accompagnati dalle loro labels corrette. L'obiettivo è che il modello impari a mappare gli input alle etichette corrette. In questa tesi vengono utilizzate tecniche che rientrano in tale campo.

- Learning non supervisionato: il modello viene addestrato utilizzando un dataset non etichettato. L'obiettivo è quello di scoprire la struttura nascosta, i pattern o le correlazioni nei dati. Questo approccio è spesso utilizzato per il clustering e la riduzione della dimensionalità.

- Learning semi-supervisionato: è una via di mezzo tra l'apprendimen-

to supervisionato e quello non supervisionato. In questo approccio, il modello viene addestrato utilizzando un piccolo insieme di dati etichettati insieme a un grande insieme di dati non etichettati. L'idea è che anche una piccola quantità di dati etichettati possa significativamente migliorare le prestazioni del modello, sfruttando la grande quantità di dati non etichettati disponibili. L'obiettivo, in tal caso, è quello di migliorare la previsione delle etichette utilizzando dati non etichettati.

L'addestramento (o training) di una rete neurale è il processo con cui il modello apprende dai dati forniti per fare previsioni o prendere decisioni, che rientrano nel set di training, appunto. Questo processo coinvolge diverse fasi e tecniche per ottimizzare le prestazioni del modello. Tipicamente non si utilizzano tutti i dati a disposizione, bensì si effettua una divisione in training set e test set. Il primo serve proprio per la fase di addestramento del sistema di ML, mentre il secondo fornisce un modo per valutare quanto bene il modello generalizza a nuovi dati che non ha mai visto prima. Inizialmente si assegnano dei valori casuali ai pesi della rete, cioè valori numerici associati ai collegamenti tra i neuroni nei vari layers. Per ognuno degli esempi nel set di addestramento, si effettua quello che prende il nome di "forward pass", che consiste nell'ottenere l'output partendo dall'input, passando per i layer intermedi, ma muovendosi solo in avanti. Una fase delicata è la scelta della funzione di perdita (loss function) che quantifica l'errore (o il costo) tra le previsioni del modello (l'output, appunto) e i valori attesi. Tipicamente, si utilizza la cross-entropy per la classificazione e l'errore quadratico medio (MSE) per la regressione. Si procede poi con la "backpropagation" o "backward pass", ovvero la fase in cui si aggiornano i parametri della rete

muovendosi in senso opposto e calcolando le derivate parziali dell'errore rispetto ad ogni peso. Per fare ciò si usano degli algoritmi di ottimizzazione, come il "gradient descent". L'obiettivo dell'aggiornamento dei pesi è quello di minimizzare il costo, con un adeguamento proporzionale al gradiente dell'errore rispetto a quel peso e moltiplicato per un fattore chiamato learning rate (o tasso di apprendimento). Per fare ciò, viene definito un numero di epoche, ovvero il numero di scansioni del set di training.

L'uso di algoritmi di ML per rilevare notizie false è diventato molto frequente, ma sono tante le sfide relative all'individuazione automatica delle fake news sui social media. In primo luogo, le fake news sono scritte intenzionalmente per ingannare i lettori, il che le rende non banali da rilevare semplicemente in base al contenuto. Ad esempio, le fake news possono citare prove vere in un contesto non corretto per sostenere un'affermazione non vera [19]. Per migliorare l'attività di detection è necessario sfruttare altre informazioni ausiliarie, come la base di conoscenza e il coinvolgimento sociale degli utenti. In secondo luogo, lo sfruttamento di queste informazioni ausiliarie pone un'altra sfida critica: la qualità dei dati a disposizione. Molto spesso non è possibile verificare alcune notizie per mancanza di prove o di affermazioni di supporto. Inoltre, l'impegno sociale degli utenti produce una mole di dati incompleti, non strutturati e rumorosi. È per questi motivi che sta emergendo l'attività di estrazione delle caratteristiche dai post nei social media, sfruttando e interazioni di rete. Negli ultimi anni, i metodi basati sui grafi hanno dato ottimi risultati, in quanto sono in grado di modellare da vicino il contesto sociale e il processo di propagazione delle notizie online. I primi approcci per la "detection" si sono serviti di metodi basati sul deep

learning, ma negli ultimi anni i metodi basati sui grafi e sulle *GNN* (Graph Neural Network) hanno prodotto risultati soddisfacenti. Il motivo principale è da ricercare nella possibilità che offrono tali metodi di modellare sia il contesto sociale che il processo di propagazione delle notizie online [16].

In questa tesi vengono proposti due esperimenti pratici. Con il primo si va ad esplorare l'applicazione di tecniche basate su grafi per effettuare un'analisi delle fake news. In particolare è stato implementato un programma per l'estrazione, tramite web scraping, delle notizie dal sito di fact-checking PolitiFact tramite l'inserimento di una parola chiave e a partire da queste è stato realizzato un grafo bipartito, usato come base dell'analisi. Nella fattispecie, il grafo in questione è stato utilizzato per comprendere se ci fossero degli utenti particolarmente inclini alla pubblicazione di fake news o argomenti rispetto ai quali si tende a diffondere maggiormente notizie false. Il secondo esperimento riguarda l'analisi di una rete di propagazione gerarchica delle fake news, condotta a partire da un esperimento preesistente, su un dataset estratto da FakeNewsNet e in cui i dati consentissero di applicare tecniche di detection basate sui grafi. In questo caso sono state valutate delle caratteristiche strutturali, temporali e linguistiche della rete analizzata, al fine di utilizzarle per l'addestramento di alcuni classificatori supervisionati, al fine di capire l'utilità di queste fetures ai fini del rilevamento delle fake news.

Il lavoro è organizzato in cinque capitoli, nel primo dei quali si presenta una panoramica del contenuto della tesi, gli strumenti che vengono utilizzati al suo interno, dunque l'IA, gli algoritmi di ML, le reti neurali e la teoria dei grafi. La seconda parte del capitolo è dedicata allo stato dell'arte, dei lavori prodotti in letteratura e analizzati al fine di comprendere lo stato attuale della materia. Si evidenzia l'importanza di adottare tecniche basate su grafi per analizzare la propagazione delle fake news, soprattutto in ottica di detection delle stesse. Si analizzano poi i dataset più utilizzati in letteratura. Il secondo capitolo è dedicato alla realizzazione di un grafo bipartito al fine di estrarre informazioni utili per la rilevazione delle fake news utilizzando dei dati estratti, tramite web scraping, direttamente dal sito di fact-checking PolitiFact. II terzo capitolo, invece, introduce lo studio di una rete di propagazione gerarchica per il rilevamento delle fake news, condotto a partire da un lavoro già esistente [18] in cui vengono utilizzati algoritmi di Machine Learning per la rilevazione delle fake news. In questo caso, il dataset di riferimento è FakeNewsNet [17] e l'obiettivo è quello di utilizzare le caratteristiche della rete delineate dal lavoro originale per effettuare previsioni sulle notizie nel social network X. Il contributo della tesi, in questo senso, è quello di creare una rete neurale Feedforward per ottenere dei risultati aggiuntivi e confrontarli con quelli ottenuti dagli altri algoritmi. Inoltre, viene condotta un'analisi di sensitività degli algoritmi di classificazione utilizzati rispetto alle caratteristiche strutturali, temporali e linguistiche della rete individuate nel lavoro originale. Nel quarto capitolo vengono presentati e discussi i risultati ottenuti. Infine, il capitolo conclusivo è dedicato alle sfide ancora aperte di fronte ai problemi della fake news detection e della loro diffusione e a dei

possibili sviluppi futuri.

## 1.1 Stato dell'arte

Il primo passo per analizzare e mitigare la diffusione delle fake news è quello di comprendere lo stato attuale della ricerca per identificare le metodologie esistenti e gli strumenti di supporto disponibili in letteratura. Questo capitolo fornisce una panoramica delle tecniche e degli approcci più rilevanti nel campo, con particolare attenzione all'uso di tecniche basate su grafi per l'analisi della propagazione delle fake news, ma anche un'analisi approfondita dei dataset più rilevanti nell'ambito della detection.

Sebbene il termine "fake news" sia ampiamente utilizzato, tanto da essere nominato parola dell'anno dal dizionario Macquire nel 2016, non esiste una definizione condivisa. È dunque opportuno riportare diverse definizioni fornite dai ricercatori e analizzarle nel dettaglio. In [3] gli autori definiscono una fake news come "una notizia intenzionalmente e verificabilmente falsa, in grado di fuorviare i lettori". Gli aspetti principali di questa definzione sono *autenticità* e *intenzionalità*: secondo questa definizione, le fake news includono informazioni false che possono essere verificate come tali per mezzo di altre fonti e sono create intenzionalmente per fuorviare o disinformare i consumatori. Un'altra definizione molto utilizzata è quella fornita in [9], secondo cui "le fake news sono informazioni inventate che imitano i media nella forma, ma non nel processo organizzativo o nelle intenzioni". In questo caso si enfatizza l'aspetto ingannevole che caratterizza la diffusione di questo tipo di notizie.

In generale, però, le fake news presentano delle peculiarità che le caratterizzano :

- *Effetto "echo chamber"*: una "camera d'eco" può essere definita come un ambiente che si concentra sulle opinioni di utenti con la stessa inclinazione politica o, più in generale, un opinione concorde su un determinato argomento. Questo concetto è rafforzato dalle ripetute interazioni con altri utenti con atteggiamenti e opinioni simili. L'idea è che nel momento in cui non si dispone di informazioni sufficienti per determinare la veridicità di una notizia, allora si sfrutta la credibilità sociale di un utente. Tuttavia, molte persone percepiscono le notizie come credibili e le condividono, creando un'approvazione generale anche nel caso si tratti di una fake news. In tal caso, si parla di euristica della frequenza.

- *Intenzione di ingannare*: questa caratteristica è definita sulla nell'ipotesi che "nessuno produce inavvertitamente informazioni inesatte nello stile degli articoli di cronaca, e il genere delle fake news è creato deliberatamente". L'inganno è motivato da ragioni politiche/ideologiche o finanziarie, ma l'obiettivo con cui vengono diffuse notizie fuorvianti può essere quello del divertimento, dell' intrattenimento o della provocazione. In questo caso, la lingua inglese offre una terminolgia più specifica. Si utilizza il termine *misinformation* per indicare una qualsiasi informazione errata, indipendentemente dall'intenzionalità con cui viene trasmessa, mentre *disinformation* significa "informazione intenzionalmente falsa". Dunque, in questo lavoro di tesi, ci si riferisce all'analisi della disinformation.

- *Account malevoli*: si tratta degli account creati principalmente per dif-

fondere fake news. Si distinguono tre tipi di account di questo genere: social bots, trolls e utenti cyborg. I social bots sono account di social network controllati da algoritmi informatici, creati con lo scopo di creare e diffondere contenuti ingannevoli. È possibile che alcuni social bots siano in grado anche di interagire in maniera automatica con altri utenti di un social network. I trolls sono persone reali che disturbano le comunità online per provocare una risposta emotiva da parte degli utenti dei social media. Il loro obiettivo è quello di manipolare le informazioni per influenzare l'opinione altrui, suscitando emozioni negative tra gli utenti dei social network, come la confusione nel riconoscere le notizie vere e quelle false, inducendoli a dubitare di quelle vere e credere a quelle false. Gli utenti cyborg sono account malevoli creati da persone reali, ma che restano attivi tramite dei programmi specifici, il che li rende particolarmente efficienti nella diffusione di fake news.

- *Autenticità*: per determinare se una notizia sia vera o falsa, bisogna innanzitutto escludere le opinioni soggettive, in quanto solo quelle oggettive possono essere valutate.

- *L'informazione è una notizia*: questa caratteristica riflette se l'informazione è una notizia o meno. Per verificare ciò, si potrebbe valutare se a pubblicare la notizia sia stata un'agenzia di stampa o una testata giornalistica, per esempio.

Sulla base delle caratteristiche sopracitate, è possibile definire una fake news come una notizia contenente affermazioni non veritiere, pubblicate da account

malevoli, che possono provocare l'effetto "echo chamber", con l'intento di ingannare il pubblico.

È opportuno precisare che le fake news rappresentano solo una parte della disinformazione che caratterizza il web. I termini più comuni sono fake news e rumors (o rumours, che sta per pettegolezzi, voci di corridoio, dicerie). In letteratura sono state proposte diverse categorizzazioni di fake news e rumors, in base alla fonte e al tipo di dati utilizzati per l'analisi. I primi studi in questo campo, soprattutto da una prospettiva computazionale, sono relativamente recenti. Pertanto, i confini della materia di studio spesso non sono chiaramente definiti. Una celebre definizione di Cass R. Sunstein afferma che i rumours sono "affermazioni di fatto - su persone, gruppi, eventi e istituzioni - che non sono state dimostrate vere, ma che passano da una persona all'altra, e quindi hanno credibilità non perché siano disponibili prove dirette a sostegno, ma perché altre persone sembrano crederci".

## 1.2 Fake news detection

Per *fake news detection* si intende il processo con cui si rilevano le notizie false. Per fornire una definizione più formale, in [19] viene presentata una definizione di tale processo basato sulle caratteristiche del contenuto e del contesto delle notizie. Date le interazioni sociali $\mathcal{E}$ tra $N$ utenti per una notizia $a$, l'obiettivo della fake news detection è quello di prevedere se $a$ è una notizia falsa o una notizia reale. Questo obiettivo è definito da una funzione di predizione $F : \mathcal{E} \rightarrow 0, 1$, tale che:

$$F(a) = \begin{cases} 1 & \text{se } a \text{ è una fake news} \\ 0 & \text{altrimenti} \end{cases} \tag{1.1}$$

Dunque, la funzione di previsione $F$ è definita come una funzione di classificazione binaria.

Per propagazione delle fake news si intende quel processo attraverso il quale le informazioni false, appunto, vengono diffuse attraverso vari canali di comunicazione, come i social media, i siti web, i mezzi di comunicazione tradizionali e altri mezzi di diffusione dell'informazione. Lo studio della propagazione delle notizie rappresenta la base di uno degli approcci per il rilevamento delle Fake News [16], come riportato in Fig. 1.1.

Si distinguono cinque principali categorie di tecniche per il rilevamento delle Fake News.

1. *Tecniche basate sul contenuto*: si concentrano sulla verifica delle informazioni presenti nelle notizie e includono gli approcci basati sulla

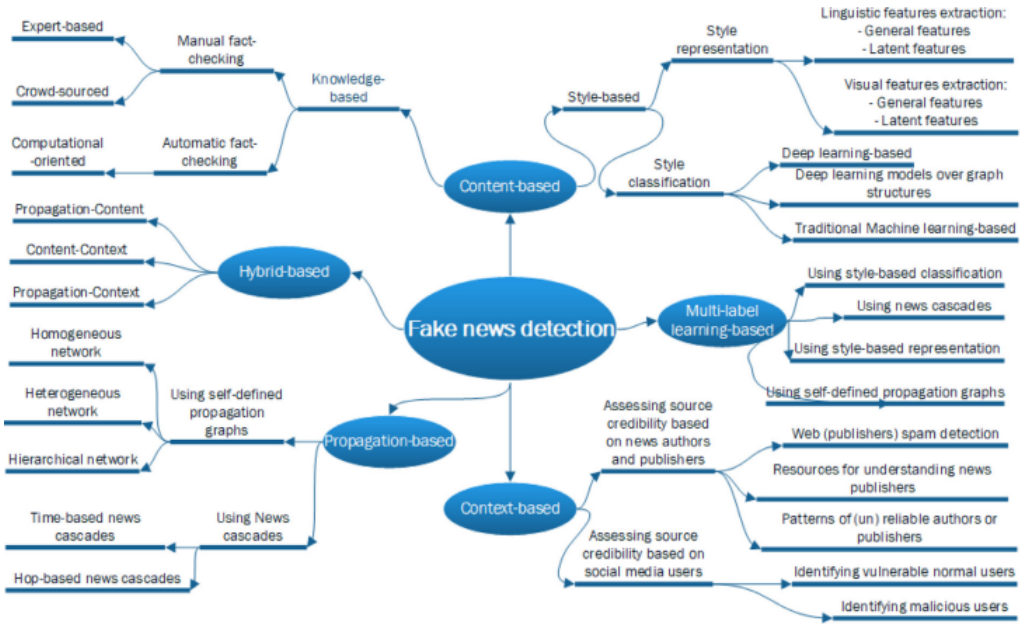



**Figura 1.1:** Panoramica delle tecniche di rilevamento delle Fake News.

conoscenza e sullo stile. Le prime, basate sulla conoscenza, mirano a utilizzare fonti esterne, manualmente o automaticamente, per stabilire la veridicità delle notizie. In particolare si parla di "manual fact-checking" nel caso in cui sia compito di alcuni esperti del dominio verificare le affermazioni manualmente (come nel caso di PolitiFact o Snopes, per esempio) oppure si sfrutti il cosiddetto "crowd-sourcing", ovvero il processo di verifica delle informazioni attraverso la partecipazione collettiva di un ampio gruppo di persone, generalmente volontari, su piattaforme online. In questo contesto, il termine "crowdsourcing" indica l'uso di contributi da parte della comunità per svolgere compiti o risolvere problemi che altrimenti richiederebbero risorse significative. Per quanto riguarda il fact-checking basato sul calcolo ha l'obiettivo di fornire un sistema automatico scalabile per classificare le affermazioni

vere e false. Le seconde, basate sullo stile, sono molto importanti in quanto i pubblicatori di fake news adottano particolari stili di scrittura al fine di attrarre e persuadere un'ampia gamma di consumatori.

2. *Tecniche basate sul contesto*: valutano la credibilità delle fonti e delle interazioni, cercando di individuare segnali di affidabilità o di dubbia provenienza. Infatti, le fake news sono scritte intenzionalmente per indurre i lettori a credere a informazioni false, il che le rende difficili e non banali da rilevare in base al solo contenuto testuale [10]. La natura dei social media facilita questo processo, in quanto è possibile analizzare la credibilità degli utenti, degli autori e dei pubblicatori.

3. *Tecniche basate sulla propagazione*: esaminano il modo in cui le fake news si diffondono attraverso le reti sociali, cercando di identificare modelli di diffusione che possano suggerire la presenza di contenuti ingannevoli. I mezzi con cui viene condotta questa analisi sono le cosiddette "news cascades" e i grafi di propagazione, che possono essere omogenei (vi è un unico tipo di entità, come un post o un evento), eterogenei (più tipi di entità) o gerarchici, che verranno approfonditi successivamente.

4. *Tecniche basate sull'apprendimento multi-label*: considerano le notizie come appartenenti a più categorie contemporaneamente, cercando di catturare la complessità e la sfumatura del fenomeno delle fake news.

5. *Tecniche ibride*: combinano più approcci per migliorare l'efficacia del rilevamento, sfruttando le diverse prospettive offerte dalle diverse ca-

tegorie di metodologie.

È stato citato il fact-checking, per cui è opportuno riportare quelli che sono i siti di fact-checking più importanti in letteratura. Questi siti analizzano affermazioni fatte da politici, personaggi pubblici, e media per stabilire la loro veridicità, spesso fornendo prove e fonti che supportano le loro conclusioni. L'obiettivo che si pongono è combattere la disinformazione e aiutare il pubblico a distinguere tra notizie reali e notizie false. È importante sottolineare che operano senza affiliazioni politiche o economiche che possano influenzare il loro giudizio, forniscono link e riferimenti alle fonti utilizzate per verificare le informazioni, spiegano i metodi usati per la verifica dei fatti e se commettono errori, li correggono pubblicamente per mantenere la credibilità.

- *PolitiFact*: sito di fact-checking nato nel 2007 come progetto del quotidiano Tampa Bay Times, in cui i giornalisti valutano le dichiarazioni originali rilasciate da attori della politica statunitense e pubblicano i loro risultati sul sito web. Ogni dichiarazione riceve una valutazione nel "Truth-O-Meter" ("veritometro"), una scala che va da "True" per le affermazioni vere a "Pants on fire" (espressione creata dal modo di dire "Liar, liar, pants on fire" - "Bugiardo, bugiardo, pantaloni in fiamme") per dichiarazioni false.

- *GossipCop*: è un sito web specializzato nella verifica delle notizie e delle voci riguardanti le celebrità e il mondo dell'intrattenimento. Fondato nel 2009, si distingue per il suo impegno nel smentire falsi rumors e fornire informazioni accurate riguardanti le star di Hollywood e altre figure

pubbliche del settore dello spettacolo. Ogni storia viene valutata con un punteggio di veridicità che va da 0 a 10, per indicare rispettivamente notizie completamente false e notizie completamente vere.

- *Snopes*: fondato nel 1994, è uno dei più antichi e conosciuti siti di fact-checking. Inizia verificando le leggende metropolitane e si espande per includere notizie virali e affermazioni pubbliche. Spazia dalla politica alla cultura pop, coprendo un'ampia gamma di argomenti.

- *FactCheck.org*: si tratta di un progetto dell'Annenberg Public Policy Center dell'Università della Pennsylvania, che verifica l'accuratezza delle dichiarazioni politiche negli Stati Uniti. Fornisce analisi dettagliate e basate su prove delle dichiarazioni politiche, con un focus su campagne elettorali e politiche pubbliche.

L'importanza dei siti di fact-checking risulta evidente se si pensa che la maggior parte dei dataset utilizzati per il rilevamente delle fake news sono realizzati a partire dalle informazioni contenute al loro interno. È proprio questo il motivo principale per cui all'interno di questo lavoro di tesi, per realizzare uno dei due esperimenti, si è scelto di estrarre le informazioni direttamente dal sito PolitiFact piuttosto che utilizzare le informazioni contenute in un dataset pubblico.

Il progresso delle tecniche di deep learning ha inaugurato una nuova era di rilevamento delle fake news, con un gran numero di metodi basati sull'apprendimento profondo, in particolare sui grafi. I metodi di fake news detection basati sui grafi, sono classificati da [7] in tre categorie, come mostrato in Fig. 1.2.

