



Progetto e realizzazione di un sistema
automatizzato per l'identificazione delle fake news
basato su tecniche di analisi semantica, immagini
e contesto

Facoltà di Ingegneria dell'Informazione, Informatica e Statistica
Corso di Laurea Magistrale in Computer Science

Candidato

Lorenzo Manduca
Matricola 1473573

Relatore

Prof. Claudio Cilli

Correlatori

Dott. Fabrizio Venettoni
Dott. Ing. Giulio Magnanini

Anno Accademico 2020/2021

Tesi discussa il 16 April 2013
di fronte a una commissione esaminatrice composta da:
Prof. Nome Cognome (presidente)
Prof. Nome Cognome
Dr. Nome Cognome

Progetto e realizzazione di un sistema automatizzato per l'identificazione delle fake news basato su tecniche di analisi semantica, immagini e contesto
Tesi di Laurea Magistrale. Sapienza – Università di Roma

© 2021 Lorenzo Manduca. Tutti i diritti riservati

Questa tesi è stata composta con L^AT_EX e la classe Sapthesis.

Versione: 21 marzo 2022

Email dell'autore: manduca.1473573@studenti.uniroma1.it

*Alla mia famiglia,
per aver sempre creduto in me,
sostegno fondamentale per il
raggiungimento di questo traguardo.*

Sommario

Ai giorni nostri, le informazioni stanno assumendo un ruolo sempre più centrale nella vita delle persone. Con l'avvento di internet, in particolare, la mole di dati a nostra disposizione è cresciuta esponenzialmente anche in conseguenza della facilità con la quale oggi è possibile condividere contenuti di qualsiasi tipo.

Allo stesso tempo sono però aumentati anche i rischi derivanti dalla mancanza di qualsiasi forma di controllo di veridicità delle informazioni pubblicate. Ciò anche in considerazione del fatto che un contenuto informativo "Fake" può avere seri impatti sulla reputazione di una persona fisica o una società, causando a volte gravi danni alla salute o al patrimonio dei malcapitati, come abbiamo avuto modo di constatare purtroppo anche in recenti fatti di cronaca.

Per cercare di ovviare al problema della verifica di un contenuto informativo "Fake" ed in particolare delle "Fake news" pubblicate su internet, sono stati fatti diversi studi ma, ad oggi, nessuno di questi si è poi tradotto in un progetto concreto realmente significativo. Inoltre, in tutti i progetti finora proposti si delinea un filo conduttore comune, ovvero l'agire esclusivamente sul contenuto e sulla fonte della notizia, esaminando, con il supporto di tecnologie emergenti, la possibile veridicità della stessa con l'obiettivo di fornire una metrica oggettiva e assoluta. D'altra parte, mancando un reale coinvolgimento del lettore, questi progetti non sono risultati mai particolarmente idonei ad aumentare la consapevolezza dell'utente finale. Il presente lavoro ha come obiettivo la realizzazione di una piattaforma tecnologica in grado di calcolare un valore di affidabilità per una qualsiasi notizia pubblicata su internet, a supporto del lettore finale, misurando il livello di imparzialità dell'autore della stessa, attraverso la valutazione di una serie di parametri qualitativi e quantitativi - frutto dell'applicazione di specifiche tecniche di analisi semantica, immagini e contesto che replicano analogicamente i criteri di cui alla c.d. Regola delle cinque W (vedi capitolo successivo) - alcuni dei quali basati anche su valutazioni soggettive fornite dal lettore medesimo.

Indice

1 Introduzione	1
1.1 Cos'è una Fake News	2
1.2 Differenza tra Fake News e altre notizie	2
1.3 Contesto Operativo	3
1.3.1 L'importanza di smascherare le Fake News	3
1.4 Diffusione	6
1.5 Cinque W [44]	7
2 Stato dell'Arte	9
3 Fake News Detector	11
3.1 Specifica dei requisiti	15
3.1.1 Requisiti e Swim-Lane Diagram	15
3.2 Use Case	16
3.2.1 Attori	16
3.2.2 Diagramma UML	17
3.2.3 Specifica Use Case	18
3.2.4 UC-01-01	18
3.2.5 UC-01-01-01	18
3.2.6 UC-01-01-02	19
3.2.7 UC-01-01-03	19
3.2.8 UC-01-01-04	21
3.2.9 UC-01-01-05	21
3.2.10 UC-01-01-06	23
3.2.11 UC-01-01-07	23
3.2.12 UC-01-01-08	24
3.2.13 UC-01-01-09	24
4 Tecnologie utilizzate nel progetto	27
4.1 Web Scraping	27
4.1.1 Goose3	27
4.1.2 Scrapy	28
4.1.3 BeautifulSoup	30
4.2 Estrazione delle località utilizzando NLP	32
4.2.1 spaCy	33
4.2.2 Geopy	35

4.2.3	Formula di Haversine	36
4.3	Estrazione Temporale	37
4.3.1	Pytime extractor	37
4.4	Selenium	37
5	Implementazione	41
5.1	Ambiente di sviluppo	41
5.2	Python 3.7	41
5.3	Spyder	41
5.4	Main.py	43
5.5	Who - UC-01-01-01	45
5.6	Translater - UC-01-01-02	46
5.7	When - UC-01-01-03	46
5.8	What - UC-01-01-04	48
5.9	Where - UC-01-01-05	49
5.9.1	Riconoscimento di entità nominate (NER)	49
5.10	Why - UC-01-06	51
5.11	Count Upper - UC-01-01-07	51
5.12	SearchG - UC-01-01-08	52
5.13	ImageAnalize - UC-01-01-09	53
5.13.1	La codifica URL di Google per la ricerca delle immagini	55
6	Test e conclusioni	57
6.1	Obiettivi e Struttura dei test	57
6.2	Lo Script creato per effettuare i test	57
6.3	Risultati dei test	58
6.4	Conclusioni	59
7	Sviluppi Futuri	61

Capitolo 1

Introduzione

Combattere contro le fake news è come battersi contro un'Idra a più teste mentre si nuota in uno tsunami di melma. (Govindraj Ethiraj) [6]

Le "Fake News" sono un fenomeno che ha da sempre caratterizzato la storia dCosa ell'umanità e che hanno avuto un impatto decisamente importante nella società. Esse fanno parte della storia dei media da molto prima della nascita di internet e dei social media. L'invenzione della stampa di Gutenberg nel 1493 amplificò drammaticamente la diffusione della Disinformazione e della Misinformation, in quanto a quell'epoca era molto difficile verificare la veridicità delle notizie, portando alla produzione della prima "fake news" su larga scala: *"The Great Moon Hoax"* del 1835. Il New York Sun pubblicò sei articoli sulla scoperta della vita sulla luna, completi di illustrazioni di creature come pipistrelli, umanoidi e unicorni blu barbuti. [33]



Figura 1.1.
The Great Moon Hoax del tabloid New York Sun del 1835.
Rappresentazione della superficie lunare.

Nel corso della storia sono presenti molti esempi di fake news. Basti pensare che nel periodo Nazista queste sono state usate dalle macchine di propaganda per costruire ed alimentare il sentimento antisemita, che ha portato alla morte di milioni di Ebrei. Hanno giocato un ruolo fondamentale anche nel periodo dell'Illuminismo in quanto, grazie alla falsa dichiarazione della Chiesa Cattolica sulle cause del terremoto di Lisbona del 1755, hanno spinto Voltaire a parlare contro il dominio religioso dell'epoca. Un altro esempio è rappresentato nel 1800 negli Stati Uniti, dove il sentimento razzista portò alla pubblicazione di false storie su presunti crimini commessi degli afroamericani.

La pubblicazione delle fake news non è casuale. *"Il sensazionalismo ha sempre venduto bene. All'inizio del 19° secolo, i giornali moderni entrarono in scena, pubblicizzando scoop e reportage, ma anche delle storie false per aumentare la circolazione."* [40]

Durante il periodo del 1890, gli editori Joseph Pulitzer e William Hearst, appartenenti a testate giornalistiche rivali, si contesero il pubblico attraverso notizie sensazionalistiche, riportando rumors come se fossero fatti realmente accaduti, una pratica che divenne nota all'epoca come *"giornalismo giallo"*. Le loro notizie ebbero un importante ruolo nel condurre gli Stati Uniti nella guerra ispano-americana del 1898. Alla fine ci fu una reazione popolare contro la mancanza di integrità giornalistica: Il pubblico richiedeva che le fonti delle notizie fossero più obiettive e affidabili possibili. Il giornalismo giallo divenne quindi molto meno comune. Questo fino a quando l'ascesa delle notizie basate sul web non riportò tutto in auge.