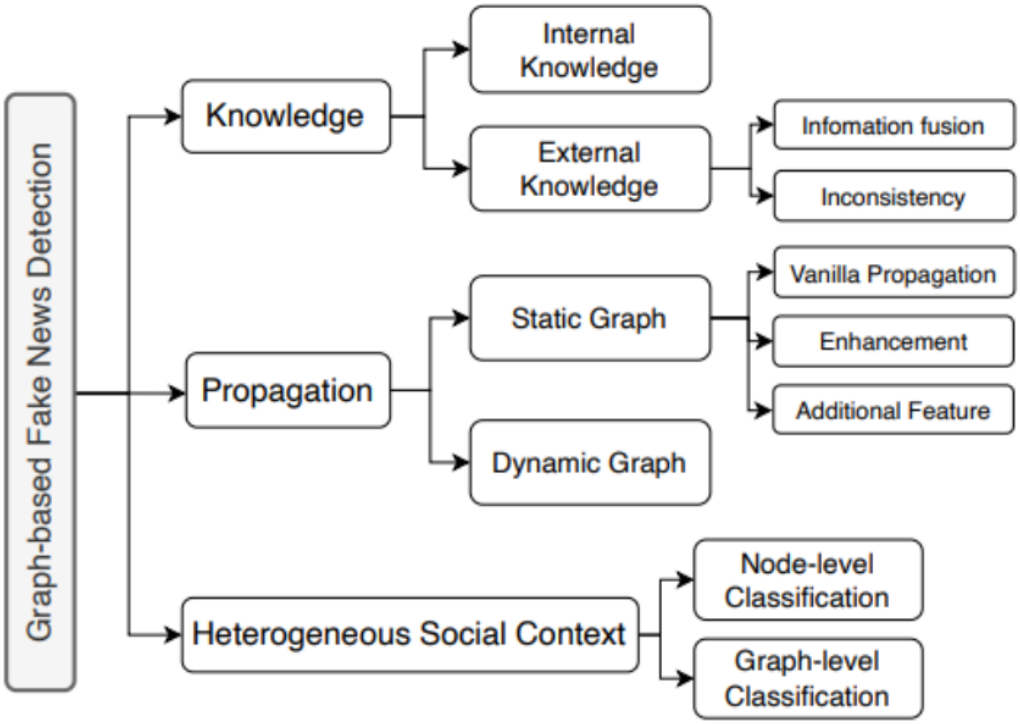



**Figura 1.2:** Tassonomia dei metodi per la Fake News Detection basati sui grafi.

- *Metodi basati sulla conoscenza*: sfruttano le entità presenti nel contenuto delle notizie ai fini della detection. Le tecniche più utilizzate sono quelle del Natural Language Processing (NLP) per pre-elaborare il contenuto del testo ed estrarre le entità di interesse. Si fa, inoltre, distinzione tra metodi basati su conoscenza interna o esterna: a volte la conoscenza diretta (interna) estratta dai contenuti delle notizie può non essere sufficiente, e quindi possono essere utilizzate fonti di conoscenza esterne (ad esempio, Wikipedia). Per certi versi, l'approccio che sfrutta la conoscenza esterna imita il modo in cui gli esseri umani provano a rilevare le fake news, cioè affidarsi a fonti autorevoli per valutare la verdicità di una notizia.Inoltre, si distinguono due approcci per accorpare conoscenza interna ed esterna. Si parla di *information fusion* quando

le entità di conoscenza interna che vengono estratte sono collegate a entità esterne, oppure di *inconsistency detection* se la veridicità delle notizie viene predetta rilevando l'inconsistenza tra il contenuto estratto internamente e quello estratto esternamente.

- *Metodi basati sulla propagazione:* si concentrano sul processo di diffusione delle notizie e possono essere classificati in due ulteriori sottocategorie, ovvero quelli *statica graph-based* e quelli *dynamic graph-based*. Il fattore determinante è quello temporale, cioè se l'intera struttura, che comprende nodi e archi, emerge istantaneamente o meno.

- *Metodi basati sulla contesto sociale eterogeneo:* hanno come obiettivo quello di estrarre il contesto della notizia di partenza. Supponendo di trovarsi nell'ambito dei social network, il contesto sarebbe rappresentato dagli altri post che provengono dallo stesso utente insieme ad altre notizie sullo stesso argomento.

Trattandosi di metodi basati sui grafi, per ognuno di essi si ottiene una tipologia di grafo diversa. Rispettivamente, il grafo ottenuto a partire da ogni tecnica è:

- *Grafo di conoscenza* $G_k$: descrive le connessioni tra le entità $\mathcal{E}\backslash=\{en_i\}$, per cui i nodi sono le entità (frasi, parole, argomenti) e gli archi le connessioni tra esse. In caso di metodi basati su conoscenza esterna, $\mathcal{E}\backslash$ include anche le entità estratte esternamente.

- *Grafo di propagazione*: gli utenti e le loro interazioni (commenti, repost) formano una struttura ad albero (grafo diretto aciclico), il cui

nodo radice è il post di origine e gli altri nodi sono commenti/re-post ad esso relativi. In questo caso, il grafo relativo a una notizia $a$ è composto da un insieme di tuple $\mathcal{E}$=$\{e_{it}\}$ e rappresenta il processo con cui la notizia $a$ si diffonde in un tempo $t$ tra $n$ utenti $\mathcal{U}$=$\{u_1,u_2,...,u_n\}$, i loro post $\mathcal{P}$=$\{p_1,p_2,...,p_n\}$ ed eventuali re-post o commenti. Ogni tupla $e_{it}$=$<u_i,p_i,t>$ rappesenta un utente $u_i$ che ricondivide una notizia $a$ al tempo $t$ e con un possibile commento $p_i$.

- *Grafo eterogeneo*: è caratterizzato da più tipi di nodi e più tipi di archi, dal momento che il contesto sociale è formato da diversi tipi di entità (utenti, notizie, commenti e anche argomenti/topic) e connessioni (ad esempio, connessioni utente-utente, follower-followee, connessioni utente-news post). In questo caso sono importanti le connessioni implicite tra più notizie, come ad esempio il fatto che più notizie vengano pubblicate dagli stessi utenti.

Una differenza fondamentale tra i metodi basati sulla propagazione e i metodi basati sul contesto sociale eterogeneo è il processo di applicazione della modellazione a grafo. Quando si utilizza un grafo per modellare il processo di propagazione delle singole notizie, si tratta di un metodo basato sulla propagazione, mentre se un grafo viene applicato per modellare un contesto sociale più ampio che coinvolge più notizie e utenti che interagiscono su articoli diversi o correlati, è necessario adottare un metodo basato sul contesto sociale eterogeneo. In quest'ultimo caso, il grafo viene utilizzato per determinare la veridicità di più notizie allo stesso tempo.

In base poi al tipo di approccio utilizzato, è possibile estrarre caratteristi-

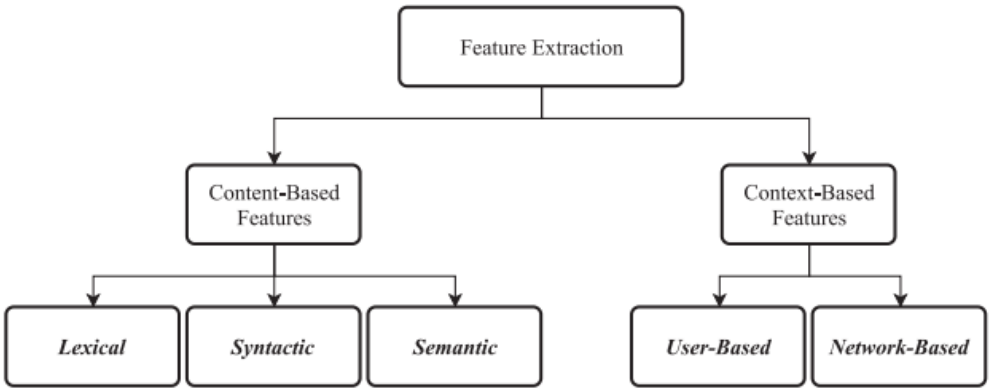



**Figura 1.3:** Presentazione dei diversi tipi di features usati in letteratura per la fake news detection.

che basate sul contenuto e caratteristiche basate sul contesto, come spiegato in Fig. 1.3 [4].

Gli approcci basati sul contenuto permettono di valutare caratteristiche del contenuto delle notizie, le quali si possono estrarre direttamente dal testo, Il vantaggio di questi approcci è che, come già detto, utilizzando tecniche come quella del NLP queste caratteristiche si possono ottenere facilmente. Si distinguono caratterisitche lessicali, sintattiche e semantiche. Le prime riguardano l'uso effettivo delle parole all'interno di un testo, per cui tipicamente si sfruttano le parole o le espressioni più significative del contenuto. È stato dimostrato che le negazioni, le parole di dubbio, le abbreviazioni e le espressioni volgari possono essere utili nel determinare la veridicità delle informazioni online [23]. Quelle sintattiche possono essere relative al numero di parole, al fine di individuare espressioni rilevanti all'interno del testo. Ad esempio, la complessità di una frase è stata identificata come indicazione dell'affidabilità delle informazioni contenute al suo interno [5]. Le caratteristiche semantiche, invece, sono estratte principalmente utilizzando tecniche

avanzate di NLP. Una delle tecniche principali prende il nome di *sentiment analysis*, cioè quel processo con cui si analizza un testo per determinare la natura delle emozioni e delle opinioni contenute nella notizia. Gli approcci basati sul contesto consentono analisi più varie e si basano su informazioni circostanti riguardo il post sui social media o la fake news. Nel caso dell'analisi degli utenti si considerano caratteristiche come le fonti delle voci o delle notizie, i pattern di propagazione delle informazioni sui social media e le reazioni degli altri utenti rispetto alla notizia o al post. Inoltre, si tiene conto delle informazioni dell'utente, dunque del numero di post che pubblica, data in cui è stato creato l'account, numero di amici/follower. Estrarre informazioni riguardo gli utenti, però, non è sempre possibile. Come spiegato in seguito, le politiche sulla privacy degli utenti sono sempre più restrittive, per cui non sempre si ha la possibilità di accedervi. Inoltre, solo poche informazioni sono rappresentate allo stesso modo su diverse piattaforme, il che ne complica l'analisi. Per quanto riguarda le caratteristiche della rete, vengono prese in considerazione le strutture di propagazione, i modelli di diffusione e le proprietà del sottografo in cui la notizia viene diffusa. La maggior parte degli studi esistenti che implementano caratteristiche orientate alla rete si limitano all'uso di statistiche sui modelli di diffusione, come il numero di retweet e i tempi di propagazione [11].

## 1.3 Dataset per il rilevamento delle fake news

La scelta del dataset, soprattutto nel più ampio contesto del rilevamento automatico delle Fake News, rappresenta la vera sfida. A prescindere dall'obiettivo, è fondamentale trovare dataset ricchi e completamente etichettati, in modo da poter addestrare e testare gli algoritmi. Di seguito vengono presentati i dataset più utilizzati per la rilevazione di Fake News [6, 15].

- ***FakeNewsNet:*** dataset in lingua inglese composto da dati di Twitter, con ID di tweet e retweet, utile per la costruzione di propagation chains. I topic principali che caratterizzano le notizie al suo interno sono politica e intrattenimento. È il dataset che presenta il maggior numero di informazioni sul contesto sociale e caratteristiche riguardo l'evoluzione della diffusione. Viene utilizzato in [17, 18], i quali lavorano rispettivamente su dati in formato CSV e JSON estratti da questo dataset.

- ***LIAR:*** dataset che raccoglie dati riguardanti aspetti politici da Politifact.com, include 12836 statement, ognuno dei quali classificato secondo 6 gradi di verità. I dati sono raccolti da Twitter e Facebook, dal 2007 al 2016. In [24, 20] vengono condotti dei lavori che usano questo dataset.

- ***PHEME:*** dataset multilingue (inglese e tedesco) per i rumours relativi a 9 eventi, ognuno dei quali classificato con la label *true, false o unverified*. Contiene 4842 tweet e 330 voci (rumors). Il dataset contiene tweet associati a eventi di crisi, insieme ad annotazioni riguardanti la loro veridicità [8].

- ***Fever:*** dataset per Fact Extraction e VERification, contenente 185445 affermazioni generate alterando frasi estratte da Wikipedia e poi verificate usando la conoscenza delle stesse dalla fonte da cui sono state estratte [21]. Le affermazioni sono etichettate come *supported, refuted* o *notEnoughInfo*. Per le prime due classi vengono annotate anche le frasi che costituiscono la prova necessaria per il loro giudizio.

- ***CREDBANK:*** dataset che raccoglie circa 60 milioni di tweet nel periodo che va da ottobre 2014 a febbraio 2015, per cui si presta molto bene a metodi di detection che sfruttano reti neurali deep [13]. I dati, relativi a flussi di tweet raccolti in questo arco temporali, topic affrontati, topic classificati come eventi o non eventi ed eventi annotati con un rating di credibilità, sono distribuiti in quattro file i cui valori sono separati da tabulazione: Streaming Tweet File, Topic File, Credibility Annotation File e Searched Tweet File.

- ***Weibo:*** dataset per la Multi-domain Fake News Detection (MFND), composto da 4488 fake news e 4640 notizie reali, provenienti da 9 domini diversi [14].

- ***ReCOVery:*** dataset composto da 2029 articoli riguardanti la pandemia COVID-19 che coprono l'arco temporale che va da gennaio 2020 a maggio 2020 [26].

- ***ISOT:*** contiente notizie false raccolte da siti segnalati come inaffidabili da PolitiFact e Wikipedia e notizie reali raccolte dal sito web Reuters [1, 2]. Il dataset contiene diversi tipi di articoli su diversi argomenti.

In particolare ci sono due file CSV (*True.csv e False.csv*), ognuno dei quali comprende più di 12600 notizie, delle quali riporta le seguenti informazioni: *titolo, testo, argomento e data.*

Un aspetto importante, per quanto riguarda i lavori prodotti in questi anni, è quello legato all'accessibilità delle informazioni pubblicate nei social media. Negli anni, le politiche con cui i social hanno gestito l'accesso ai dati sono cambiate significativamente, principalmente per ragioni legate alla privacy e alla sicurezza, ma anche a leggi più restrittive come il GDPR in Europa e il CCPA in California. Nel caso specifico di X, fino al 2020 ha messo a diposizione l'API v1, che forniva un accesso abbastanza robusto ai dati pubblici, consentendo agli sviluppatori di ottenere tweet, profili utente, e altre informazioni pubbliche. Tra il 2021 e il 2022, è stata introdotta l'API v2 per offrire una struttura più moderna e versatile, ma con alcuni cambiamenti nelle modalità di accesso ai dati. In particolare, sono stati introdotti miglioramenti nella raccolta e nell'uso dei dati e la necessità di autenticazione OAuth 2.0 (Open Authorization). Con l'acquisizione di Twitter da parte di Elon Musk, nel 2023, sono state ridotte alcune funzioni gratuite e introdotti modelli a pagamento per l'accesso a dati di alto volume. Questi cambiamenti hanno determinato delle difficoltà significative nello studio della letteratura, rendendo obsoleti o comunque non più funzionali i lavori che facessero utilizzo dell'API di X prima che cambiassero le politiche, dal momento che i piani a pagamento sono molto ingenti.

# Capitolo 2

# Grafo bipartito

Il primo esperimento condotto in questa tesi verte sulla realizzazione di un grafo bipartito e dunque sull'uso della teoria dei grafi per effettuare delle analisi su un dataset prodotto da uno dei siti di fact-checking più usato in letteratura come fonte di afferazioni verificate.

## 2.1 Creazione del dataset

Per avere a disposizione informazioni quanto più aggiornate possibili e riguardo topic di interesse, si è scelto di estrarre i dati direttamente dal sito di fact-checking PolitiFact, sfruttando la tecnica del *web scraping*. Il termine "scraping" può essere tradotto con "grattare, raschiare", infatti il web scraping è un'attività con cui si vanno a prelevare dati e metadati da un sito web sfruttando dei software, simulando la navigazione umana. Per fare ciò, in Python, è stata utilizzata la libreria *Beautiful Soup*. Il programma è stato implementato sfruttando il progetto accessibile al seguente link: `https://github.`

com/ChangyWen/PolitiFact-scraping?tab=readme-ov-file. Il programma originale consente di effettuare una ricerca inserendo un numero di pagine $N$ e la sorgente della notizia, che su PolitiFact può trattarsi di una persona, di un post su un social (Facebook, Instagram, Threads, X, ecc.), un'organizzazione o un comitato, restituendo tutte le $N$ pagine più recenti relative ad affermazioni con la sorgente specificata. In letteratura, come nel dataset LIAR ad esempio, il termine "sorgente" viene alternato con "speaker", che significa letteralmente "oratore, colui che parla". In particolare, sono stati realizzati due script: uno per creare e popolare il file CSV (Comma-separated values) e l'altro per aggiornarlo con le notizie più recenti relative alla ricerca effettuata, laddove ve ne siano. Ai fini di un'analisi mirata, il programma è stato modificato in modo da permettere una ricerca inserendo una parola chiave e restituire le N pagine più recenti relative alla ricerca effettuata sul sito PolitiFact. I risultati dello scraping sono stati salvati in un file in formato CSV, in modo da poterlo utilizzare per delle elaborazioni. I campi di interesse, i cui valori sono stati memorizzati in questo file, sono:

- *Author*: è il "fact-checker", cioè l'esperto che ha verificato la notizia e l'ha etichettata con una label.
- *Statement*: il contenuto testuale dell'affermazione.
- *Source*: la sorgente della notizia (speaker), che può essere una persona, il post di un social, un'organizzazione.
- *Date*: data della revisione della dichiarazione in formato Mese giorno, anno.

- *Target*: è la label assegnata dal revisore. In PolitiFact il "Truth-O-Meter" prevede sei i possibili valori per questo campo: *True, Mostly True, Half True, Mostly False, False, Pants on Fire.*

- *Keywords*: parola chiave con cui è stata ricercata la dichiarazione.

Per riportare dei risultati a titolo di esempio, sono state effettuate ricerche con sei diverse keywords ("Ukraine", "Russia", "Zelensky", "Putin", "Trump" e "Biden"), fissando il numero di pagine a 25 (il programma itera sulle N-1 pagine più recenti, dunque la ricerca si ferma alla ventiquattresima). Considerando che ogni pagina contiene al più 5 affermazioni, il dataset analizzato comprende 652 affermazioni. Bisogna anche specificare che durante l'analisi delle notizie estratte dal sito PolitiFact, è stato riscontrato che alcune di esse non sono etichettate correttamente. Infatti, oltre le sei etichette già citate, relative al "Truth-O-Meter", PolitiFact prevede anche il "Flip-O-Meter", un meccanismo per valutare i cambiamenti di posizione di un funzionario politico in base alle dichiarazioni che esso rilascia. Questo misuratore prevede tre etichette: *No Flip*, *Half Flip* e *Full Flop*. La prima indica che non c'è stato un cambio di posizione significativo, la seconda che c'è stato un cambiamento parziale e la terza un cambiamento completo di posizione. Si tratta di un aspetto ben diverso da quello del "veritometro", motivo per cui non viene considerato all'interno di questa analisi. All'interno del codice si definiscono come etichette valide solo quelle relative alla fakeness della notizia.

## 2.2 Grafo bipartito

L'obiettivo primario di questo esperimento è stato quello di realizzare un grafo bipartito che avesse da una parte gli speakers e dall'altra parte le keywords con cui sono state effettuate le ricerche, a partire dai dati estratti da PolitiFact e organizzati come descritto precedentemente. Un grafo si dice bipartito quando i suoi nodi possono essere divisi in due gruppi disgiunti, tale per cui ogni nodo di ciascun gruppo è connesso ad almeno un altro nodo dell'altro gruppo. Non si considerano quindi connessioni intra-gruppo. In [22] si dice che in un gruppo ci sono gli agenti e nell'altro le classi cui gli agenti sono affiliate, infatti la matrice che caratterizza l'intera rete è detta *matrice di affiliazione*. Questa matrice prevede sulle righe gli agenti e sulle colonne le classi di affiliazione in cui gli agenti sono coinvolti. Il grafo bipartito ad essa associato è un grafo non diretto e pesato, in cui i pesi sono relativi al grado di fakeness delle notizie. Assegnare un peso agli archi di un grafo bipartito lo rende significativo circa la veridicità delle notizie pubblicate riguardo un determinato argomento. Per facilitare la gestione del grado di veridicità di un'affermazione, ad ogni valore del campo target viene associato un valore numerico nel range tra 0 e 1, con un passo 0.2 (trattandosi di sei valori), come riportato in Tab.2.1. Il valore che indica un grado di fakeness maggiore è associato al valore 1, in quanto si vuole dare maggiore peso alle notizie false.

Per la scelta riguardo il peso da assegnare agli archi sono state valutate due alternative:

1. Prevedere un arco per ogni grado di fakeness e associare un peso pari

| Valori del campo target | Valori numerici |
|---|---|
| Pants on fire | 1 |
| False | 0.8 |
| Mostly-false | 0.6 |
| Half-true | 0.4 |
| Mostly-true | 0.2 |
| True | 0 |

**Tabella 2.1:** Corrispondenza valori del campo *target* con valori numerici.

al prodotto tra il numero di notizie pubblicate dalla stessa sorgente con la parola chiave specificata e il valore numerico associato a quel grado di fakeness.

2. Prevedere un solo arco tra la sorgente e la parola chiave con un peso $P_{sk}$ pari a:

$$P_{sk} = \sum_{i=0}^{5} \left( N_{sk} \cdot \frac{i}{5} \right) \tag{2.1}$$

Con $N_{isk}$ pari al numero di notizie pubblicate dalla sorgente ottenute effettuando una ricerca con quella specifica parola chiave, con grado di fakeness $i$. In questo modo si ha un'informazione sintetica e rappresentativa della veridicità di tutte le notizie pubblicate da uno speaker per un determinato topic.

È stata scelta la seconda opzione, in quanto risulta la più significativa e la più indicata per non far esplodere il numero di archi. Il risultato dell'esecuzione di questo programma viene riportato in un altro file CSV contenente le triple *Source, Keywords, Total Weight*, in ordine decrescente per peso totale e il cui nome è legato alla data e all'ora in cui viene eseguito, nel formato

`"Result_YYYY_mm_DD_hh_MM_ss"`. Questo file viene dato in input ad uno script realizzato in Matlab per un analisi più approfondita.

Il primo passo è stato quello di ricavare, a partire dal file CSV di input, la matrice di affiliazione $E$ tra sorgenti e parole chiave, dunque una matrice $N \times M$, con $N$ pari al numero di sorgenti ed $M$ al numero di parole chiave e il cui generico valore $a_{ij}$ indica il grado di fakeness della notizia. A partire da questa matrice, è stato realizzato un grafo bipartito. In questo caso, il peso degli archi è determinato dalla somma del numero di notizie pubblicate da una sorgente con quella keyword, con un grado di fakeness $i$ e il valore numerico corrispondente al grado di fakeness stesso.

Per un'ulteriore analisi, si è scelto anche di eseguire, separatamente, una normalizzazione sul numero di notizie pubblicate dallo speaker su quel topic. La normalizzazione è stata ottenuta dividendo il peso per $N_{sk}$, cioè il numero di notizie pubblicate da uno speaker circa una keyword specifica. In tal caso, il valore massimo (pari a 1) si otterrebbe se tutte le notizie pubblicate da uno speaker per quel topic fossero false. Dunque, la normalizzazione del peso degli archi nel grafo bipartito porta a una rappresentazione più equilibrata della relazione tra gli speaker e le keywords, tenendo conto non solo della quantità ma anche della qualità delle notizie pubblicate. Infatti, un peso più alto indica ancora una forte associazione tra gli speaker e le keywords, ma ora tiene conto non solo della quantità di notizie pubblicate, ma anche della qualità in termini di fakeness di tali notizie.

## 2.3 Proiezioni del grafo bipartito

Nella teoria dei grafi, ad ogni grafo bipartito possono essere associate due proiezioni, una nello spazio degli agenti e una nello spazio delle classi e per ognuna di esse si ottiene un grafo non diretto e pesato in cui i nodi sono della natura del gruppo scelto per effettuare la proiezione. Per quanto riguarda la proiezione sugli agenti, due nodi (ad esempio $a_1$ e $a_2$) sono collegati da un arco il cui peso è il numero di classi cui entrambi gli agenti sono affiliati ($c_1$ ed $c_2$). La proiezione del grafo bipartito nello spazio degli agenti è rappresentato dalla *matrice di omofilia*, che è possibile calcolare a partire dalla matrice di affiliazione moltiplicandola per la sua trasposta. La matrice risultante $EE^T$ ha dimensione $N \times N$. Interpretando la matrice di omofilia come matrice di adiacenza, è possibile realizzare il grafo associato alla proiezione sulle sorgenti. In questo grafo di sources, due nodi sono connessi solo se hanno prodotto risultati in funzione di almeno una stessa keyword. Dunque, rappresenta il modo in cui i nodi relativi alle sorgenti sono tra loro collegati in base a delle keywords per cui hanno prodotto dei risultati nella ricerca. Quanto più alto è il peso che collega due nodi, tanto più è alto il grado di diffusione di fake news per quei due sources, indipendentemente dalla keyword, per cui pubblicano fake news su keywords diverse.

La seconda proiezione del grafo bipartito può essere effettuata nello spazio delle classi (keywords nel caso di studio). La matrice di adiacenza del grafo pesato risultante può essere ottenuta premoltiplicando la trasposta della matrice di affiliazione per se stessa ($E^T E$), assumendo dunque dimensione $M \times M$. In tal, caso, però, è stata considerata sia la matrice dicotomica (ma-

trice di adiacenza ottenuta sostituendo con degli 1 tutti gli elementi della matrice originale con valore diverso da zero) associata alla matrice di affiliazione, in quanto il sito PolitiFact è polarizzato soprattutto su notizie con alto grado di fakeness, sia la matrice di affiliazione originale. Nel primo caso, un arco che collega due keywords indica il numero di speaker che hanno pubblicato fake news su entrambi i topic relativi alla coppia di keywords considerate, mentre nel secondo esprime l'intensità di associazione (in termini di fakeness) delle due keywords co-citate dagli stessi speakers.

A partire poi da questi due grafi sono state calcolate sia delle misure di nodo sia di rete.

Gli indici di nodo sono misure che quantificano la centralità di un nodo nella rete rispetto a particolari caratteristiche.

Un indice di nodo molto comune è il *grado* di un nodo che si valuta considerando i nodi ai quali esso è direttamente collegato. Per questo motivo, si dice che il grado è un indice di nodo locale. In un grafo non diretto e pesato si definisce grado di un nodo la somma degli archi collegati a quel nodo, senza cioè considerare il peso degli archi. Considerando $W$ la matrice dei pesi, il grado del nodo i $\theta_i$ si esprime come:

$$\theta_i = \sum_{j=1}^{n} (\delta_i j), \qquad (\delta_i j) = \begin{cases} 1 & [W]_{ij} \neq 0 \\ 0 & [W]_{ij} = 0 \end{cases} \tag{2.2}$$

La *betweenness* rappresenta la capacità di un nodo di fungere da intermediario nel connettere altri nodi della rete all'interno di percorsi a distanza minima, cioè lungo le geodetiche. La betweenness di un nodo i, indicata con

$\beta_i$ è il numero di geodetiche dal nodo s al nodo t che passano per il nodo i (indicato con $n_{st}^i$), diviso per il totale delle geodetiche che vanno da s a t (indicato con $g_{st}$), al variare dei nodi s e t nella rete.

$$\beta_i = \sum_{s,t=1}^{n} (\frac{n_{st}^i}{g_{st}}) \tag{2.3}$$

La Tab. 2.2 riporta alcuni parametri strutturali della rete relativa alla proiezione sulle sorgenti considerando i pesi non normalizzati come il numero di nodi $n$ e il numero degli archi $m$ e la densità, che misura il rapporto tra il numero di archi esistenti (M) e il massimo numero di archi possibili tra N nodi .

$$\delta = \frac{2M}{N(N-1)} \tag{2.4}$$

Il diametro $d$, invece, rappresenta la distanza minima da percorrere perché due qualsiasi nodi della rete siano tra loro raggiungibili. In altri termini, il diametro è la misura della geodetica di lunghezza massima all'interno del grafo. Il calcolo del diametro di una rete può essere ottenuto a partire dalla definizione di distanza tra due nodi. In particolare tale distanza, indicata con $d_{ij}$, rappresenta il numero minimo di archi da percorrere per andare dal nodo i al nodo j, considerando il peso ad essi associato. Prendendo il massimo di questi massimi al variare dei nodi della rete si ottiene il diametro:

$$d = \max \left\{ \max\{d_{ij}\}_{j=1}^{N} \right\}_{i=1}^{N} \tag{2.5}$$

Trattandosi di grafi pesati, bisogna fare una precisazione sul significato del

peso associato agli archi: dal momento che nella rilevazione il peso dell'arco viene inteso tanto maggiore quanto più forte è il legame, il peso dell'arco viene definito come il reciproco del peso.

$$[W_d]_{ij} = \begin{cases} \frac{1}{[W]_{ij}} & \text{se } [W]_{ij} \neq 0 \\ 0 & \text{se } [W]_{ij} = 0 \end{cases} \tag{2.6}$$

Inoltre, si assume che $[W_d]_{ij} = 0$ per $i = 1, ..., N$, poiché la distanza di un nodo da sé stesso è nulla.

La Tab. 2.3 riporta invece i parametri relativi alla rete ottenuta con le proiezioni sulle keywords.

Avendo considerato anche il grafo ottenuto a partire dalla matrice dicotomocia associata a quella di affiliazione, in Tab. 2.4 sono riportati i parametri della rete così ottenuta.

| Simbolo | Parametro | Valore |
|---|---|---|
| $n$ | Numero di nodi | 90 |
| $m$ | Numero di archi | 1936 |
| $\overline{\theta}$ | Grado medio | 410.895 |
| $\delta$ | Densità | 0.483 |
| $d$ | Diametro | 0.704 |

**Tabella 2.2:** Parametri della rete associata alla proiezione sulle sorgenti.

| Simbolo | Parametro | Valore |
|---|---|---|
| $n$ | Numero di nodi | 6 |
| $m$ | Numero di archi | 15 |
| $\overline{\theta}$ | Grado medio | 4792.2 |
| $\delta$ | Densità | 1 |
| $d$ | Diametro | 0.002 |

**Tabella 2.3:** Parametri della rete associata alla proiezione sulle keywords.

| Simbolo | Parametro | Valore |
|---|---|---|
| $n$ | Numero di nodi | 6 |
| $m$ | Numero di archi | 15 |
| $\overline{\theta}$ | Grado medio | 64.667 |
| $\delta$ | Densità | 1 |
| $d$ | Diametro | 0.191 |

**Tabella 2.4:** Parametri della rete associata alla proiezione sulle keywords considerando la matrice dicotomica della matrice di affliazione.

# Capitolo 3

# Rete di propagazione gerarchica

Il secondo esperimento condotto all'interno di questa tesi consiste nell'analisi di una rete di propagazione per la rilevazione delle fake news. Il motivo alla base di questo studio è che nel mondo reale, le notizie si diffondono attraverso reti di propagazione. La fase di ricerca è avvenuta principalmente all'interno delle pubblicazioni su questo argomento, con particolare attenzione a quelle contenenti il link dei progetti realizzati. Una criticità che ha polarizzato la scelta del progetto è stata quella relativa all'API di X: come già detto, negli anni, le politiche del social network sono cambiate, per cui sono stati presi in considerazione esclusivamente progetti che non facessero uso delle API di X. La scelta è ricaduta sul progetto contenuto nello studio pubblicato da Shu et al. [18], in cui viene realizzata una rete di propagazione gerarchica ai fini della fake news detection. L'obiettivo dell'esperimento è quello di comprendere le correlazioni tra le reti di propagazione delle notizie e le fake news, per cui il primo passo consiste nel realizzare la rete di propagazione gerarchica a partire dal macro-livello e dal micro-livello delle

notizie, sia vere che false. Viene poi eseguita un'analisi comparativa delle caratteristiche della rete di propagazione dal punto di vista strutturale, temporale e linguistico tra le fake news e le notizie vere, al fine di dimostrare il potenziale dell'utilizzo delle stesse per rilevare le fake news. Infine, utilizzando degli algoritmi di ML supervisionati su un dataset provvisto di label si mostra l'efficacia di queste caratteristiche della rete di propagazione per il rilevamento delle fake news. Gli obiettivi fissati in questa tesi sono molteplici. Innanzitutto è stato replicato il lavoro, applicando delle modifiche funzionali per risolvere alcune incompatibilità riscontrate nelle fasi iniziali, principalmente riguardanti le dipendenze dalle librerie utilizzate. Una volta fatto ciò, è stato riscontrato che il progetto originale fa uso di un set di dati campionato e non del dataset completo, per cui il lavoro è stato con l'obiettivo di ottenere dei risultati in linea con quelli originali. Per migliorare i risultati ottenuti, è stato utilizzato un ulteriore classificatore oltre a quelli proposti (in particoalare una rete Feedforward) e per ognuno di essi sono stati valutati gli iperparametri migliori. Infine, è stata valutata l'importanza di ogni feature per tutti gli algoritmi considerati. L'approccio seguito da questo lavoro ha anche un altro obiettivo, ovvero di supportare l'individuazione delle fake news non considerando esclusivamente il contenuto testuale delle notizie, in quanto le fake news sono scritte intenzionalmente per ingannare i lettori.

## 3.1 Dataset

Essendo i dati dei social media su larga scala, multimodali, per lo più generati dagli utenti, talvolta anonimi e rumorosi, per realizzare la rete di propagazione gerarchica viene utilizzato un dataset estratto dal più ampio set di dati utilizzato in ambito della detection FakeNewsNet. Come già detto, questo dataset estrae informazioni dai siti di fact-checking PolitiFact e GossipCop, per cui i dati vengono organizzati in base alla sorgente da cui provengono. Per rispettare le politiche sulla privacy di X, le informazioni degli utenti vengono rese anonime e i contenuti dei tweet non vengono condivisi. I dati sono nel formato GRAPH JSON (JavaScript Object Notation) e rappresentano i grafi di propagazione delle notizie su X. Un file in questo formato contiene le coppie chiave-valore rappresentanti nodi e archi del grafo, fornendo metadati relativi agli stessi. I nodi di questi grafi rappresentano dei tweet, mentre gli archi le relazioni tra essi. Di seguito viene proposta una descrizione della struttura dei file JSON:

- *time*: timestamp relativo alla pubblicazione del tweet.
- *type*: tipo dell'evento. Il valore 1 è associato al nodo principale, relativo alla notizia, 2 ai tweet che riportano la notizia al loro interno, 3 a un retweet e 4 a una risposta.
- *user*: ID randomico dell'utente che ha generato l'interazione.
- *tweet_id*: ID del tweet, del retweet o della risposta associata all'evento.
- *id*: ID univoco del tweet associato all'interazione.

- *children*: campo opzionale, è una lista delle interazioni collegate al nodo.

- *bot_score*: campo opzionale che indica la probabilità che l'utente sia un bot. È un parametro che riguarda esclusivamente la rete di retweet, per cui riguarda solo il livello di propagazione macro.

- *sentiment*: campo opzionale il cui valore è compreso in un intervallo tra -1 (negativo) e 1 (positivo). Si tratta dunque di una scala continua che permette di esprimere la polarità del sentimento in modo graduale e in cui il valore 0 rappresenta un sentimento neutro. Questo parametro riguarda solo la reply-chain, dunque la rete di micro-livello.

I file sono organizzati in quattro directories: per ognuno dei due siti di fact-checking da cui FakeNewsNet estrapola informazioni (PolitiFact e GossipCop), c'è una cartella relativa alle notizie reali e una dedicata alle fake news. Le statistiche del dataset, comprese le loro dimensioni, sono riportate in Tab.3.1. Bisogna precisare che il progetto non contiene il codice per la produzione del dataset utilizzato come caso di studio. Si sceglie casualmente l'80% delle notizie per la fase di training e il restante 20% per il testing. Analizzando i dati, però, è stata riscontrata una criticità non evidenziata dal lavoro originale: per la rete di micro-livello generata a partire dai dati di GossipCop, non vi sono informazioni relativamente al sentiment delle notizie.

| Statistica | PolitiFact | GossipCop |
|---|---|---|
| #Notizie reali | 277 | 6945 |
| #Fake news | 351 | 3684 |
| #Utenti | 384,813 | 739,166 |
| #Tweets | 275,058 | 1,058,330 |
| #Retweets | 293,438 | 530,833 |
| #Risposte | 125,654 | 232,923 |

**Tabella 3.1:** Statistiche del dataset FakeNewsNet.

## 3.2 Struttura delle reti di propagazione

Le reti di propagazione hanno una struttura gerarchica, che comprende reti di propagazione di macro-livello e di micro-livello: le reti di propagazione a macro-livello mostrano il percorso di diffusione dalle notizie ai post sui social media che condividono la notizia e i re-post degli stessi. Le reti di macro-livello per le fake news si dimostrano più profonde, più estese e con un maggior numero di social bot rispetto a quelle delle notizie reali, il che fornisce indizi per individuare le fake news. D'altra parte, le reti di propagazione a micro-livello illustrano le conversazioni degli utenti sotto i post o i re-post, come le risposte e i commenti. Le reti di micro-livello rappresentano la rete di risposte in cui le informazioni sono condivise a livello locale. Le reti a livello macro rappresentano la propagazione globale delle informazioni in X attraverso una cascata di retweet. In figura Fig. 3.1 viene riportato un esempio di rete di propagazione relativa a una fake news. La rete di propagazione a livello macro include i nodi di notizie, i nodi di tweet e i nodi di retweet. La rete di propagazione a micro-livello indica l'albero di conversazione rappresentato dai nodi di risposta.

Per la rete di propagazione macro-livello, i nodi rappresentano i tweet e

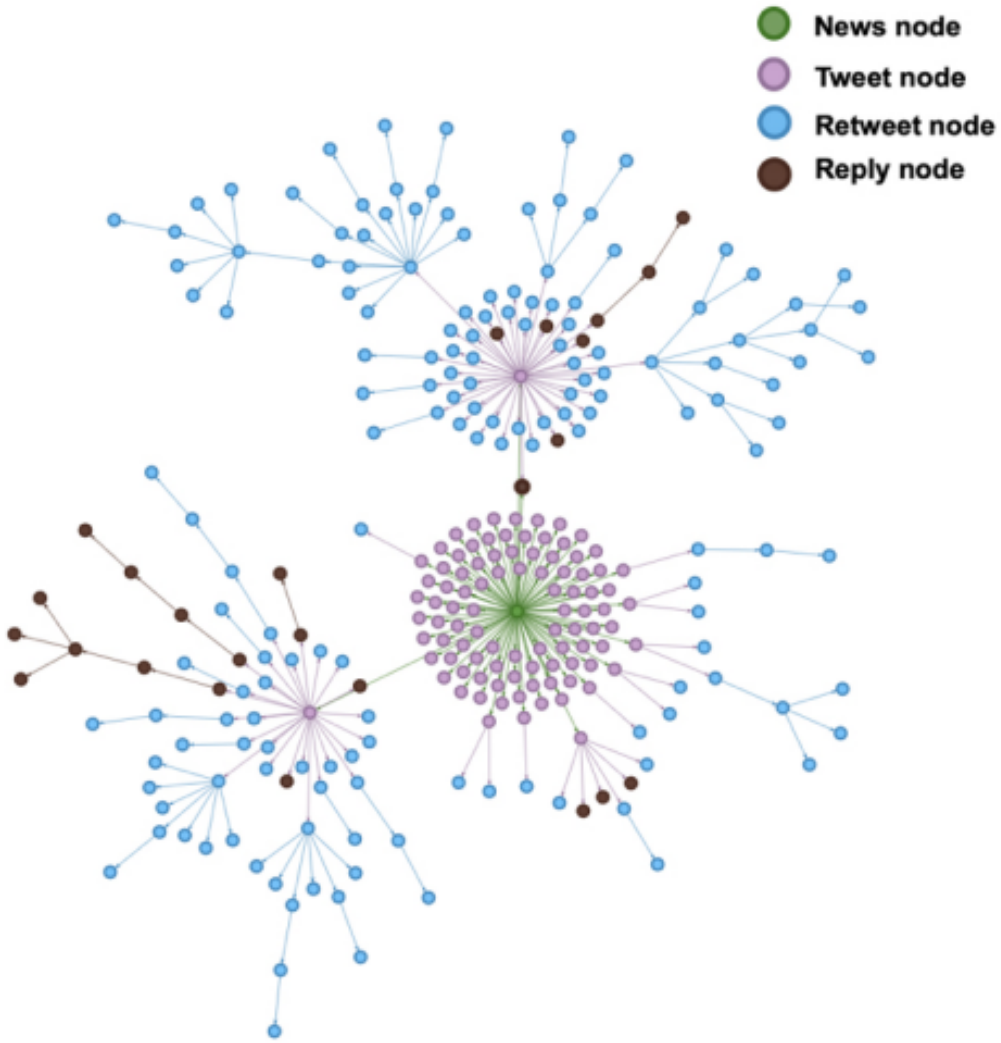



**Figura 3.1:** Esempio della rete di propagazione gerarchica di una fake news verificata da PolitiFact.

gli archi rappresentano la relazione di retweet tra di loro. In una rete macro, esiste un arco dal nodo $u$ a $v$ quando un tweet $u$ viene ritwittato da alcuni utenti $x$ e il nodo $v$ viene creato come risultato di questo evento. In X, un tweet o un retweet può essere ritwittato. Tuttavia, nei dati relativi ai retweet raccolti dall'API ufficiale di X, non vi è alcuna indicazione se le fonti ritwittate rappresentano un tweet originale o un retweet. Quindi la rete retweet non può essere esplicitamente realizzata a partire dai dati disponibili dall' API ufficiale di X. Per dedurre la fonte del retweet, si possono cercare di identificare i potenziali amici/followers dell'utente che hanno ritwittato il tweet. Per dedurre se il tweet è originale o è un re-post si va a confrontare il timestamp dello stesso con quello dei tweet/retweet degli amici: se il timestamp dei retweet dell'utente è maggiore del timestamp del retweet dell'amico dell'utente, allora l'utente deve aver visto il tweet di uno dei suoi amici e averlo ritwittato. Nel caso in cui non venga trovato il retweet immediato dell'amico di un utente, è possibile considerare che il retweet venga pubblicato a partire dal tweet originale anziché dal retweet di un altro retweet.

Per la rete di propagazione a micro-livello, i nodi rappresentano le risposte ai tweet in cui vengono pubblicate le notizie e gli archi rappresentano la relazione tra di loro. In X, un utente può rispondere al tweet o alla risposta di un altro utente. Se un utente risponde a un tweet originale, un arco collega il tweet e il nodo corrente, mentre se un utente risponde al commento di un altro utente, allora nel grafo di propagazione si viene a creare una catena di risposte (reply-chain).

## 3.3 Analisi della rete di propagazione di macro-livello

L'analisi viene condotta su aspetti strutturali e temporali, mentre l'analisi linguistica non è applicabile poiché le stesse informazioni testuali relative a una notizia sono condivise attraverso la rete di livello macro, per cui non fornirebbero informazioni aggiuntive.

*Analisi strutturale*: serve a capire il modello di diffusione globale delle notizie. Vengono caratterizzate e confrontate le reti di propagazione di macro-livello cercando varie caratteristiche di rete come segue.

- *(S1) Tree depth*: la profondità della rete di propagazione di macro-livello, cattura quanto le informazioni sono diffuse/ritwittate dagli utenti nei social media.

- *(S2) Numero di nodi*: il numero di nodi in una macro-rete indica il numero di utenti che condividono la notizia e può essere un segnale per comprendere il modello di diffusione.

- *(S3) Grado di uscita massimo*: grado massimo nella macro-rete potrebbe rivelare il tweet/retweet con la maggior influenza nel processo di propagazione.

- *(S4) Numero di cascate*: il numero di tweet originali che postano l'articolo originale.

- *(S5) Profondità del nodo con grado di uscita massimo*: la profondità che caratterizza il nodo con il massimo grado di uscita. Questo indica

i passi di propagazione necessari affinché un articolo sia diffuso da un nodo influente il cui post è ritwittato da più re-post di qualsiasi altro utente.

- *(S6) Numero di cascate con retweet*: indica numero di cascate (tweet) che sono state ritwittate almeno una volta.

- *(S7) Frazione di cascate con retweet*: indica la frazione di tweet con retweet tra tutte le cascate.

- *(S8) Numero di utenti bot che retwittano*: questa caratteristica cattura il numero di utenti bot che ritwitta le notizie corrispondenti.

- *(S9) Frazione di utenti bot che retwittano*: è il rapporto di utenti bot tra tutti gli utenti che twittano e ritwittano una notizia. Questa funzione può mostrare se le notizie hanno più probabilità di essere diffuse da bot o esseri umani reali.

*Analisi temporale*: l'impegno temporale dell'utente, inteso come la frequenza di interazioni in una rete macro-livello, rivela la frequenza e l'intensità del processo di diffusione delle notizie. La distribuzione della frequenza dei post degli utenti nel tempo può essere codificata in reti neurali ricorrenti (RNN - Recurrent neural network) per apprendere le caratteristiche per rilevare fake news. Tuttavia, le funzionalità apprese non sono interpretabili e la spiegazione del motivo per cui le funzionalità apprese possono aiutare rimane poco chiara. Di seguito sono riportate le caratteristiche temporali estratte dalla rete di propagazione macro:

- *(T1) Differenza media di tempo tra nodi di retweet adiacenti*: indica quanto velocemente i tweet sono ritwittati nel processo di diffusione delle notizie.

- *(T2) Differenza di tempo tra il primo tweet e gli ultimi retweet*: cattura la durata del processo di diffusione delle notizie.

- *(T3) Differenza di tempo tra il primo tweet e il tweet con il massimo grado di uscita*: i tweet con il massimo grado di uscita nella rete di propagazione rappresentano i nodi più influenti. Questa caratteristica dimostra quanto tempo ci è voluto affinché un articolo venisse ritwittato dal nodo più influente.

- *(T4) Differenza di tempo tra il primo e l'ultimo tweet relativi alla pubblicazione di notizie*: questo indica per quanto tempo i tweet relativi a un articolo sono pubblicati in X.

- *(T5) Differenza di tempo tra il tweet che pubblica notizie e l'ultimo nodo relativo a un retweet nella cascata più profonda*: l a cascata più profonda rappresenta la rete più propagata nell'intera rete di propagazione. Questa differenza di tempo indica la durata della notizia nella cascata più profonda e può mostrare se le notizie crescono in modo veloce o lento.

- *(T6) Differenza media di tempo tra nodi relativi a retweet adiacenti nella cascata più profonda*: questa caratteristica indica con quale frequenza un articolo di notizie è ritwittato nella cascata più profonda.

- *(T7) Tempo medio tra i tweet che pubblicano notizie*: questo tempo indica se i tweet relativi a una specifica notizia sono pubblicati in un breve intervallo di tempo.

- *(T8) Differenza di tempo media tra il tempo in cui viene postato un tweet e il tempo relativo al primo retweet*: la differenza di tempo media tra i primi tweet e il primo nodo di retweet in ogni cascata può indicare quanto presto i tweet vengono ritwittati.

## 3.4 Analisi della rete di propagazione di micro-livello

La rete di propagazione a livello micro descrive le opinioni e le emozioni degli utenti riguardo le notizie attraverso una catena di risposte nel tempo.

*Analisi strutturale*: l'analisi strutturale nella rete di micro-livello consiste nell'identificare modelli strutturali nei thread di conversazione degli utenti che esprimono il loro punto di vista sui tweet pubblicati in relazione alle notizie.

- *(S10) Profondità dell'albero*: cattura fino a che punto si estende l'albero di conversazione per i tweet/retweet che diffondono un articolo.
- *(S11) Numero di nodi*: il numero di nodi nella rete di propagazione micro-livello indica il numero di commenti che sono coinvolti. Può misurare quanto popolare sia il tweet del nodo radice.
- *(S12) Grado di uscita massimo*: nella micro-rete, il massimo grado di uscita indica il numero massimo di nuovi commenti nella catena a partire da un particolare nodo di risposta.
- *(S13) Numero di cascate nelle reti a micro-livello*: questa funzione indica il numero di cascate che hanno almeno una risposta.
- *(S14) Frazione di cascate nelle reti a micro-livello*: questa caratteristica indica la frazione delle cascate che hanno almeno una risposta tra tutte le cascate.

*Analisi temporale*: rappresenta le opinioni e le emozioni degli utenti attraverso una catena di risposte nel tempo. Di seguito sono riportate alcune delle caratteristiche estratte dalla rete di micro-propagazione:

- *(T9) Differenza media di tempo tra risposte adiacenti in cascata*: indica la frequenza con cui gli utenti si rispondono tra loro.
- *(T10) Differenza di tempo tra il primo tweet che pubblica la notizia e il primo nodo di risposta*: indica quanto velocemente la prima risposta viene pubblicata in risposta a un tweet che pubblica una notizia .
- *(T11) Differenza di tempo tra il primo tweet che pubblica la notizia e l'ultimo nodo di risposta nella micro-rete*: indica quanto lungo sia un albero di conversazione a partire dal tweet/retweet in cui viene pubblicata una notizia.
- *(T12) Differenza media di tempo tra le risposte nella cascata più profonda*: indica con che frequenza gli utenti si rispondono tra loro nella cascata più profonda.
- *(T13) Differenza di tempo tra il primo tweet che pubblica la notizia e l'ultimo nodo di risposta nella cascata più profonda*: indica la durata della conversazione nella cascata più profonda della micro-rete.

*Analisi linguistica*: le persone esprimono le proprie emozioni o opinioni nei confronti delle notizie false attraverso post sui social media, come opinioni scettiche o reazioni sensazionali. È stato dimostrato che queste informazioni testuali sono correlate al contenuto delle notizie originali.

- *(L1) Sentiment ratio*: si considera il rapporto tra il numero di risposte con sentiment positivo e il numero di risposte con sentiment negativo come una caratteristica per ogni notizia perché aiuta a capire se le notizie false ottengono più commenti positivi o negativi.

- *(L2) Sentiment medio*: punteggi medi del sentiment dei nodi nella rete di micro-propagazione. La feature L1 (sentiment ratio) non cattura la differenza relativa nei punteggi del sentiment e quindi si utilizza il sentiment medio.

- *(L3) Sentiment medio delle risposte di primo livello*: indica se le persone inviano commenti positivi o negativi sui post/tweet immediati che condividono notizie false e reali.

- *(L4) Sentiment medio delle risposte nella cascata più profonda*: la cascata più profonda generalmente indica i nodi che sono più propagati nell'intera rete di propagazione. Il sentiment medio delle risposte nella cascata più profonda cattura il sentiment dei commenti degli utenti nella cacata più influente.

- *(L5) Sentiment di risposta di primo livello nella cascata più profonda*: il sentiment della risposta di primo livello indica le emozioni dell'utente alla cascata di informazioni più influenti.

Definita $G_i$ la rete di propagazione della notizia $a_i$, vengono estratti tutti i tipi di features di propagazione e vengono concatenati in un unico vettore di features $f_i$. Questo vettore contiene sia le caratteristiche relative alle notizie vere che false e si utilizza il formato "pickle". Questo tipo di file viene usato

spesso per serializzare oggetti in Python, soprattutto se relativi a risultati di elaborazioni complesse o modelli di ML, come nel caso di studio. Infatti, una volta caricato il dataset, i vettori delle features vengono memorizzate in questi file pickle, in modo da poterli utilizzare all'occorrenza, una volta per l'addestramento del modello sul dataset di PolitiFact e una volta su quello di GossipCop. Prima dell'addestramento, i dati devono essere processati al fine di normalizzarli, in modo che le prestazioni dei classificatori siano le migliori possibili. Tramite la libreria scikit-learn è possibile scegliere la modalità di scaling. La scelta è ricaduta su "StandardScaler", una tecnica di pre-processing che trasforma le caratteristiche del dataset in modo tale che abbiano una media pari a zero e una deviazione standard pari a uno: in altre parole, si sceglie una distribuzione normale dei dati, in modo che siano comparabili. L'utilità della standardizzazione sta nel fatto che garantisce che tutte le caratteristiche contribuiscano equamente durante l'addestramento e previene che alcune caratteristiche dominino su altre a causa delle diverse scale di valori. Inoltre, molti algoritmi basati su gradienti, come la regressione logistica e le reti neurali, convergono più rapidamente se le caratteristiche sono standardizzate. Tipicamente lo scaling viene effettuato sui dati di training, in modo che si trovino i parametri per trasformare anche i dati di test e di validation. Questo assicura che i dati di test e validation siano trasformati nello stesso modo dei dati di training e che non influenzino il modello durante il training ("data leakage", fenomeno che può portare a sovrastimare le prestazioni del modello).

## 3.5 Metriche per la valutazione delle prestazioni di rilevamento delle fake news

Per valutare le prestazioni degli algoritmi di rilevamento delle notizie false, si usano le seguenti metriche: *Accuracy* (Acc), *Precision* (Prec), *Recall* (Rec) e *F1-score*.

- L'accuratezza è una misura generale della capacità di un modello di classificazione di classificare correttamente le istanze. È definita come il rapporto tra il numero di predizioni corrette (sia in termini di veri positivi che di falsi negativi) e il numero totale di predizioni effettuate dal modello. In altre parole, indica la percentuale di casi correttamente classificati rispetto al totale dei casi.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

- La precision è una misura della frazione di istanze predette come positive che sono realmente positive. È definita come il rapporto tra il numero di veri positivi e il numero totale di predizioni positive effettuate dal modello. La precision fornisce informazioni sulla qualità delle predizioni positive del modello.

$$Precision = \frac{TP}{TP + FP}$$

- La recall è una misura della frazione di istanze positive che il modello ha correttamente identificato. È definito come il rapporto tra il numero

di veri positivi e il numero totale di istanze positive presenti nel dataset. Fornisce informazioni sulla capacità del modello di trovare tutte le istanze positive.

$$Recall = \frac{TP}{TP + FN}$$

- L'F1-score è la media armonica tra precision e recall. È una metrica bilanciata che tiene conto sia della precisione che del richiamo. È particolarmente utile quando si desidera un'indicazione complessiva delle prestazioni del modello che tenga conto sia della qualità delle predizioni positive (precision) che della capacità di trovare tutte le istanze positive (recall). La media armonica è un tipo di media utilizzata in statistica che è particolarmente utile quando si vogliono combinare valori che hanno una relazione inversa tra loro. Mentre la media aritmetica è la media dei valori, la media armonica è il reciproco della media dei reciproci dei valori, per cui si tratta di una media ponderata che penalizza fortemente i valori più bassi tra precision e recall, perché se anche solo uno dei valori è basso, essa sarà più vicina a quel valore piuttosto che agli altri, anche se più alti. Questo significa che l'F1-score sarà maggiore solo se i valori di entrambe le metriche sono alti. In altre parole, se uno dei due è basso, l'F1-score sarà a sua volta basso, anche se l'altro valore è alto . L'F1-score raggiunge il suo valore massimo di 1 quando sia la precision che il recall sono al massimo, e il suo valore minimo di 0 quando sono entrambi minimi. Un valore alto di F1-score indica un buon bilanciamento tra precision e recall. È una metrica di valutazione particolarmente utile quando c'è una distribuzione sbilanciata tra le

classi.

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

Il processo viene eseguito per 5 volte e viene riportata la prestazione media, oltre che la deviazione standard delle diverse metriche.

## 3.6 Algoritmi di classificazione

Vengono testate le prestazioni di rilevamento del modello usando degli algoritmi di apprendimento:

1. Gaussian Naive Bayes (GNB): è un classificatore probabilistico basato sul Teorema di Bayes, che presuppone che le caratteristiche siano indipendenti dalla classe. Nonostante questa forte assunzione di indipendenza, i classificatori di Naive Bayes ottengono risultati notevoli in molte applicazioni reali, soprattutto per la classificazione dei testi. Quando si tratta di dati continui, il modello Gaussiano di Naive Bayes presuppone che i dati siano normalmente distribuiti. È un modello semplice ma potente, che funziona bene con piccoli insiemi di dati ed è efficiente dal punto di vista computazionale.

2. Decision Tree (DT): è un algoritmo di apprendimento supervisionato utilizzato per prendere decisioni basate su caratteristiche di input. Modella le decisioni e le loro possibili conseguenze come una struttura ad albero composta da nodi, rami e foglie. Ogni nodo interno rappresenta un "test" o una "decisione" su un attributo, ogni ramo rappresenta il risultato del test e ogni nodo foglia rappresenta un'etichetta di classe. Il modello è facile da interpretare e da visualizzare, il che lo rende particolarmente utile per ottenere approfondimenti sul processo decisionale di modelli complessi. Tuttavia, gli alberi decisionali possono essere inclini all'overfitting, soprattutto con insiemi di dati di piccole dimensioni. L'overfitting è quel fenomeno per cui un classificatore si

specializza eccessivamente sui dati di addestramento e di fronte a dati mai visti prima risulta inefficace.

3. Logistic Regression (LR): è un modello lineare utilizzato per compiti di classificazione binaria. Modella la probabilità che un dato input appartenga a una particolare classe utilizzando una funzione logistica. La LR è adatta ai casi in cui la relazione tra le caratteristiche di input e le probabilità di classe è approssimativamente lineare. È facile da implementare e da interpretare.

4. Random Forest (RF): è un metodo di apprendimento collettivo che combina più alberi decisionali durante l'addestramento per migliorare l'accuratezza della classificazione e controllare l'overfitting comunemente associato ai singoli alberi decisionali. Ogni albero della foresta è costruito a partire da un sottoinsieme casuale dei dati e delle caratteristiche di addestramento, promuovendo la diversità tra gli alberi e migliorando la capacità di generalizzazione del modello.

5. Support Vector Machine (SVM): è un potente metodo di classificazione che cerca di trovare l'iperpiano che meglio separa le classi nello spazio delle caratteristiche. Funziona bene sia con i dati lineari che con quelli non lineari, utilizzando le funzioni kernel per mappare le caratteristiche in ingresso in spazi di dimensioni superiori in cui è possibile trovare un separatore lineare.

6. Rete Neurale Feedforward (FNN): è un tipo di rete neurale artificiale in cui le connessioni tra i nodi non formano un ciclo. Nel contesto della

classificazione, le FNN sono costituite da un layer di ingresso, uno o più layers nascosti e un layer di uscita. La peculiarità delle reti di questo tipo è che l'informazione viaggia in una sola direzione, dal layer di input verso quello di output, passando per quelli intermedi. Ogni neurone di un layer è collegato a tutti i neuroni del layer successivo, per cui rientra nel campo delle reti fully-connected. Le FNN sono in grado di modellare complesse relazioni non lineari e sono molto versatili, il che le rende adatte a un'ampia gamma di compiti di classificazione.

Per l'implementazione dei classificatori, all'interno del progetto originale, viene utilizzato il framework *scikit-learn* di Python e per questo motivo, anche la rete neurale Feedforward viene implementata utilizzando un modulo di tale libreria: *MLPClassifier*. MLP sta per "Multi-layer Perceptron", un algoritmo di apprendimento supervisionato che apprende una funzione $f : R^m \rightarrow R^o$ addestrandosi su un set di dati, dove $m$ è il numero di dimensioni per input e $o$ è il numero di dimensioni per l'output. MLPClassifier esegue il training su due array: array X di dimensione (n_samples, n_features), che contiene i campioni di training rappresentati come vettori di features e l'array Y di dimensione (n_samples,), che contiene i valori target (etichette di classe) per i campioni di addestramento. La struttura di una FNN implementata con questo modulo prevede che per il layer di input il numero di neuroni corrisponde al numero di features nel dataset di input (nel caso di studio il vettore delle features è composto da 32 elementi), mentre per il layer di output il numero di neuroni corrisponde al numero di classi target (real o fake nel caso di studio). Il numero di neuroni per ognuno dei layer nascosti è un iperparametro che può essere specificato. Come funzione di attivazione

da applicare ai neuroni nei layer intermedi, si può scegliere tra ReLU, Tanh e Logistic Sigmoid, mentre per quello di output è supportata la Softmax per problemi di classificazione multi-classe, e per quelli di classificazione binaria (come quello del caso di studio) la Logistic Sigmoid. Per il layer di output, però, la libreria non consente di specificare la funzione di attivazione, bensì la gestisce autonomamente in base al tipo di problema. Come funzione di perdita, MLPClassifier supporta solo la Cross-Entropy Loss. Quest'ultima misura la discrepanza tra la distribuzione delle probabilità previste dal modello e la distribuzione delle probabilità reali delle classi. Più precisamente, per ogni esempio di input, la Cross-Entropy Loss è data dalla seguente formula:

$$L(y, \hat{y}) = -\frac{1}{N} \sum_{i=1}^{N} (y_i \cdot \log(\hat{y}_i) + (1 - y_i) \cdot \log(1 - \hat{y}_i))$$

Dove:

- $N$ è il numero totale di esempi nel set di dati.
- $y_i$ è l'etichetta di classe reale per l'esempio $i$ (1 per la classe positiva, 0 per la classe negativa).
- $\hat{y}_i$ è la probabilità prevista dal modello che l'esempio $i$ appartenga alla classe positiva.

Questo modulo sfrutta la regolarizzazione L2, regolata tramite il parametro *alpha*, per prevenire il fenomeno dell'overfitting penalizzando i pesi con grandezze elevate. Il parametro *alpha* controlla il trade-off tra la minimizzazione del costo dell'addestramento e la penalizzazione dei coefficienti più

grandi. Tipicamente, un valore di *alpha* più grande porta a una maggiore regolarizzazione.

## 3.7 Selezione degli iperparametri

L'analisi originale condotta nel paper di riferimento, è stata realizzata utilizzando i parametri di default di ogni classificatore. Per il tuning degli iperparametri in questa tesi, invece, viene utilizzato il modulo *GridSearch-CV* di scikit-learn, con l'obiettivo di migliorare i risultati di partenza. Questo modulo opera sistematicamente attraverso diverse combinazioni di parametri, effettua una *cross validation* di ogni combinazione e seleziona il set di parametri che produce le migliori prestazioni. Fare cross validation significa dividere il dataset in k porzioni di uguale dimensione, per poi fare training sui primi k-1 elementi e test sull'ultima porzione. La cross validation è una delle tecniche più utilizzate per mitigare il rischio di overfitting. Di seguito sono riportati i valori utilizzati per effettuare il tuning degli hyperparameters di ognuno dei classificatori.

| **Iperparametro** | **Valori** |
|---|---|
| var_smoothing | 1e-9 |

**Tabella 3.2:** Valori degli iperparametri valutati per il classificatore Gaussian Naive Bayes.

Per GaussianNB, si fissa a 1e-9 il valore del parametro var_smoothing, è utilizzato per stabilizzare le stime della varianza dei predittori.

| **Iperparametro** | **Valori** |
|---|---|
| C | [0.001, 0.01, 0.1, 1, 10, 100] |
| class_weight | [None, balanced] |
| max_iter | [500] |

**Tabella 3.3:** Valori degli iperparametri valutati per il classificatore Logistic Regression.

Per LR si valutano diverse opzioni per il parametro C, che serve a controllare la complessità del modello e viene interpretato come l'inverso della forza di regolarizzazione. Inoltre, si offre la possibilità di bilanciare automaticamente le classi durante l'addestramento, oppure di lasciarle inalterate. Il parametro max_iter specifica il numero massimo di iterazioni durante l'ottimizzazione.

| Iperparametro | Valori |
|---|---|
| criterion | [gini, entropy] |
| max_depth | [None, 10, 20, 30, 40, 50] |
| max_features | [None, sqrt, log2] |

**Tabella 3.4:** Valori degli iperparametri valutati per il classificatore Decision Tree.

Per quanto riguarda DT il parametro criterion, specifico delle strutture ad albero, specifica la funzione per misurare la qualità di uno split. max_depth indica la profondità massima dell'albero. Se None, i nodi vengono espansi fino a quando tutte le foglie sono pure o fino a quando tutte le foglie contengono meno di min_samples_split samples, il cui valore di default è 2. Per il parametro max_features, che indica il numero massimo di caratteristiche da considerare per la migliore divisione, si specificano tre valori: per None il numero massimo di features è il numero di features originale (n_features), per sqrt è $sqrt$(n_features) e per log2 è $log_2$(n_features).

| Iperparametro | Valori |
|---|---|
| criterion | [gini, entropy] |
| max_depth | [None, 10, 20, 30, 40, 50] |
| max_features | [None, sqrt, log2] |
| n_estimators | [10, 20, 50, 100] |

**Tabella 3.5:** Valori degli iperparametri valutati per il classificatore Random Forest.

Il classificatore RF addestra un certo numero di alberi decisionali, per cui i parametri sono gli stessi del DT con l'aggiunta di quello relativo al numero di alberi da includere nella foresta n_estimators.

| Iperparametro | Valori |
|---|---|
| C | [0.1, 1, 10, 100, 1000] |
| kernel | [linear] |
| max_iter | [2000] |

**Tabella 3.6:** Valori degli iperparametri valutati per il classificatore Support Vector Machine.

Per SVM il parametro C funziona come nella LR, kernel viene fissato a linear e max_iter è il numero massimo di iterazioni affinché l'algoritmo converga.

| Iperparametro | Valori |
|---|---|
| hidden_layer_sizes | [[10], [20], [10,10], [20,10], [50], [10, 10, 10], [20, 20, 20], [50, 50, 50], [10, 20], [20, 10, 20], [50, 20, 10], [100], [100, 50], [100, 50, 25], [10, 50, 10], [50, 10, 50], [100, 50, 10]] |
| activation | [relu, tanh, logistic] |
| solver | [adam, lbfgs, sgd] |
| learning_rate | [constant, adaptive] |
| max_iter | [2000] |
| alpha | [0.0001, 0.001, 0.01] |
| batch_size | [8, 16, 32, 64] |
| early_stopping | [True] |
| n_iter_no_change | [15] |

**Tabella 3.7:** Valori degli iperparametri valutati per la Feedforward Neural Network.

Il primo iperparametro che viene valutato per la FNN è quello relativo alla dimensione dei layers nascosti. Vengono selezionati una serie di possibili valori che rappresentano il numero di neuroni per ognuno dei livelli inter-

medi. La funzione di attivazione di una rete neurale è la funzione che il neurone applica alla somma pesata per fare predizione e serve a introdurre non-linearità nel modello, consentendo alla rete di apprendere e modellare relazioni complesse tra i dati di input e output. Vengono valutate:

- Logistic Sigmoid: ha una forma a "S" e produce un valore compreso tra 0 e 1 che risulta molto utile per modellare probabilità. Tende a saturare rapidamente agli estremi.

- Tanh (Tangente Iperbolica): mappa gli input a un intervallo tra -1 e 1, con una sigmoide simmetrica rispetto all'origine. Essendo le uscite centrate attorno allo zero, la convergenza viene facilitata durante l'addestramento. Il problema è che, come la Logistic Sigmoid, agli estremi satura rapidamente.

- ReLU (Rectified Linear Unit): è una delle funzioni di attivazione più popolari per la sua semplicità computazionale, che rende il processo di addestramento più veloce. Questa funzione mappa tutte le entrate negative a zero e lascia invariati tutti i valori positivi.

Il campo solver è quello che in una rete neurale prende il nome di algoritmo di ottimizzazione, usato per aggiornare il valore dei pesi della rete durante il processo di addestramento al fine di minimizzare la funzione di perdita (MLPClassifier utilizza la backpropagation per aggiornare i pesi). Vengono conisderati:

- SGD (Stochastic Gradient Descent): è fondamentale per l'addestramento delle reti neurali. Si tratta di un algoritmo di ottimizzazione

utilizzato per minimizzare la funzione di costo iterativamente. L'algoritmo aggiorna i pesi della rete neurale basandosi sul gradiente della funzione di costo calcolato su un singolo esempio o su un piccolo batch di esempi.

- Adam (Adaptive Moment Estimation): combina i vantaggi di altri due ottimizzatori, che sono RMSprop e Momentum. Il primo effettua una normalizzazione con la somma dei quadrati dei gradienti, mentre il secondo è una variante di SGD che accumula il gradiente di iterazioni precedenti per accelerare la discesa lungo i gradienti che puntano nella stessa direzione.

- LBFGS (Limited-memory Broyden–Fletcher–Goldfarb–Shanno): viene usato per risolvere problemi di minimizzazione non lineare.

Il learning rate (o tasso di apprendimento) è un iperparametro fondamentale utilizzato durante l'ottimizzazione dei pesi della rete, poiché determina quanto i pesi della rete vengono aggiornati in base al gradiente calcolato durante la fase di backpropagation. In tal caso, vengono scelti due valori: il primo mantiene costante il tasso di apprendimento durante tutta la fase di addestramento e il secondo serve per adattarlo in base alla performance del modello, andando a diminuirne il valore quando il modello sta per convergere, in modo da evitare che si blocchi in punti di minimo locale. Il parametro "max_iter" indica il numero massimo di epoche (scansioni del training set) che la rete deve deve eseguire durante la fase di training. Si tratta di un iperparametro molto importante, in quanto se l'ottimizzazione non converge entro questo numero di iterazioni, l'addestramento viene interrotto. L'am-

piezza del batch indica il numero di campioni da utilizzare in ogni iterazione di addestramento. Impostando a *True* il parametro *early stopping*, si consente di interrompere prematuramente l'addestramento quando le prestazioni del modello su un validation set non migliorano per un certo numero di iterazioni consecutive. In questo modo, si previene l'overfitting e si riduce il tempo di addestramento. È direttamente collegato all'iperparametro successivo n_iter_no_change, in quanto quest'ultimo specifica proprio il numero di iterazioni consecutive da considerare per l'early stopping.

## 3.8 Feature importance analysis

Una volta ottenuti i migliori parametri per ciascun classificatore, è stata valutata l'importanza delle caratteristiche considerate, utilizzando il modulo *Permutation feature importance* di scikit-learn. Si tratta di una tecnica che valuta l'importanza delle features permutando casualmente i valori di una singola feature e osservando quanto ciò influisce sulle prestazioni del modello, mantenendo invariati gli altri valori. In altre parole, si effettua uno shuffle randomico dei valori di una singola caratteristica rispetto alle etichette corrispondenti. Il modello viene nuovamente valutato utilizzando il set di dati di test modificato. Questo calcolo delle prestazioni viene ripetuto per tutte le caratteristiche. Questa tecnica viene utilizzata perché assicura che l'importanza delle caratteristiche sia calcolata in modo robusto attraverso più iterazioni, mitigando l'effetto di eventuali variazioni nei risultati dovute a particolari suddivisioni dei dati di training e di test. L'importanza di ogni caratteristica è calcolata confrontando le prestazioni del modello prima e dopo la permutazione della caratteristica stessa. Una feature è considerata importante se il suo shuffle porta a una diminuzione significativa delle prestazioni del modello, indicando che il modello dipende fortemente da quella caratteristica. Al contrario, se la permutazione non influisce sulle prestazioni, la caratteristica potrebbe essere meno importante. Di seguito è proposto l'algoritmo utilizzato per il calcolo dell'importanza delle features.

**Inputs:**

- Modello predittivo addestrato $m$
- Dataset in formato tabellare (training o validation) $D$

**Steps:**

1. Calcolare lo score $s$ del modello $m$ sui dati $D$.
2. Per ogni feature $j$ (colonna di $D$):
   (a) Per ogni ripetizione $k$ in $1, \ldots, K$:
      - Effettuare uno shuffle randomico della colonna $j$ del dataset $D$ per generare una versione corrotta dei dati denominata $\tilde{D}_{k,j}$.
      - Calcolare lo score di $s_{k,j}$ del modello $m$ sui dati alterati $\tilde{D}_{k,j}$.

   (b) Calcolare l'importanza $i_j$ della feature $f_j$ definita come:

$$i_j = s - \frac{1}{K} \sum_{k=1}^{K} s_{k,j}$$

L'importanza della caratteristica viene dunque determinata confrontando le performances del modello con e senza la permutazione della caratteristica. La differenza in termini di prestazioni indica quanto il modello dipende da quella caratteristica. Un valore positivo di importanza per una feature indica che quella feature è utile per il modello. Più alto è il valore, maggiore è l'importanza della feature. Un valore positivo implica dunque che la permutazione (ossia lo scambio dei valori tra le osservazioni) di questa feature peggiora le performance del modello rispetto alla sua baseline, il che significa che la feature contiene informazioni preziose che aiutano il modello a fare previsioni accurate. Un valore negativo di importanza per una feature indica che la permutazione di questa feature migliora effettivamente le performances del modello rispetto alla baseline. Questo suggerisce che la caratteristica potrebbe introdurre rumore o essere ingannevole per il modello.

Tuttavia, è importante notare che l'importanza delle caratteristiche calcolata con questo metodo è relativa al modello specifico e ai dati utilizzati per l'addestramento e per il test.

# Capitolo 4

# Risultati

In questo capitolo vengono presentati e discussi i risultati emersi al termine di entrambi gli esperimenti condotti in questo lavoro di tesi.

## 4.1 Grafo bipartito

In questa sezione sono riportati i risultati ottenuti dall'analisi del grafo bipartito e delle relative proiezioni prodotti nel capitolo 3. È opportuno ricordare il contenuto del dataset di partenza, un file CSV chiamato "archive_with_keywords.csv", che contiene i risultati relativi alle prime 25 pagine delle ricerche effettuate su PolitiFact per sei diverse keywords: "Ukraine", "Russia", "Zelensky", "Putin", "Trump" e "Biden". In totale, ci sono 652 record.

Una rappresentazione del grafo bipartito è riportata in Fig. 4.1, dove per consentire una migliore visualizzazione sono stati etichettati esclusivamente

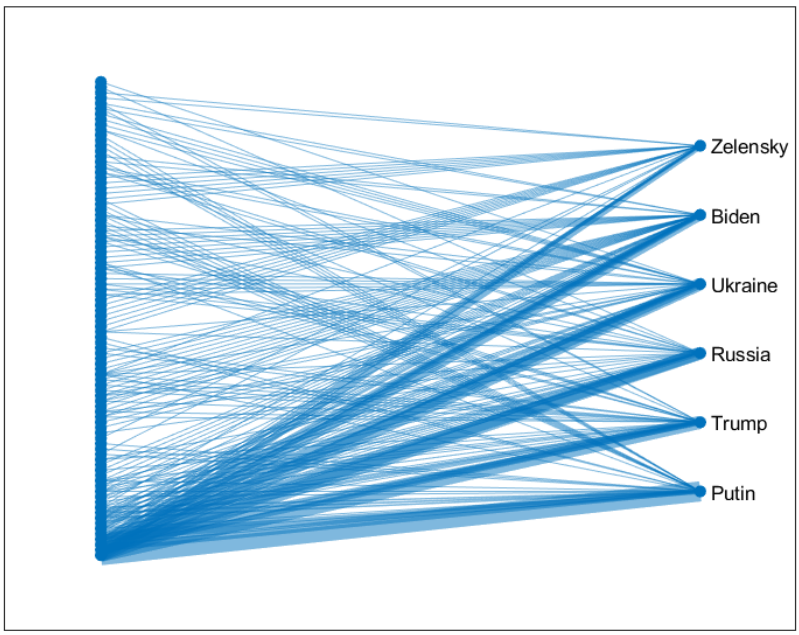



**Figura 4.1:** Rappresentazione del grafo bipartito con etichettatura dei soli nodi delle keywords.

i nodi relativi alle parole chiave, mentre quelli relativi agli speaker, essendo in numero maggiori non sono stati etichettati.

Per comprendere il significato del grafo in questione, per ognuna delle parole chiave scelte per effettuare le ricerche sono stati prodotti tre barplot:

1. Un barplot che sulle ascisse riporta tutti gli speaker che hanno pubblicato notizie relative a quella parola chiave e sulle ordinate il relativo grado di fakeness calcolato con il peso normalizzato. Gli speaker sono ordinati per valori di peso decrescenti. Questo ordinamento viene mantenuto per gli altri due grafici di ogni parola chiave.

2. Un barplot che sulle ascisse riporta tutti gli speaker che hanno pubbli-

cato notizie relative a quella parola chiave, ordinati per valori di peso normalizzato decrescenti, e sulle ordinate il relativo grado di fakeness calcolato con l'equazione 2.1.

3. Un barplot che sulle ascisse riporta tutti gli speaker che hanno pubblicato notizie relative a quella parola chiave, ordinati per valori di peso normalizzato decrescenti, e sulle ordinate il numero di notizie pubblicate da quegli speaker per quella keyword.

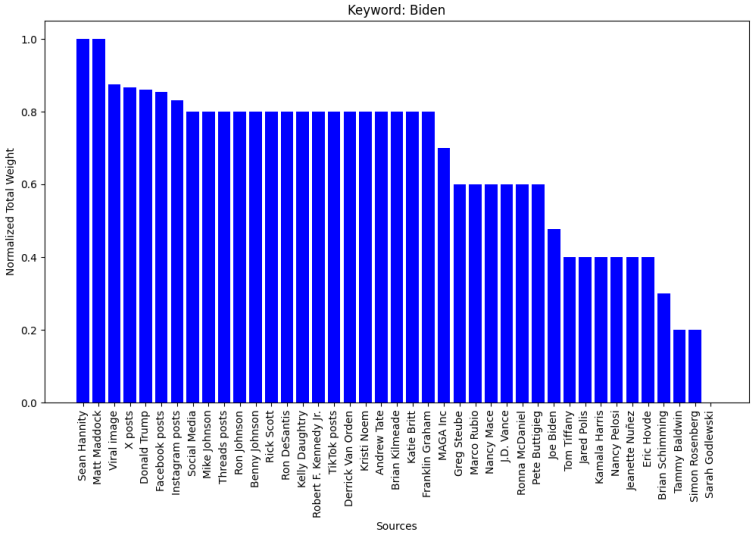



**Figura 4.2:** Barplot dei pesi normalizzati per gli speaker che hanno pubblicato notizie relative alla keyword "Biden".

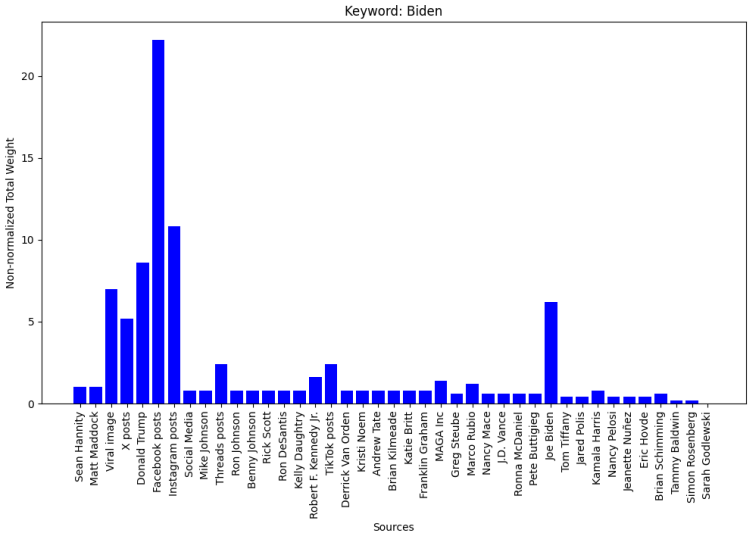



**Figura 4.3:** Barplot dei pesi originali per gli speaker che hanno pubblicato notizie relative alla keyword "Biden".

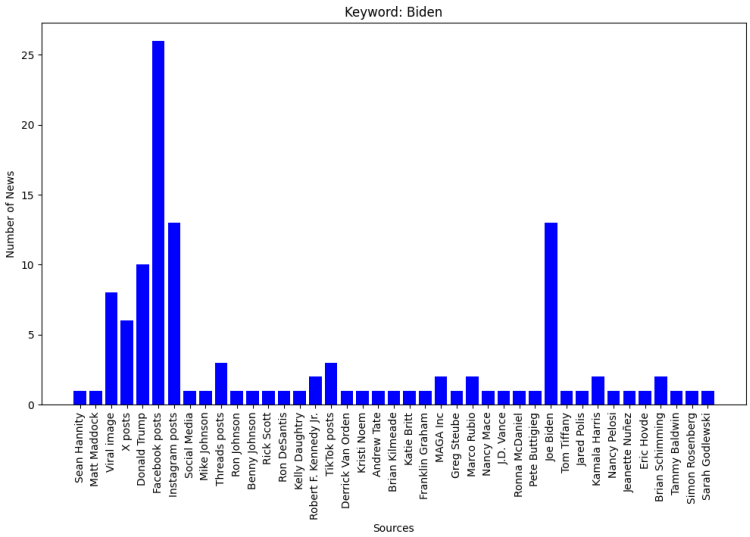



**Figura 4.4:** Barplot relativo al numero di notizie pubblicate dagli speaker per la keyword "Biden".

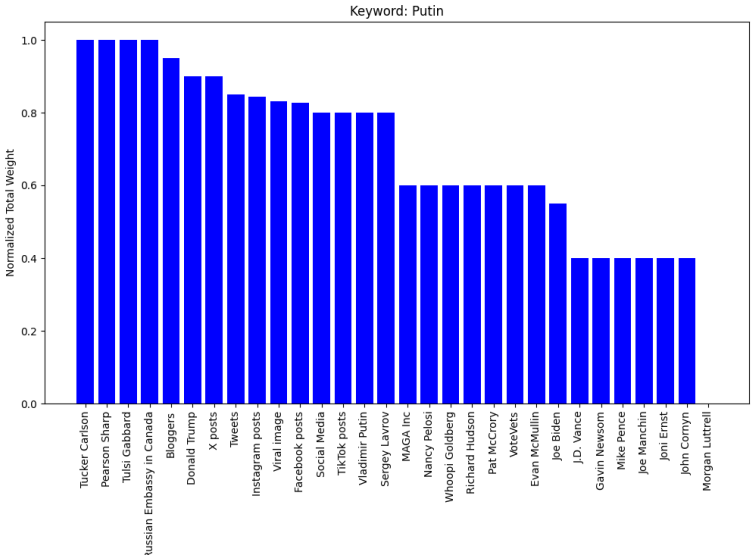



**Figura 4.5:** Barplot dei pesi normalizzati per gli speaker che hanno pubblicato notizie relative alla keyword "Putin."

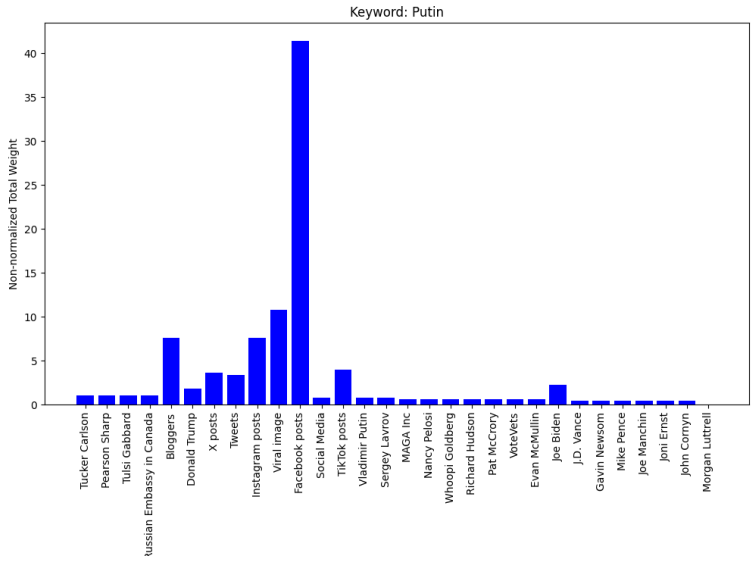



**Figura 4.6:** Barplot dei pesi originali per gli speaker che hanno pubblicato notizie relative alla keyword "Putin".

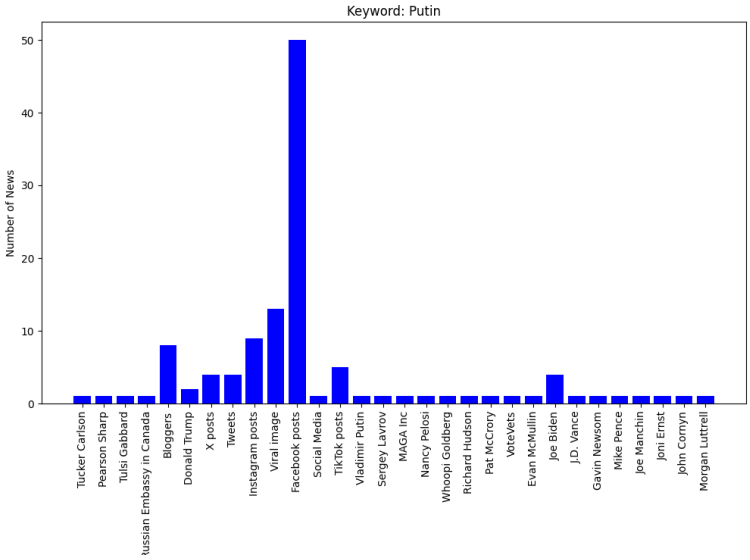



**Figura 4.7:** Barplot relativo al numero di notizie pubblicate dagli speaker per la keyword "Putin".

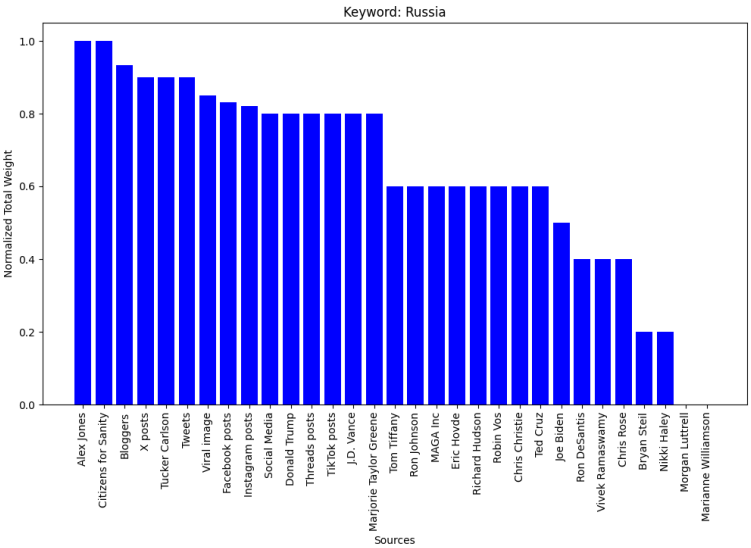



**Figura 4.8:** Barplot dei pesi normalizzati per gli speaker che hanno pubblicato notizie relative alla keyword "Russia."

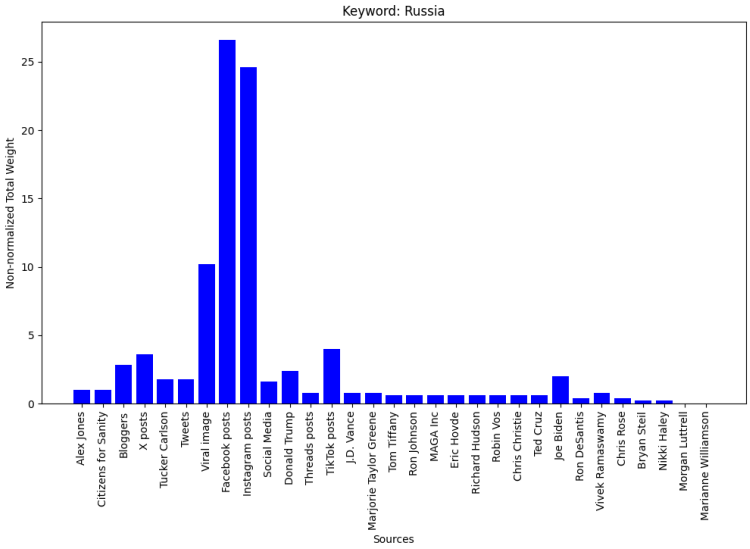



**Figura 4.9:** Barplot dei pesi originali per gli speaker che hanno pubblicato notizie relative alla keyword "Russia".

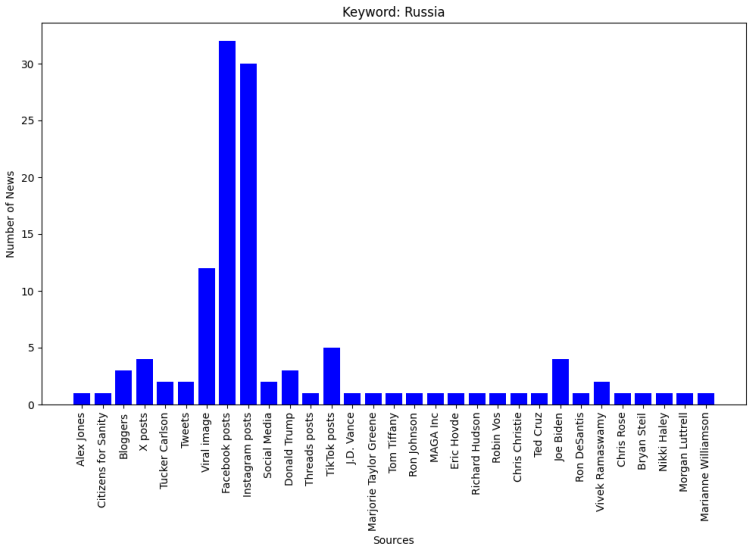



**Figura 4.10:** Barplot relativo al numero di notizie pubblicate dagli speaker per la keyword "Russia".

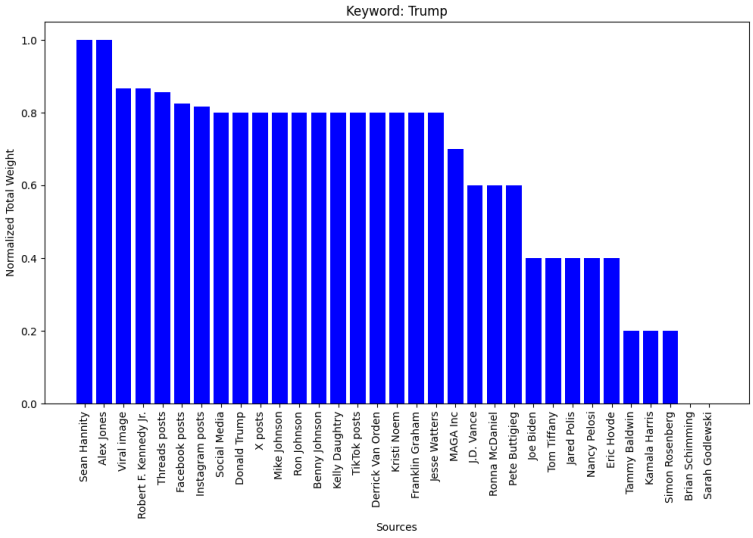



**Figura 4.11:** Barplot dei pesi normalizzati per gli speaker che hanno pubblicato notizie relative alla keyword "Trump."

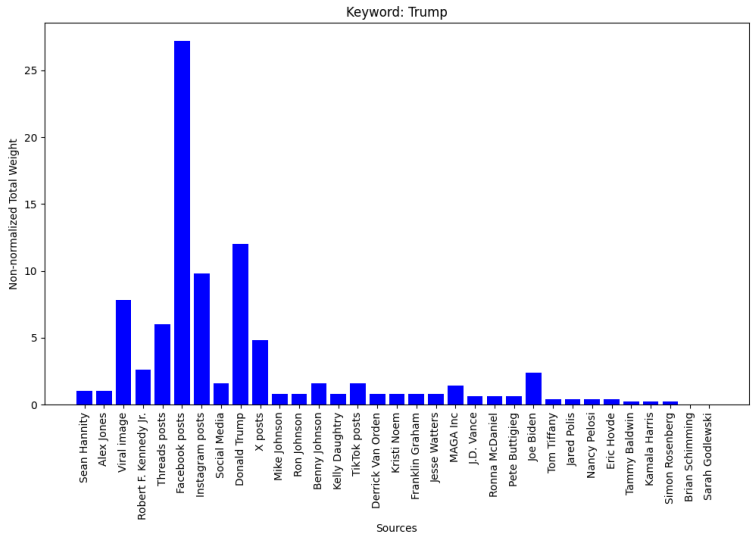



**Figura 4.12:** Barplot dei pesi originali per gli speaker che hanno pubblicato notizie relative alla keyword "Trump".

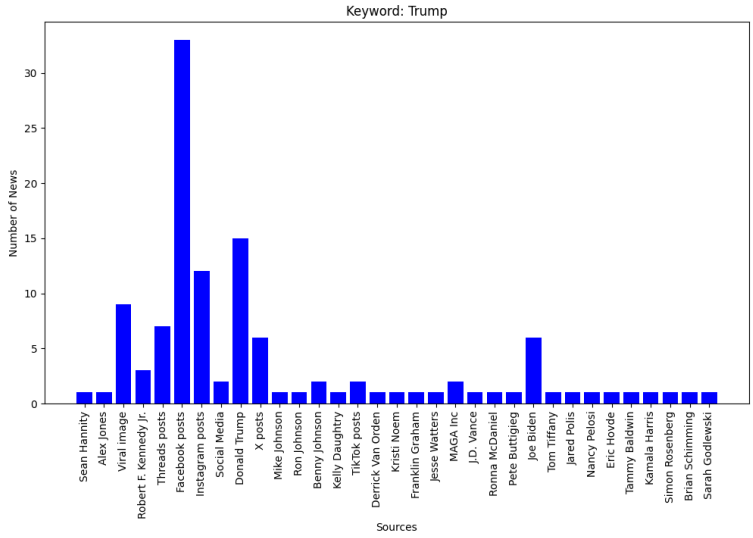



**Figura 4.13:** Barplot relativo al numero di notizie pubblicate dagli speaker per la keyword "Trump".

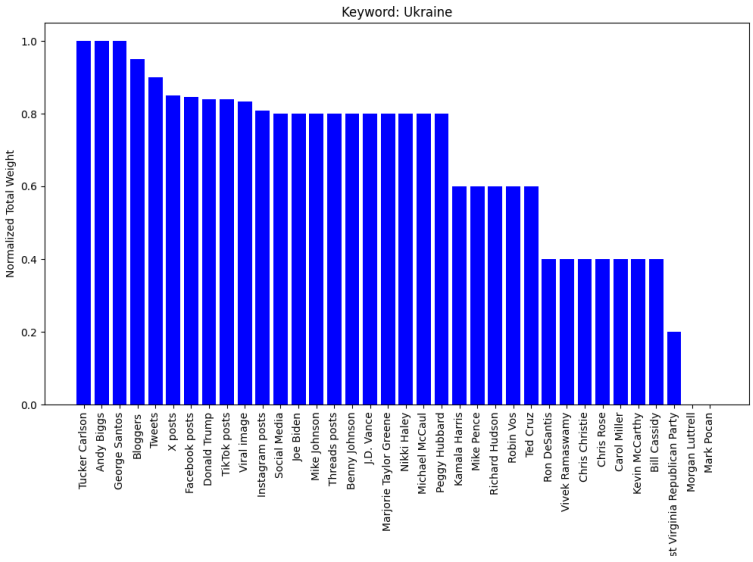



**Figura 4.14:** Barplot dei pesi normalizzati per gli speaker che hanno pubblicato notizie relative alla keyword "Ukraine."

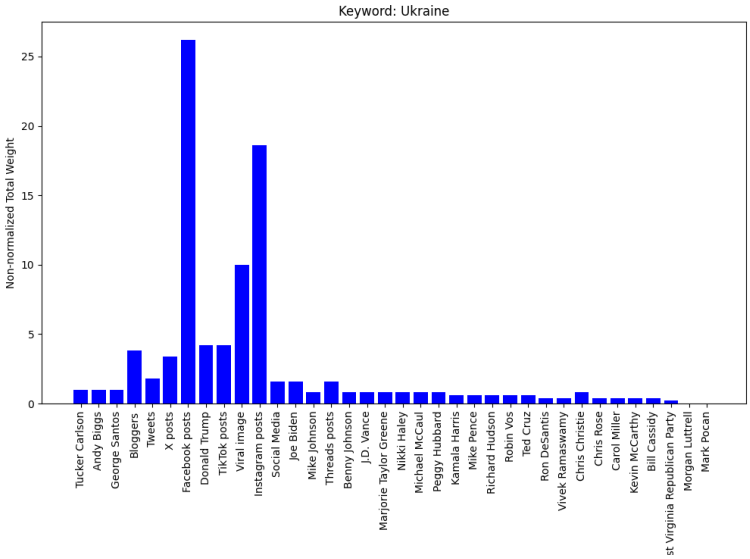



**Figura 4.15:** Barplot dei pesi originali per gli speaker che hanno pubblicato notizie relative alla keyword "Ukraine".

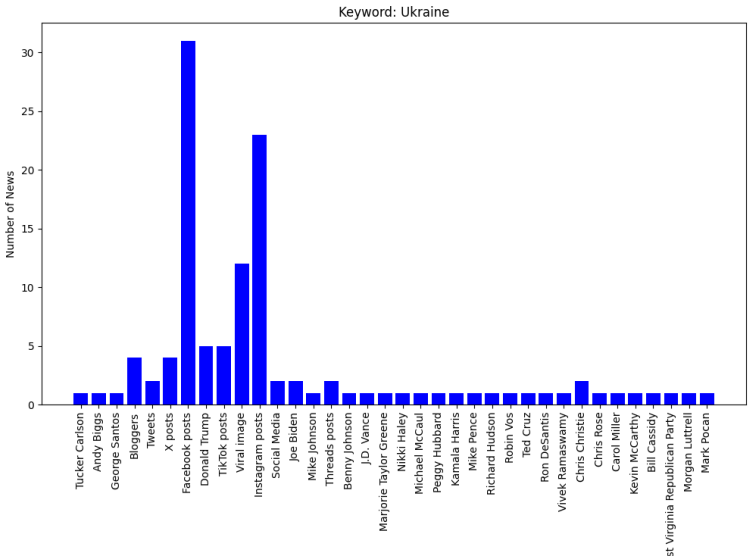



**Figura 4.16:** Barplot relativo al numero di notizie pubblicate dagli speaker per la keyword "Ukraine".

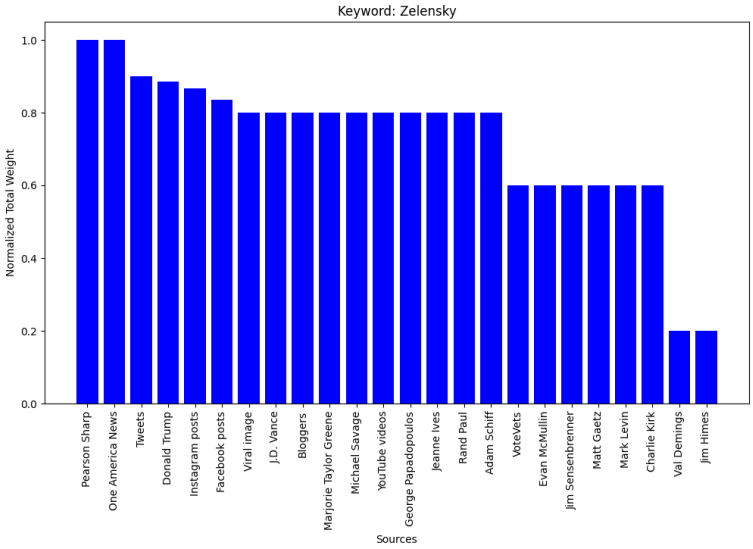



**Figura 4.17:** Barplot dei pesi normalizzati per gli speaker che hanno pubblicato notizie relative alla keyword "Zelensky."

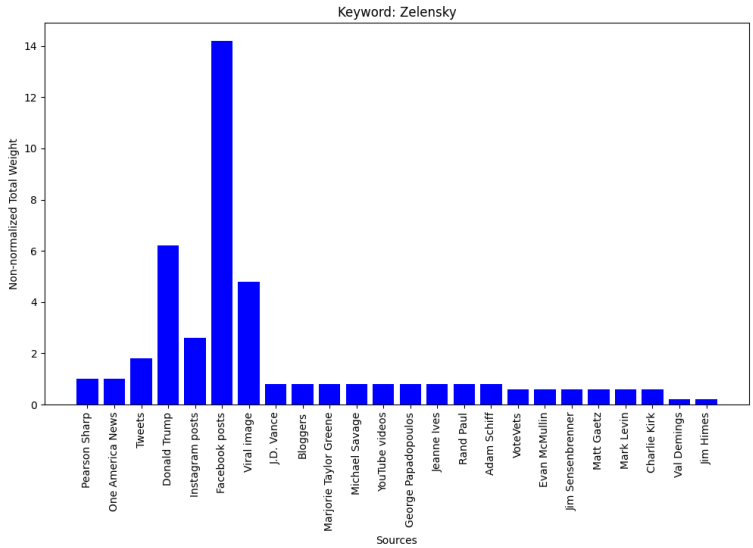



**Figura 4.18:** Barplot dei pesi originali per gli speaker che hanno pubblicato notizie relative alla keyword "Zelensky".

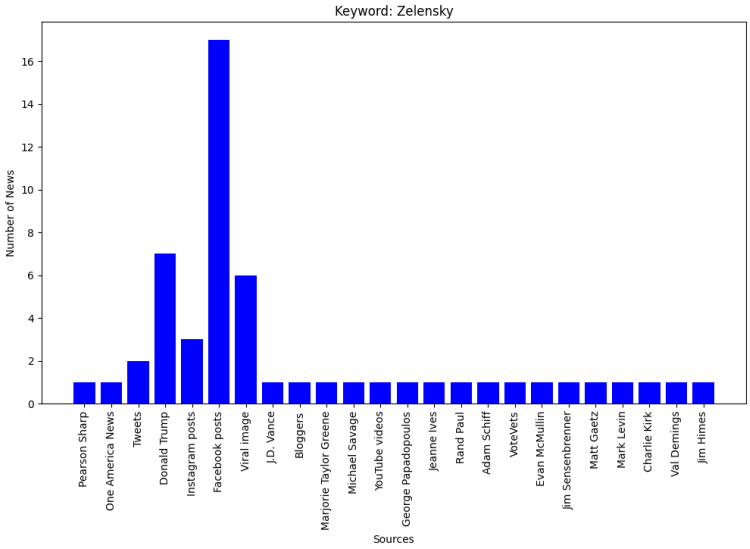



**Figura 4.19:** Barplot relativo al numero di notizie pubblicate dagli speaker per la keyword "Zelensky".

L'esigenza di realizzare questi grafici nasce dal fatto che per effettuare delle considerazioni significative è necessario considerare il numero di notizie pubblicate da ognuno degli speaker in riferimento alla keyword considerata. Andando a normalizzare il grado di fakeness in base al numero di notizie pubblicate da uno speaker per ogni keyword, infatti, si ottiene un'indicazione sulla percentuale di notizie false pubblicate da ciascuno degli speakers, perdendo però l'informazione riguardo il numero di notizie da esso pubblicate. È interessante notare i casi in cui si registrano valori elevati sia per il peso normalizzato che quello non normalizzato associati a uno speaker che ha pubblicato un numero elevato di notizie. In questo modo, infatti, si ha un'indicazione sulle sorgenti per le quali si registra un alto tasso di pubblicazione di notizie false. Osservando i risultati ottenuti si può affermare che in base agli argomenti scelti, i social media risultano in generale le sorgenti

con il numero più alto di notizie pubblicate. In questo modo, però, emergono anche speaker che non risultano essere diffusori di notizie con un grado di fakeness elevato, per cui è possibile effettuare una prima classificazione delle notizie pubblicate dai diversi speakers per le keywords considerate.

Di seguito sono riportati i risultati e le relative considerazioni riguardo le proiezioni del grafo bipartito nello spazio delle sorgenti e delle keywords, considerando la casistica in cui il peso originale non è stato normalizzato. In Fig. 4.20 viene rappresentato il grafo relativo alla proiezione del grafo bipartito sulle sorgenti. L'aspetto interessante che viene fuori osservando la figura è una forma di clusterizzazione degli speakers. La vicinanza di alcuni nodi, in questo caso, indica che da quelle sorgenti è stato pubblicato un numero elevato di notizie relative agli stessi argomenti (in termini di keywords in comune).

Inoltre, vengono prodotti dei barplot relativi alle misure di nodo, considerando i nodi con i valori più alti. Il grado di un nodo del grafo rappresenta il numero di notizie pubblicate in comune con gli altri speaker, pesato sul grado di fakeness delle stesse. Dunque, un valore alto in termini di grado potrebbe indicare una sorgente che tende a diffondere un alto numero di notizie su argomenti di tendenza, in quanto tanti altri soggetti hanno diffuso notizie su quell'argomento, indipendentemente dal grado di fakeness delle stesse. L'informazione relativa a quest'ultimo aspetto non può essere direttamente determinata senza un'analisi dedicata, in quanto il grado di un nodo tiene conto sia del grado di fakeness, sia del numero di notizie pubbblicate, dunque si potrebbe anche trattare di molte notizie con un grado di fakeness non troppo alto. In Fig. 4.21 sono stati riportati gli speakers con i valori di grado

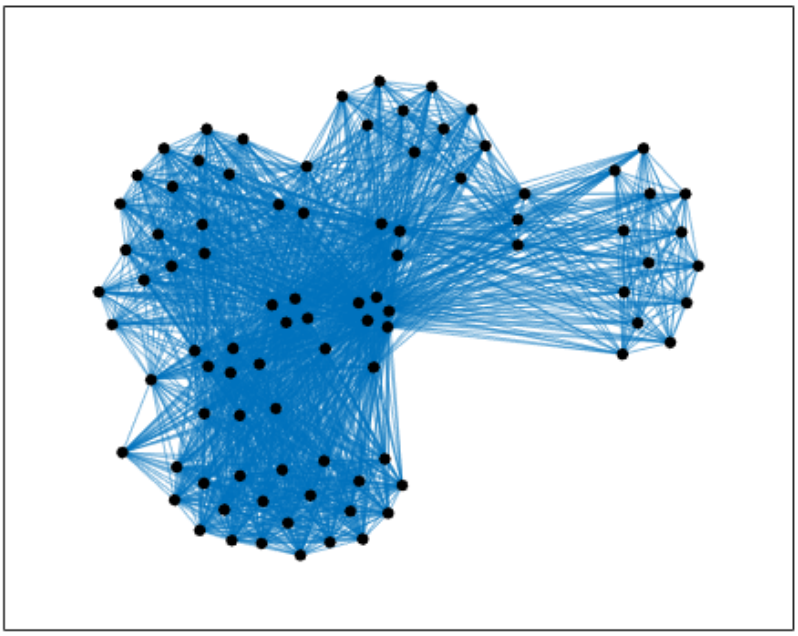


**Figura 4.20:** Grafo relativo alla proiezione del grafo bipartito sulle sorgenti.

maggiori. Un nodo caratterizzato da un alto valore di betweenness rappresenta uno speaker che ha pubblicato notizie riguardo più keywords, quindi connette gruppi di soggetti che hanno pubblicato notizie rispetto keywords diverse. Dunque un nodo con alta betweenness riesce a connettere cluster di sorgenti diversi. Si osserva che le notizie pubblicate su Facebook risultano essere le uniche con un valore di betweenness diverso da zero ma molto alto: ciò evidenzia la fallacia delle notizie pubblicate su questo social network, ma anche l'eterogeneità delle stesse.

In merito al grafo ottenuto dalla proiezione sulle keywords, invece, si possono effettuare considerazioni interessanti in quanto, rispetto alle sorgenti, sono in numero inferiore e visivamente l'analisi è più intuitiva (come mostrato

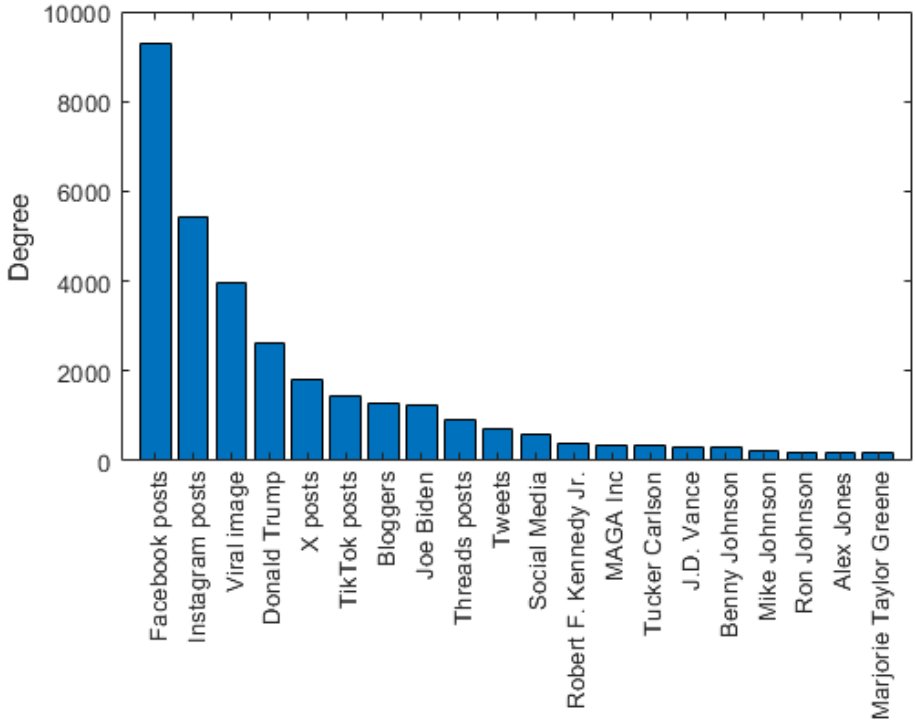
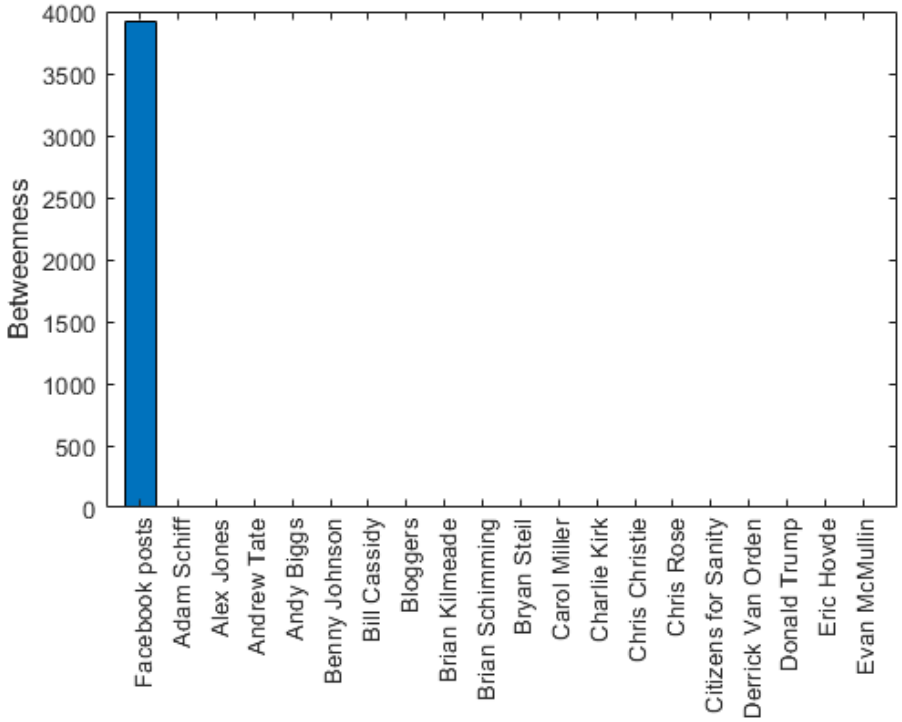



**Figura 4.21:** Valori più rilevanti in termini di grado relativi alla proiezione sulle sorgenti.




in Fig. 4.22). Sugli archi viene riportata la distanza tra i nodi, calcolata con il peso originale. In questo caso, dunque, un valore maggiore indica che esiste una forte associazione tra quelle due keywords perché gli speaker tendono a pubblicare notizie, indipendentemente dal grado di fakeness, su entrambe le keywords. Come già detto nel capitolo 3, per questa proiezione è stata considerata anche la matrice dicotomica associata alla matrice di affiliazione, mostrata in Fig. 4.23. In tal caso il peso associato all'arco che connette due nodi indica il numero di speaker che hanno pubblicato fake news su entrambi i topic relativi alla coppia di keywords considerate.

Inoltre, essendo il grafo completo, cioè ogni suo nodo è connesso con ogni altro nodo, è stata prodotta una mappatura delle distanze (Fig. 4.24), in modo da poter effettuare delle valutazioni sulle coppie di keywords. Le distanze

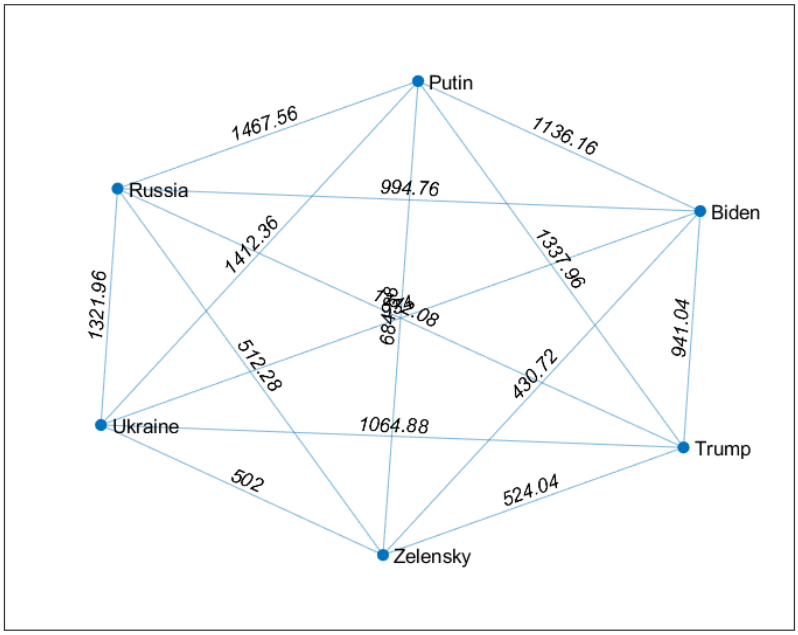



**Figura 4.22:** Grafo relativo alla proiezione del grafo bipartito sulle keywords.

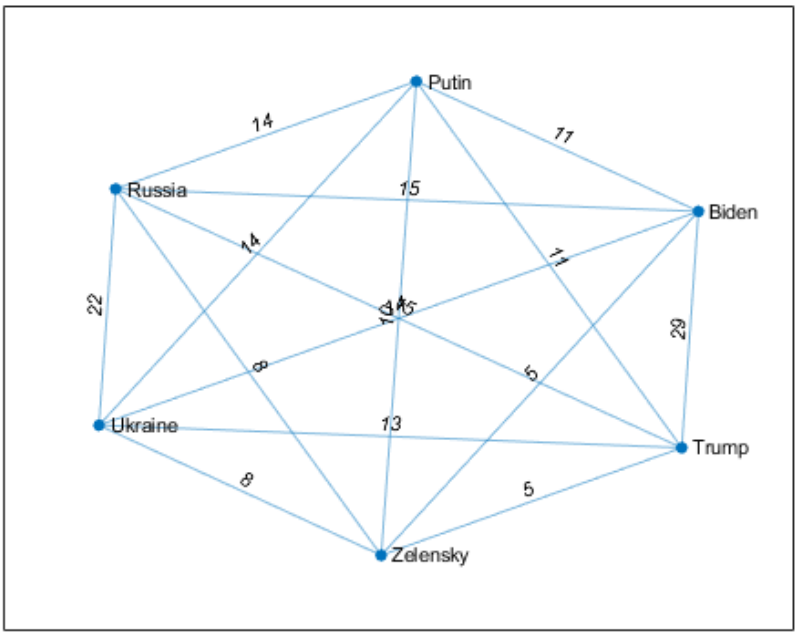



**Figura 4.23:** Grafo relativo alla proiezione del grafo bipartito sulle keywords utilizzando la matrice dicotomica associata a quella di affiliazione.

sono state calcolate con il reciproco del peso originale, per cui quanto più due keywords sono vicine, tanto più diversi soggetti hanno pubblicato fake news su entrambi gli argomenti. Vengono considerati solo i valori delle distanze tra nodi diversi e si osserva che la mappa ha valori simmetrici: ciò è giustificato dal fatto che la matrice relativa alle proiezioni nello spazio delle keywords sia a sua volta simmetrica. Si osservano delle correlazioni interessanti. Il valore minimo di distanza si registra nella la coppia di keywords "Russia" e "Ukraine", che sono due argomenti di grande interesse attualmente, per cui sono tante le sorgenti che pubblicano notizie false sia sull'uno che sull'altro topic. Per quanto riguarda invece le distanze più alte, queste si registrano tra

| | Biden | Putin | Russia | Trump | Ukraine | Zelensky |
|---|---|---|---|---|---|---|
| Biden | | 0,00088 | 0,001005 | 0,001063 | 0,001071 | 0,002322 |
| Putin | 0,00088 | | 0,000681 | 0,000747 | 0,000708 | 0,00146 |
| Russia | 0,001005 | 0,000681 | | 0,000899 | 0,000756 | 0,001952 |
| Trump | 0,001063 | 0,000747 | 0,000899 | | 0,000939 | 0,001908 |
| Ukraine | 0,001071 | 0,000708 | 0,000756 | 0,000939 | | 0,001992 |
| Zelensky | 0,002322 | 0,00146 | 0,001952 | 0,001908 | 0,001992 | |

**Figura 4.24:** Mappa delle distanze reciproche tra i nodi del grafo relativo alla proiezione sulle keywords.

le coppie "Zelensky"-"Biden" e "Zelensy"-"Ukraine". In tal caso, le notizie false riguardo questi argomenti sono diffuse da speaker diversi. L'eterogeneità degli speaker potrebbe risultare un fattore determinante in un'analisi mirata ad individuare i maggiori diffusori di fake news, poiché se si andasse ad analizzare, ad esempio, una coppia di keywords con una distanza bassa e si avesse la possibilità di determinare i maggiori diffusori di fake news per un argomento, sarebbe molto probabile che quel soggetto sia un diffusore di notizie false anche per l'altro topic. Inoltre, questi risultati fanno emergere un aspetto molto interessante riguardo le keywords che si potrebbero considerare vicine perché relative ad argomenti correlati: si consideri il caso della parola chiave "Zelensky", la quale ha distanze elevate rispetto tutte le altre keywords considerate, comprese "Ukraine" e "Putin". Dunque, non è detto che keywords tra loro correlate siano necessariamente vicine. D'altro canto, bisogna considerare che la parola chiave "Zelensky" è quella caratterizzata dal grado più basso: rientrando il numero di connessioni tra le informazioni che contribuiscono a determinare il peso associato agli archi, le distanze risultano essere alte principalmente per questo motivo.

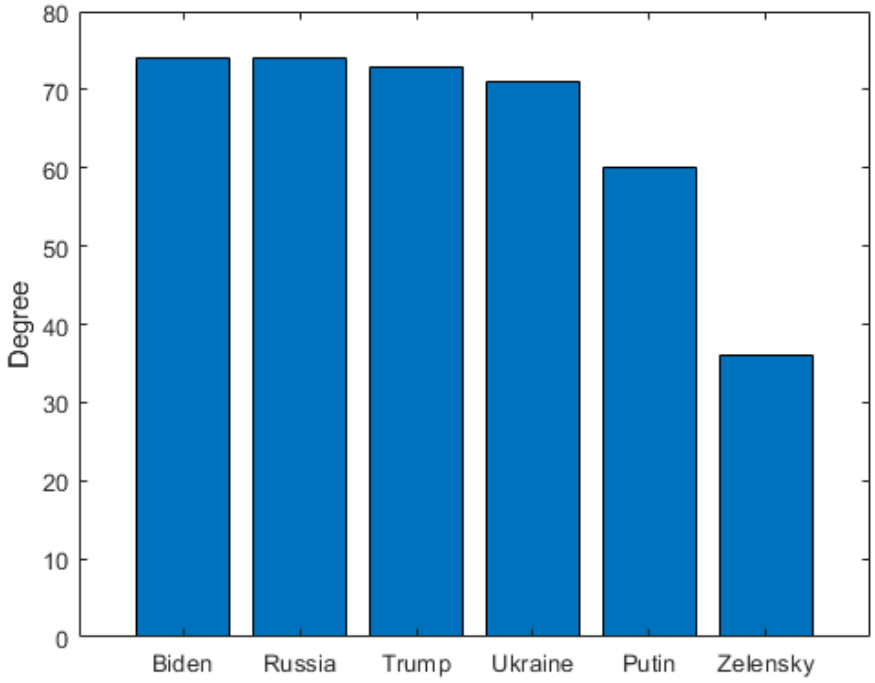



**Figura 4.25:** Barplot dei valori dei gradi relativi alle keywords nel grafo delle proiezioni calcolato a partire dalla matrice dicotomica.

Per la proiezione nello spazio delle keywords sono stati prodotti dei barplot relativi al grado dei nodi relativi al grafo ottenuto considerando sia la matrice dicotomica associata alla matrice di affiliazione, sia la matrice di affiliazione originale.

Per quanto riguarda il grado dei nodi relativi al grafo la cui matrice è stata ottenuta considerando la matrice dicotomica di quella di affiliazione, un valore alto indica che notizie non veritiere su quella keywords vengono condivise da tanti speakers che condividono notizie anche su altri argomenti. I risultati sono riportati in Fig. 4.25.

Per il caso in cui non è stata considerata la matrice dicotomica, invece,

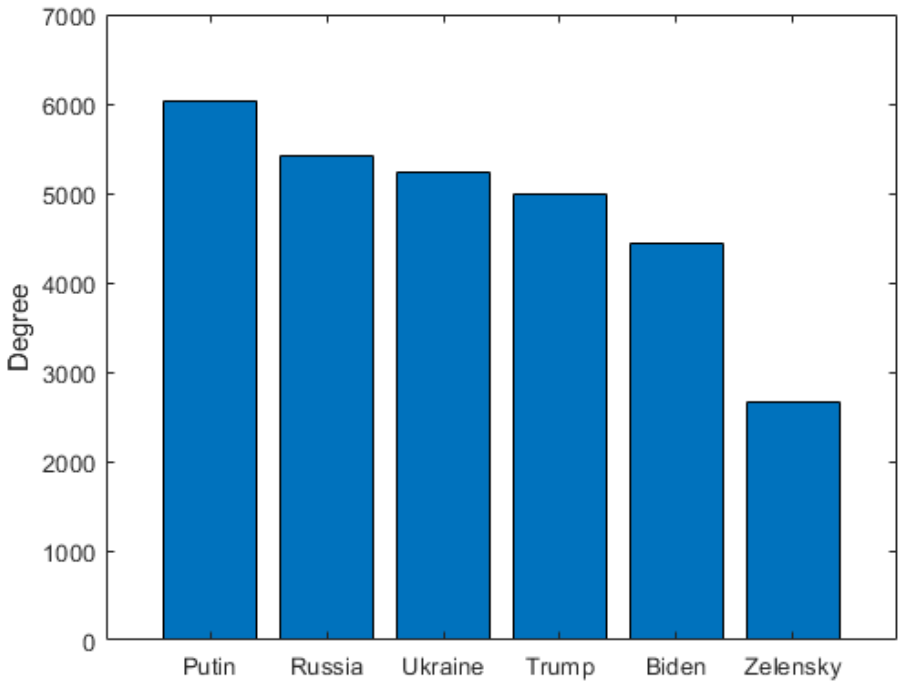



**Figura 4.26:** Barplot dei valori dei gradi relativi alle keywords nel grafo delle proiezioni.

un valore alto per il grado di un nodo indica che le notizie false su quelle keywords vengono condivise da tanti speakers che condividono notizie anche su altri argomenti. Dunque, in questo caso si ha un'indicazione sui diffusori di notizie false. Il barplot relativo è riportato in Fig. 4.26. Nel caso in cui una keyword avesse un grado alto nel primo caso e basso nel secondo, quella keyword potrebbe essere considerata oggetto di notizie che tendono ad avere gradi di fakeness bassi.

## 4.2 Rete di propagazione gerarchica

Il contributo di questa tesi, rispetto al primo dei due esperimenti realizzati, è quello di produrre un'analisi del dataset completo, con l'obiettivo di migliorare i risultati di partenza utilizzando il modulo *GridSearchCV* di scikit-learn e fornendo un'ulteriore valutazione dei risultati utilizzando una rete neurale di tipo Feedforward. Infine, si realizza un'analisi di sensitività degli algoritmi utilizzati per valutare le prestazioni di rilevamento delle fake news rispetto alle features considerate nell'analisi utilizzando il modulo *Permutation Feature Importance*.

Per quanto riguarda la fase relativa alla scelta dei migliori parametri per i classificatori, si riportano quelli relativi al Random Forest e alla rete neurale Feedforward, i quali hanno prodotto, in generale, i risultati migliori per entrambi i dataset. La lista dei migliori parametri è stata salvata per ogni classificatore in un file CSV e il cui contenuto viene riportato nella Tab.4.1. Per quanto riguarda i parametri della FNN per la parte di dati relativa a PolitiFact, ha un singolo layer nascosto composto da 50 neuroni: il motivo delle dimensioni ridotte della rete è da ricercarsi nel numero ridotto di dataset a disposizione e risulta comunque una scelta che bilancia la capacità di apprendimento del modello e la prevenzione dell'overfitting. Un batch_size pari a 8 significa che l'aggiornamento dei pesi avviene più frequentemente, il che può portare a una convergenza più rapida e stabile, specialmente con un dataset di piccole dimensioni. Il tasso di apprendimento 'adaptive' fa in modo che il suo valore sia inizialmente costante, per poi diminuire se i risultati di validazione non dovessero migliorare. Per la porzione di dati relativa a a

GossipCop i parametri della rete sono pressoché gli stessi, con la differenza che la funzione di attivazione è ReLu piuttosto che Tanh e il batch_size passa da 8 a 16. Con un dataset più ampio, la rete può beneficiare di una funzione di attivazione che tende a convergere più velocemente rispetto alla tangente iperbolica. Inoltre, Tanh si adatta meglio a dataset di piccole dimensioni in quanto la sua non linearità aiuta a catturare catturare relazioni complesse nei dati. Un altro aspetto da considerare è che la ReLU tende a mantenere gradienti più stabili durante la fase di addestramento, riducendo problemi come quello del "vanishing gradient", che possono caratterizzare funzioni come la Tanh: con un maggior numero di dati, la stabilità dei gradienti può rappresentare un vantaggio. Il fenomeno del vanishing gradient si verifica quando i gradienti delle funzioni di perdita diventano estremamente piccoli, rendendo inefficace l'aggiornamento dei pesi nei layer più profondi della rete. Quando si utilizzano funzioni come Tanh o la sigmoide, i gradienti calcolati durante la fase di backpropagation possono diventare molto piccoli. Questo è dovuto al fatto che queste funzioni di attivazione saturano molto velocemente agli estremi.

Per quanto riguarda RF, i parametri ottimali per il modello Random Forest Classifier trovati con GridSearchCV su dataset di diverse dimensioni evidenziano come la dimensione del dataset possa influenzare la configurazione del modello stesso. Il criterio Gini tende ad essere leggermente più veloce da calcolare rispetto all'entropia, il che può essere vantaggioso con dataset più grandi. D'altro canto, Entropy può catturare informazioni più dettagliate e fornire una migliore separazione delle classi nei piccoli dataset, dove ogni informazione in più può essere cruciale per migliorare la perfor-

mance del modello. Un valore maggiore di max_depth permette alla foresta di catturare relazioni più complesse nei dati, il che è possibile e utile con un numero maggiore di dati, mentre con dataset più piccoli aumenterebbe il rischio di imbattersi nel problema dell'overfitting. Dunque, per il dataset di dimensione inferiore, un valore più basso di questo parametro limita la complessità del modello. Ragionamento analogo per il numer odi estimatori (alberi): con un dataset più grande, un numero maggiore di alberi (100) può migliorare la stabilità e la performance del modello. Con più dati a disposizione, si possono costruire alberi robusti che catturano meglio le variazioni nei dati. Con dataset più piccoli, un numero minore di alberi (50) è spesso sufficiente per ottenere buone performance, riducendo anche il tempo di addestramento e il rischio di overfitting. L'impostazione max_features a sqrt è una scelta comune che funziona bene in molti scenari, bilanciando la diversità tra gli alberi nella foresta. Questa scelta non cambia tra i due dataset poiché è un'impostazione che offre buone performance indipendentemente dalla dimensione del dataset.

| Classifier | PolitiFact | GossipCop |
|---|---|---|
| FNN | activation: tanh,<br>alpha: 0.0001,<br>batch_size: 8,<br>early_stopping: True,<br>hidden_layer_sizes: [50],<br>learning_rate: adaptive,<br>max_iter: 2000,<br>n_iter_no_change: 15,<br>solver: adam | activation: relu,<br>alpha: 0.0001,<br>batch_size: 16,<br>early_stopping: True,<br>hidden_layer_sizes: [50],<br>learning_rate: adaptive,<br>max_iter: 2000,<br>n_iter_no_change: 15,<br>solver: adam |
| RF | criterion: entropy,<br>max_depth: 20,<br>max_features: sqrt,<br>n_estimators: 50 | criterion: gini,<br>max_depth: 40,<br>max_features: sqrt,<br>n_estimators: 100 |

**Tabella 4.1:** Valori degli iperparametri ottenuti per i classificatori FNN e RF.

Di seguito si analizzano i risultati per ognuno dei classificatori utilizzati, rispettivamente per PolitiFact e GossipCop.

Il primo aspetto che salta all'occhio è il valore nullo relativamente alla deviazione standard per alcuni classificatori (SVM, LR e GNB) in entrambi i dataset. In generale, una bassa deviazione standard indica che le prestazioni del modello sono consistenti tra le diverse iterazioni. Il motivo per cui si verifica questo fenomeno è che alcuni classificatori producono risultati molto stabili tra diverse esecuzioni del modello, al contrario di altri che invece presentano una certa variabilità nelle loro prestazioni. In particolare, SVM, LR e GNB tendono ad essere deterministici, nel senso che a partire da un insieme di dati di training e test, producono sempre lo stesso modello e quindi le stesse predizioni. Inoltre, non sono basati su processi stocastici, come l'inizializzazione casuale dei persi, per cui non prevedono componenti casuali che possano introdurre variabilità nei risultati, come il bootstrap sampling in Random Forest (per costruire alberi ogni volta diversi) o l'inizializzazione dei pesi in reti neurali.

Per quanto riguarda PolitiFact i risultati sono riportati in Tab.4.2. Il classificatore RF presenta i migliori risultati in termini di accuracy e precision. Infatti, emerge che l'81.8% delle predizioni del modello sono corrette e avendo una deviazione standard relativamente bassa, le prestazioni del modello sono stabili su diverse esecuzioni. Un valore di precision dell'81.1% indica la percentuale delle notizie identificate come false che sono ffettivamente false. Il miglior risultato in termini di recall, però, lo riporta la FNN, che è in gradi identificare il 90.9% delle fake news, sebbene sia caratterizzata da una variabilità (0.047) mggiore di quella del RF (0.009). Un valore alto per l'F1-score

indica che c'è un buon equilibrio tra precision e recall. In altre parole, il classificatore non solo riesce a rilevare una buona parte delle notizie vere, ma mantiene anche un valore ragionevole di precision, riducendo il numero di falsi positivi. Quello che emerge, dunque, è che se l'obiettivo è minimizzare i falsi positivi (massimizzare la precision), il RF risulta essere il classificatore più indicato. Ma se l'obiettivo è quello di identificare il maggior numero di fake news, come nel caso d istudio, allora la FNN è la scelta migliore.

| Classifier | Accuracy | Precision | Recall | F1-score |
|---|---|---|---|---|
| RF | **0.818 ± 0.011** | **0.811 ± 0.021** | 0.825 ± 0.009 | **0.818 ± 0.008** |
| FNN | 0.796 ± 0.011 | 0.740 ± 0.021 | **0.909 ± 0.047** | **0.816 ± 0.011** |
| SVM | 0.802 ± 0.000 | 0.754 ± 0.000 | 0.891 ± 0.000 | **0.817 ± 0.000** |
| DT | 0.750 ± 0.035 | 0.737 ± 0.043 | 0.775 ± 0.044 | 0.754 ± 0.032 |
| LR | 0.748 ± 0.000 | 0.690 ± 0.000 | 0.891 ± 0.000 | 0.778 ± 0.000 |
| GNB | 0.766 ± 0.000 | 0.699 ± 0.000 | **0.927 ± 0.000** | 0.797 ± 0.000 |

**Tabella 4.2:** Lista dei migliori parametri per i classificatori che hanno prodotto i migliori risultati per PolitiFact.

Per il dataset GossipCop i risultati sono riportati in Tab.4.3 e si osservano valori comparabili tra RF e FNN per tutte le metriche considerate, le quali risultano anche le migliori, eccezion fatta per la precision che vede un valore leggermente maggiore nell'algoritmo di LR. Entrambi i modelli mostrano una buona stabilità, con il RF che ha una variabilità leggermente inferiore rispetto alla FNN.

| Classifier | Accuracy | Precision | Recall | F1-score |
|---|---|---|---|---|
| RF | **0.879 ± 0.003** | 0.866 ± 0.003 | **0.898 ± 0.004** | **0.881 ± 0.003** |
| FNN | **0.874 ± 0.003** | 0.861 ± 0.006 | **0.891 ± 0.010** | **0.875 ± 0.004** |
| SVM | 0.854 ± 0.000 | 0.877 ± 0.000 | 0.824 ± 0.000 | 0.850 ± 0.000 |
| DT | 0.840 ± 0.005 | 0.843 ± 0.002 | 0.837 ± 0.012 | 0.840 ± 0.006 |
| LR | 0.843 ± 0.000 | **0.882 ± 0.000** | 0.791 ± 0.000 | 0.834 ± 0.000 |
| GNB | 0.655 ± 0.000 | 0.858 ± 0.000 | 0.370 ± 0.000 | 0.518 ± 0.000 |

**Tabella 4.3:** Lista dei migliori parametri per i classificatori che hanno prodotto i migliori risultati per GossipCop.

I risultati sono in linea con quelli prodotti dal lavoro orginale, sebbene fossero riportati solo quelli relativi al dataset PolitiFact e rispetto a una porzione ridotta del dataset considerato in questa analisi. Risulta evidente come le prestazioni dei modelli tendano a migliorare all'aumentare del numero di dati a disposizione.

I risultati evidenziano che con un dataset più ampio, entrambi i modelli mostrano, in generale, miglioramenti nelle metriche rispetto al dataset più piccolo. Questo suggerisce che entrambi i modelli beneficiano di una maggiore quantità di dati, migliorando le loro capacità di generalizzazione e le prestazioni complessive.

Per l'analisi di sensitività degli algoritmi rispetto alle features utilizzate vengono riportati i dati relativi agli algoritmi che hanno prodotto i risultati migliori (RF e FNN). In entrambi i classificatori l'importanza delle features produce dei valori abbastanza in linea tra di loro e con i risultati originali. Le caratteristiche temporali del dataset PolitiFact hanno punteggi di importanza più elevati rispetto alle caratteristiche strutturali e linguistiche, il che

dimostra che le features temporali svolgono un ruolo primario nella classificazione delle fake news. Le caratteristiche strutturali funzionano meglio delle caratteristiche linguistiche in entrambi i set di dati poiché la micro-rete ha contenuti linguistici limitati.

Entrando nel vivo dell'analisi dei risultati relativi all'importanza delle features, si inizia con PolitiFact. In Fig.4.27 è riportato il risultato relativo al RF, mentre in Fig. 4.28 quello relativo alla FNN. Il valore più alto si registra per la feature T9, la quale indica che la frequenza media con cui gli utenti si rispondono reciprocamente nella micro-rete è il fattore più importante per la classificazione delle notizie false. Ciò indica che considerare le reply-chain con tempi ridotti tra una risposta e l'altra potrebbe portare all'individuazione di una notizia falsa, piuttosto che di una veritiera. Un valore alto per la feature T4 indica che la differenza temporale tra il primo e l'ultimo tweet relativi a una notizia falsa è tipicamente più breve di quella che caratterizza una notizia vera, per cui può rappresentare un aspetto rilevante per distinguere le notizie vere da quelle false. La durata delle notizie nella macro-rete catturata da T2 mostra il punteggio di importanza più alto per la FNN. Ciò indica che in generale quando viene diffusa una fake news, il tempo medio che intercorre tra il primo tweet e l'ultimo retweet è molto basso. La feature T11 mostra che la durata degli scambi nella micro-rete è la caratteristica più importante nella classificazione delle notizie false. Dunque, individuare interazioni di breve durata nella rete di micro-livello, potrebbe essere un buon indicatore per individuare fake news. Per quanto riguarda le caratteristiche che hanno fatto registrare un impatto negativo sulle prestazioni del modello, emerge che si tratta principalmente di caratteristiche

strutturali. Infatti, per la FNN è la feature S12 che riporta un valore più alto in negativo: essa rappresenta il grado di uscita massimo nella micro-rete, cioè il numero massimo di nuovi commenti nella reply-chain di un particolare nodo. Questo significa che ai fini del rilevamento delle fake news utilizzando un modello basato su FNN non è rilevante considerare questa caratteristica. Per RF, invece, la caratteristica S9, caratteristica della rete di macro-livello relativa alla frazione di utenti bot che retwittano, è quella con il più alto valore di importanza negativo. Questo rappresenta un risultato sorprendente, in quanto un numero maggiore di retweet da parte di bot potrebbe essere associato più a notizie vere che a fake news, contrariamente a quanto previsto inizialmente. È opportuno, però, fare chiarezza su un aspetto: se una feature ha un'importanza negativa, non è detto che sia esclusivamente dannosa per il modello. Infatti, la feature potrebbe essere caratterizzata da un'alta variabilità, motivo per cui non ha una relazione chiara con la classificazione delle fake news. Dunque, l'analisi dell'importanza delle features pone le basi per uno studio delle caratteristiche, come la S9, che producono risultati diversi da quelli attesi, per cui potrebbe essere utile analizzarle al fine di comprendere qual è il motivo per cui assume questo comportamento. In merito alle caratteristiche linguistiche, non si registrano valori di importanza significativi in entrambi i classificatori. Questo dimostra che il tentativo di estrarre informazioni linguistiche utilizzando un valore del sentiment non è sufficiente.

Analizzando invece i risultati dell'analisi della varianza degli algoritmi RF (Fig. 4.29) e FNN (Fig. 4.30 rispetto alle features per GossipCop, è evidente che la mancanza di informazioni relative alle caratteristiche linguistiche pro-

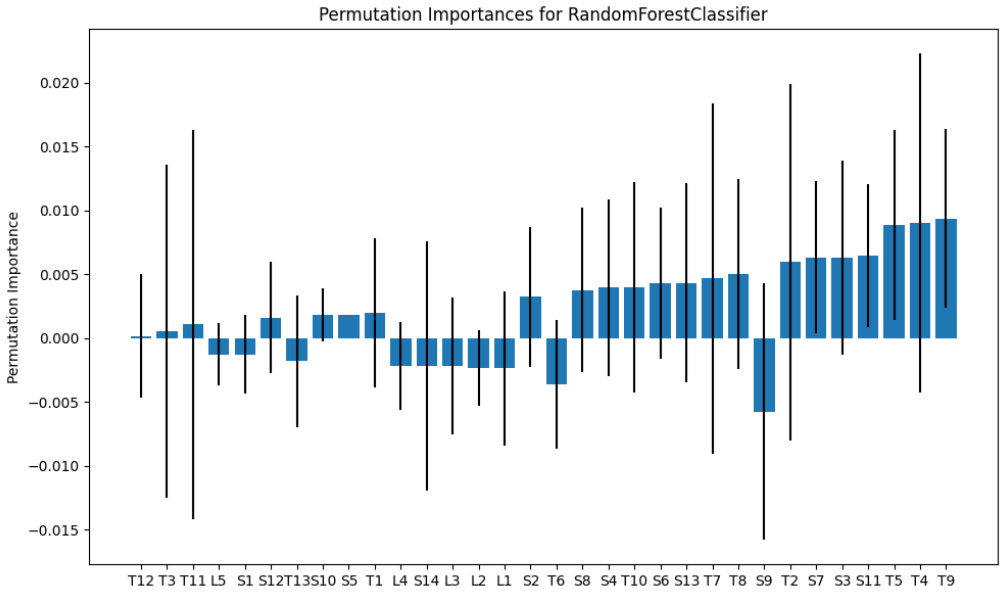



**Figura 4.27:** Grafico relativo alla feature importances per RF in PolitiFact.

duce valori nulli in termini di importanza. Rispetto a quanto osservato per PolitiFact, in GossipCop ci sono meno valori negativi di importanza e ai fini della rilevazione delle fake news risultano più significative le caratteristiche strutturali piuttosto che quelle temporali. La caratteristica che fa registrare il valore di importanza più alto è, per entrambi i classificatori, quella relativa alla frazione di tweet con retweet tra tutte le cascate (S7). Ciò inidica che in media sono più i tweet che pubblicano notizie false ad essere maggiormente retwittati. La feature S14, relativa alla rete di micro-livello, è la seconda con un valore più alto ed è relativa alla frazione di cascate che hanno almeno una risposta ntra tutte le cascate. Il motivo è che le notizie false hanno maggiore probabilità di essere correlate ad argomenti controversi e di tendenza, che generano un maggiore coinvolgimento in termini di risposte/commenti rispetto alle notizie reali.

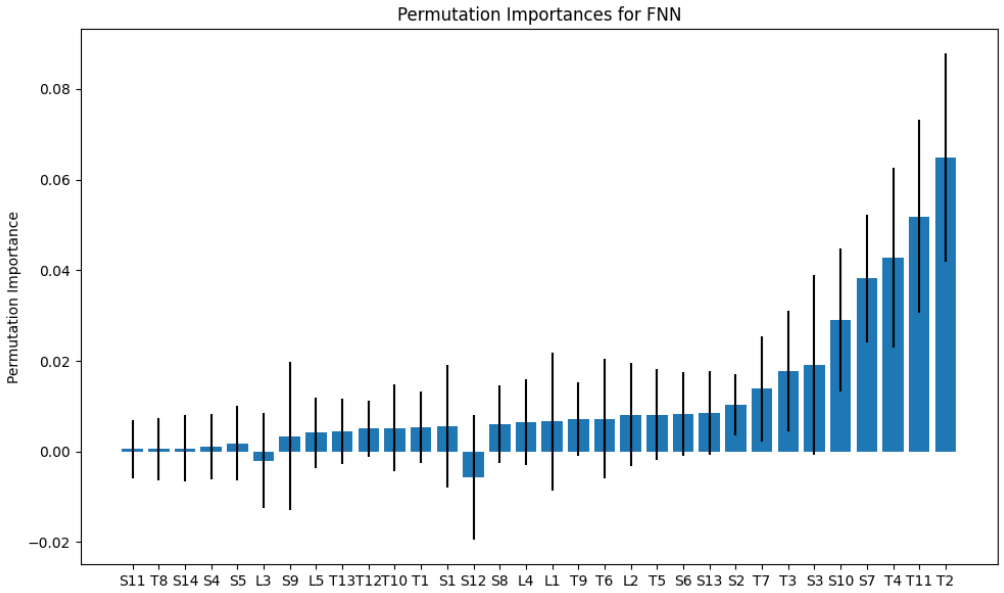



**Figura 4.28:** Grafico relativo alla feature importances per la FNN in PolitiFact.

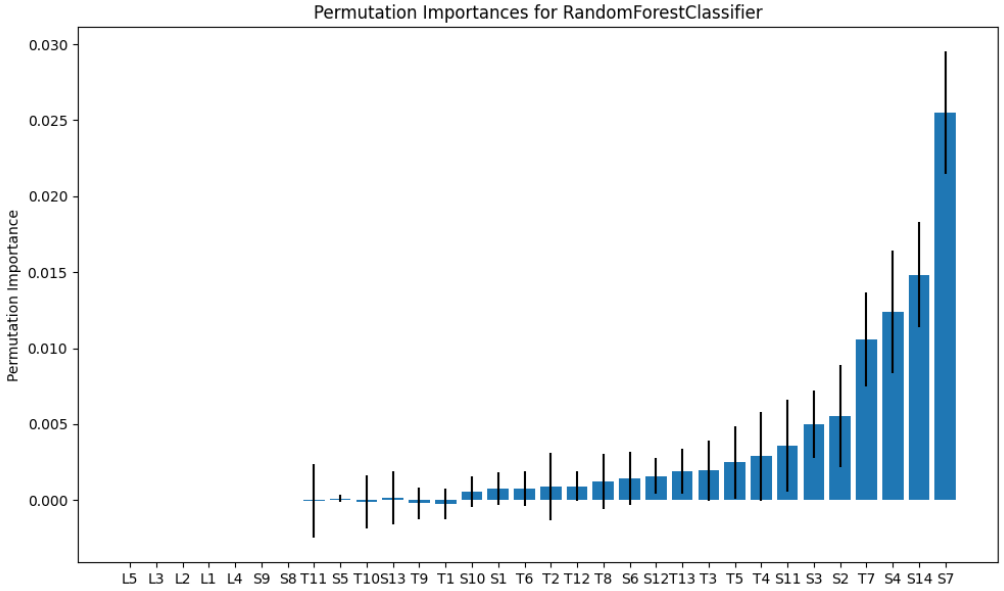



**Figura 4.29:** Grafico relativo alla feature importances per RF in GossipCop.

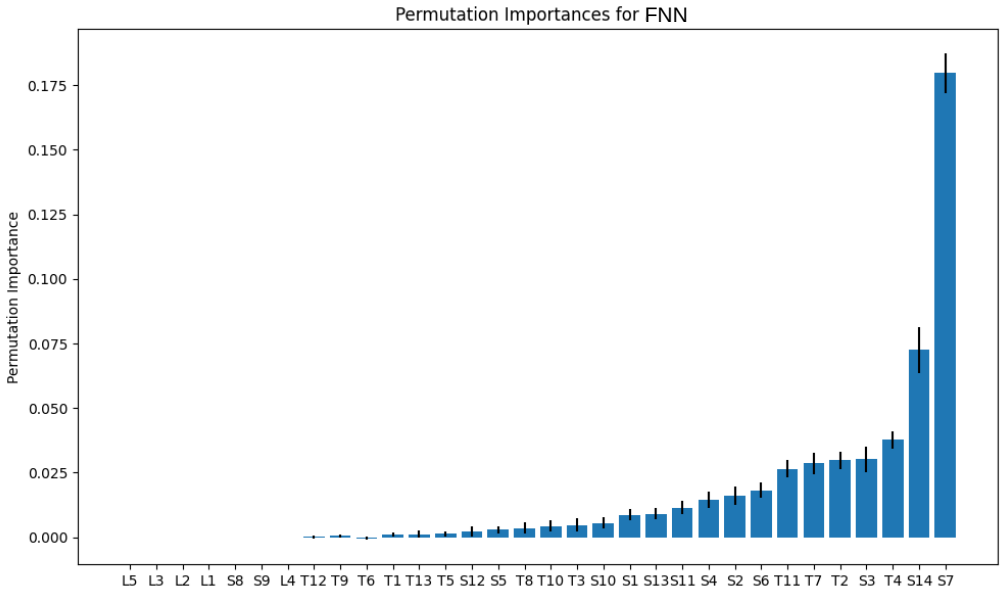



**Figura 4.30:** Grafico relativo alla feature importances per la FNN in GossipCop.

# Capitolo 5

# Conclusioni e sviluppi futuri

Questa tesi ha esplorato in maniera approfondita l'uso di metodologie avanzate per il rilevamento delle fake news, con un focus particolare sull'analisi dei grafi. Attraverso un'analisi dettagliata della letteratura esistente, è emerso che le tecniche basate su grafi offrono un potente strumento per comprendere la diffusione delle informazioni false nei social media. In particolare, queste tecniche permettono di modellare la rete di interazioni tra utenti e contenuti, facilitando l'identificazione di pattern specifici associati alla loro propagazione.

La mancanza di un dataset di riferimento per la fake news detection rende complessa la valutazione dell'efficacia di ogni approccio, oltre che il confronto tra di essi. La ricerca sulla raccolta di dati per questo tipo di analisi dovrebbe concentrarsi sulla sulla costruzione di set di dati su larga scala e soprattutto sull'identificazione di un benchmark chiaro e accettato per la valutazione, in modo da consentire un'analisi a un livello più simile a quello degli scenari reali.

## 5.1 Grafo bipartito

Il primo esperimento condotto in questa tesi ha permesso di realizzare una rappresentazione tramite un grafo bipartito delle notizie pubbicate da alcune sorgenti riguardo particolari argomenti determinati dalle keywords utilizzate per effettuare le ricerche sul sito PolitiFact. Avendo considerato sia il peso originale, sia quello normalizzato sul numero di notizie pubblicate da ciascuno speaker per ogni keyword, le analisi emerse e riportate sotto forma di barplots hanno permesso di effettuare confronti incrociati. A partire dal grafo bipartito, poi, sono state considerate le sue proiezioni nello spazio delle sorgenti e in quello delle keywords. In questo modo sono state condotte delle valutazioni da più punti di vista. L'analisi degli speaker ha permesso di identificare un comportamento che non sarebbe stato possibile osservare immediatamente con un'analisi tradizionale. La rappresentazione della proiezione delle sorgenti, infatti, mostra una tendenza a creare dei gruppi di speaker che hanno diffuso notizie relative ad argomenti comuni. Per questo motivo si potrebbe imbastire un'analisi di rete futura per caratterizzare una clusterizzazione degli speaker rispetto alle fake news, al fine di realizzare una sorta di profilatura di quelli che possono essere i potenziali diffusori di notizie su keyword comuni e con un grado di verità simile. Si potrebbe, dunque, individuare una rete significativa di sorgenti da monitorare in quanto potenziali diffusori di fake news. L'analisi condotta sulla proiezione delle keywords, invece, ha consentito di individuare una mappatura delle distanze (calcolate come il reciproco dei pesi originali) tra le coppie di keywords. Dall'analisi è stato possibile individuare coppie di keywords per cui sono tante sorgen-

ti che pubblicano fake news. L'eterogeneità degli speaker potrebbe essere sfruttata in un'analisi futura come informazione per individuare gli speaker che tendono a diffondere maggiormente fake news. Si pensi a una coppia di keywords separati da una distanza molto bassa e si potesse determinare uno speaker che tende a diffondere notizie false su una delle due keywords, sarebbe molto probabile che possa risultarlo anche per l'altra. In questo modo si potrebbero individuare aree tematiche che sono spesso oggetto di fake news e monitorarle. La possibilità di avere a disposizione un dataset continuamente aggiornato e con dei topic sempre attuali, rappresenta un fattore significativo per le analisi future. Sarebbe interessante anche estendere l'analisi ad altri aspetti estratti dal sito PolitiFact, in modo da renderla più accurata. Ad esempio, si potrebbe aggiungere la dimensione degli autori, intesi come coloro che classificano le notizie assegnando il target del "Truth-O-Meter". Realizzando fact-checking manuale, si potrebbe considerare l'impatto che pesa sulle scelte effettuate dagli autori.

## 5.2 Rete di propagazione gerarchica

Lo studio condotto nel secondo esperimento proposto in questa tesi può essere utilizzato in futuro come base di partenza per un'analisi di natura simile, ma la proposta potrebbe essere quella di utilizzare un dataset più ampio, con dei dati più puliti e con la possibilità di accedere alle informazioni relative agli utenti e ai tweet. I risultati sperimentali hanno dimostrato che l'approccio basato sui grafi è efficace nel rilevamento delle fake news. In particolare, l'uso di tecniche di machine learning, come Random Forest e Feedforward Neural Network, ha mostrato prestazioni robuste su dataset di diverse dimensioni. Tuttavia, è emerso che la qualità dei dati è cruciale per il successo del modello, evidenziando la necessità di dataset accurati e rappresentativi. Un aspetto da considerare è la natura "streaming" dei social media, la quale porta al cosiddetto "concept drift" (deriva del contenuto) [25]. Nelle piattaforme social, l'importanza dei dati e le caratteristiche considerate, indipendentemente dalla loro natura (basata sul contenuto o sul contesto), potrebbe variare nel tempo. Dunque, le features estratte per un determinato evento o periodo di tempo, potrebbero non essere scalabili e generalizzabili per le applicazioni del mondo reale o per i dati provenienti da un contesto diverso. L'analisi delle feature ha inoltre rivelato informazioni interessanti, ma potrebbe essere utile andare ad approfondire le motivazioni che determinano la crescita o la decrescita dell'importanza di determinate features, che tramite quest'analisi non è possibile comprendere. Gli strumenti presentati in questa tesi permettono di prevedere se un utente diffonderà o meno una fake news studiando la struttura delle reti di propagazione gerarchica, che è un

modo per mitigare la diffusione delle fake news. Ma fare ciò con informazioni sugli utenti rese anonime e con i contenuti delle notizie non condivisi, non fornisce un contributo pratico per la soluzione del problema. È chiaro che un dataset del genere non è ancora disponibile pubblicamente, ma lo studio della propagazione delle notizie sui social media e la detection delle stesse sono notevolmente limitati da questi aspetti.

Investire in ricerca e sviluppo nel campo della rilevazione delle fake news è fondamentale per proteggere la diffusione dell'informazione. Negli ultimi anni, soprattutto in periodi delicati caratterizzati da pandemie e guerre, l'importanza di accedere a informazioni verificate non può essere sottovalutata. Le situazioni di emergenza generano incertezza e paura, rendendo le persone particolarmente vulnerabili alla disinformazione. In tali contesti, la diffusione di informazioni false o fuorvianti può avere conseguenze catastrofiche, mettendo a rischio vite umane e compromettendo la coesione sociale.

# Ringraziamenti

La parte dei ringraziamenti è quella che preferisco, perché mi permette di esprimere gratitudine nei confronti delle persone che mi hanno accompagnato durante questo percorso e che hanno contribuito al raggiungimento di questo traguardo.

Il primo ringraziamento va ai miei relatori, i Professori Antonio Pecchia e Francesco Vasca. Credo sia stato un privilegio aver avuto la possibilità di lavorare sotto la loro guida, avendo anche la possibilità di conoscere in parte il loro lato umano. Vi ringrazio per la comprensione e la disponibilità che avete mostrato nei miei confronti. Custodirò gelosamente ogni consiglio, didattico e non, sperando di sfruttarli al meglio in futuro. Ci tengo anche a ringraziare l'Ingegnere Carmela Bernardo, che mi ha pazientemente sostenuto in questi mesi.

Un ringraziamento speciale va a mia nonna Pia. Ho sempre sognato di poterti dedicare questo traguardo, sapendo quanto ci tenessi e quanto ti avrebbe fatto piacere. Dedicartelo oggi non mi permette di poter godere del tuo sorriso o dei tuoi baci, ma so che da lassù mi hai sempre accompagnato. Spero di averti resa orgogliosa.

Ringrazio mia madre, che è la persona che mi supporta in ogni momento.

È soprattutto grazie a te che ho superato i momenti più difficili in questi anni.

Ringrazio mia sorella Sofia, che in questi anni mi ha sostenuto e ha ricevuto poco da parte mia. Rappresenti un punto di riferimento per me, sempre.

Ringrazio mio padre, che mi ha supportato durante questo percorso della magistrale. Per me è stato importante.

Ringrazio i miei cugini, compresi quelli "acquisiti". Siete anche voi un punto di riferimento per me. Questi anni sono stati strani, vissuti tra momenti brutti e momenti belli, ma averli condivisi con voi mi fa sentire fortunato.

Ringrazio Tommaso, a cui è doveroso dedicare uno spazio esclusivo. La gioia che ha portato la tua nascita è indescrivibile. Grazie a te le pause dallo studio sono state rivitalizzanti.

Ringrazio i miei zii Mario e Cosimo. Voglio ringraziarvi per tutto quello che fate per me ogni giorno, il vostro supporto è stato determinante per restare sempre concentrato sui miei obiettivi.

Ringrazio le mie zie Francesca e Loredana. Ognuna, a modo suo, mi ha dimostrato affetto e per questo vi sono molto riconoscente.

Ringrazio Giuseppe e tutta la sua famiglia, per il bene che mi dimostrano in ogni occasione.

Ringrazio i miei compagni Roberto e Donato, con voi ho condiviso questo percorso e sono sicuro che senza di voi il risultato non sarebbe stato lo stesso. Sono fortunato ad aver trovato due persone come voi, la vostra amicizia è preziosa.

Ringrazio i ragazzi del gruppo Elio, amici di una vita. La vostra co-

stante presenza mi ha permesso di non sentirmi mai solo. Ognuno di voi è importante per me a modo suo, vi voglio bene boys.

Ringrazio Teresa e Ilenia, le mie due amiche del cuore. Grazie per l'interesse che mostrate nei miei confronti, per le chiacchierate e per i consigli. Vi voglio bene.

Ringrazio i ragazzi di casa Di Leo: Donato, Pietro e Giuseppe. Ogni progetto universitario ha una costante, legato appunto al luogo in cui è stato svolto. La vostra accoglienza, accompagnata sempre da un sorriso, ha sempre alleggerito il lavoro da fare.

Ringrazio Gianni e Guenda, due persone speciali che da lontano hanno sempre avuto un pensiero per me.

Ringrazio la famiglia del Wellcome, il presidente Pietro Milano, gli allenatori Francesco Giangregorio e Leo Martone, il tuttofare Alessandro De Blasis, i dirigenti e tutti i ragazzi con cui ho condiviso lo spogliatoio. Vi ringrazio per ogni allenamento che abbiamo condiviso, per ogni partita giocata, per le amicizie create e per la felicità che avete portato nella mia vita. Sempre forza Wellcome!

Ringrazio il Mister Mario Coopt e la famiglia del Brigante, che mi hanno accolto il primo anno di questa magistrale, facendo rinascere in me la voglia di giocare a pallone, ciò che più amo.

Ringrazio Pompea. In questo percorso mi hai sostenuto sempre, se oggi riesco a sentirmi una persona migliore, il merito è anche tuo. Per questo ti sarò sempre grato.

Ringrazio Ombra e Snoopy, i miei amici a quattro zampe. La vostra compagnia è stata fondamentale per tirarmi su il morale nei momenti bui.

# Bibliografia


[1] Hadeer Ahmed, Issa Traore, and Sherif Saad. Detection of online fake news using n-gram analysis and machine learning techniques. In Issa Traore, Isaac Woungang, and Ahmed Awad, editors, *Intelligent, Secure, and Dependable Systems in Distributed and Cloud Environments*, pages 127–138, Cham, 2017. Springer International Publishing.

[2] Hadeer Ahmed, Issa Traore, and Sherif Saad. Detecting opinion spams and fake news using text classification. *SECURITY AND PRIVACY*, 1(1):e9, 2018.

[3] Hunt Allcott and Matthew Gentzkow. Social media and fake news in the 2016 election. *Journal of Economic Perspectives*, 31(2):211–36, May 2017.

[4] Alessandro Bondielli and Francesco Marcelloni. A survey on fake news and rumour detection techniques. *Information Sciences*, 497:38–55, 2019.

[5] Erica J. Briscoe, D. Scott Appling, and Heather Hayes. Cues to deception in social media communications. In *2014 47th Hawaii International Conference on System Sciences*, pages 1435–1443, 2014.

[6] Arianna D'Ulizia, Maria Chiara Caschera, Fernando Ferri, and Patrizia Grifoni. Fake news detection: a survey of evaluation datasets. *PeerJ Computer Science*, 7:e518, June 2021.

[7] Shuzhi Gong, Richard O. Sinnott, Jianzhong Qi, and Cecile Paris. Fake news detection through graph-based neural networks: A survey, 2023.

[8] Elena Kochkina, Maria Liakata, and Arkaitz Zubiaga. All-in-one: Multi-task learning for rumour verification, 2018.

[9] David M. J. Lazer, Matthew A. Baum, Yochai Benkler, Adam J. Berinsky, Kelly M. Greenhill, Filippo Menczer, Miriam J. Metzger, Brendan Nyhan, Gordon Pennycook, David Rothschild, Michael Schudson, Steven A. Sloman, Cass R. Sunstein, Emily A. Thorson, Duncan J. Watts, and Jonathan L. Zittrain. The science of fake news. *Science*, 359(6380):1094–1096, 2018.

[10] Shaohua Li, Weimin Li, Alex Munyole Luvembe, and Weiqin Tong. Graph contrastive learning with feature augmentation for rumor detection. *IEEE Transactions on Computational Social Systems*, pages 1–10, 2023.

[11] Jing Ma, Wei Gao, and Kam-Fai Wong. Detect rumors in microblog posts using propagation structure via kernel learning. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, 2017.

[12] Ministero della Salute. Archivio delle fake news sul nuovo coronavirus, 2024. Accessed: 2024-05-30.

[13] Tanushree Mitra and Eric Gilbert. Credbank: A large-scale social media corpus with associated credibility annotations. *Proceedings of the International AAAI Conference on Web and Social Media*, 9(1):258–267, Aug. 2021.

[14] Qiong Nan, Juan Cao, Yongchun Zhu, Yanyan Wang, and Jintao Li. Mdfend: Multi-domain fake news detection. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, CIKM '21, page 3343–3347, New York, NY, USA, 2021. Association for Computing Machinery.

[15] Ray Oshikawa, Jing Qian, and William Yang Wang. A survey on natural language processing for fake news detection, 2020.

[16] Huyen Trang Phan, Ngoc Thanh Nguyen, and Dosam Hwang. Fake news detection: A survey of graph neural network methods. *Applied Soft Computing*, 139:110235, 2023.

[17] Kai Shu, Deepak Mahudeswaran, Suhang Wang, Dongwon Lee, and Huan Liu. Fakenewsnet: A data repository with news content, social context and spatialtemporal information for studying fake news on social media, 2019.

[18] Kai Shu, Deepak Mahudeswaran, Suhang Wang, and Huan Liu. Hierarchical propagation networks for fake news detection: Investigation and exploitation. *Proceedings of the International AAAI Conference on Web and Social Media*, 14(1):626–637, May 2020.

[19] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. Fake news detection on social media: A data mining perspective. *SIGKDD Explor. Newsl.*, 19(1):22–36, sep 2017.

[20] S. Sivasankari. Tracing the fake news propagation path using social network analysis. *Soft Computing*, 2021.

[21] James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. FEVER: a large-scale dataset for fact extraction and VERification. In *NAACL-HLT*, 2018.

[22] Francesco Vasca, Dora Ricci, and Carmela Bernardo. *Analisi numerica delle reti sociali: Una guida operativa all'uso di MATLAB e R*. 02 2022.

[23] Soroush Vosoughi, Mostafa 'Neo' Mohsenvand, and Deb Roy. Rumor gauge: Predicting the veracity of rumors on twitter. *ACM Trans. Knowl. Discov. Data*, 11(4), jul 2017.

[24] William Yang Wang. "liar, liar pants on fire": A new benchmark dataset for fake news detection, 2017.

[25] Gerhard Widmer and Miroslav Kubat. *Machine Learning*, 23(1):69–101, 1996.

[26] Xinyi Zhou, Apurva Mulay, Emilio Ferrara, and Reza Zafarani. Recovery: A multimodal repository for covid-19 news credibility research. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, CIKM '20, page 3205–3212, New York, NY, USA, 2020. Association for Computing Machinery.

[27] Arkaitz Zubiaga, Ahmet Aker, Kalina Bontcheva, Maria Liakata, and Rob Procter. Detection and resolution of rumours in social media: A survey. *ACM Comput. Surv.*, 51(2), feb 2018.