Andando a ricercare le cause, una delle motivazioni che hanno portato gli editori dei giornali del 1890 a fare del giornalismo giallo è la stessa dei creatori di fake news odierni: creare notizie esagerate con titoli scioccanti e sensazionali che abbiano lo scopo di attirare l'attenzione e vendere il maggior numero di copie di giornale (o nel caso del web a cliccare sulla notizia), promuovendo anche la pubblicità.

1.1 Cos'è una Fake News

Con Fake News si intendono tutte quelle notizie, storie o bufale create appositamente con lo scopo di disinformare o ingannare deliberatamente i lettori. Queste notizie sono create per influenzare le opinioni delle persone, promuovere la propaganda politica, religiosa o causare confusione generale e spesso possono anche essere un ottimo business redditizio. *"Sono informazioni create ad hoc che replicano il contenuto delle notizie vere nella loro struttura, ma... mancano delle regole e processi editoriali dei media accreditati per garantire l'accuratezza e la credibilità delle informazioni"* [26]. Questa tipologia di informazioni possono infatti ingannare le persone, sembrando provenire da siti web affidabili o usando nomi e indirizzi simili a quelli di editori di notizie rispettabili. [30]

1.2 Differenza tra Fake News e altre notizie

Quello che differenzia le Fake News dalle altre tipologie di notizie è principalmente il loro intento. Come possibile vedere dalla tabella sotto, una Fake News presenta



Figura 1.2. Fake News = Fatto Falso

un'intenzione malevola, creata con il scopo di destabilizzare l'ambiente e creare scompiglio. Al contrario, di una False News, non sappiamo se l'intenzione sia necessariamente malevola o no come anche per la Disinformation e Rumor.

	Autenticità	Intenzione	News?
Fake News	Falsa	Malevola	Si
False News	Falsa	Sconosciuta	Si
Satire News	Sconosciuta	Non malevola	Si
Disinformazione	Falsa	Malevola	Sconosciuta
Misinformation	Falsa	Sconosciuta	Sconosciuta
Rumor	Sconosciuta	Sconosciuta	Sconosciuta

1.3 Contesto Operativo

1.3.1 L'importanza di smascherare le Fake News

Le Fake News ad sono oggi considerate una delle più grandi minacce alla democrazia, alla giustizia, alla fiducia pubblica, alla libertà di espressione, al giornalismo e all'economia. Come anche suggerito dal Papa durante la 55esima Giornata mondiale delle Comunicazioni sociali, è necessario *"Smascherare le fake news"* e *"stare attenti alle insidie del web"*. [25] Queste notizie possono infatti avere un impatto "devastante" sui vari aspetti della società.

Aspetto Politico

- **Referendum per la Brexit:** grazie alla disinformazione effettuata da parte di alcuni dei maggiori quotidiani britannici (come il Daily Mail e il Daily Express) che hanno pubblicato articoli che promuovevano l'odio verso i migranti e l'UE in generale, si è arrivati alla Brexit. [9]
- **Elezioni presidenziali del 2016 negli Stati Uniti:** Un rapporto dell'Australian strategic policy institute ha dimostrato come le fake news abbiano

un'influenza decisiva nel condizionare l'esito delle elezioni. Così fu per il caso delle elezioni presidenziali del 2016 che portano alla vittoria l'ex presidente degli Stati Uniti Donald Trump. Basti pensare che grazie ai social network ci furono circa 8.711.000 tra azioni, reazioni e commenti su Facebook per i 20 titoli FAKE più discussi e solo 7.367.000 per i 20 di storie VERE più discussi. [12]

Aspetto Economico

- **Obama ferito in un'esplosione:** Nel 2013, 130 miliardi di dollari di valore azionario sono stati letteralmente "spazzati via" in pochi minuti dopo un tweet dell'Associate Press (AP) su una presunta "esplosione" che ferì l'allora presidente degli Stati Uniti Barack Obama. AP ha poi successivamente rettificato affermando che il proprio account Twitter fu oggetto di violazione da parte di ignoti. Anche se i prezzi delle azioni si ripresero in poco tempo, questo caso è indicativo di come le notizie che circolano sui social media possano essere manipolate per avere un impatto sugli algoritmi di trading che operano ad alta frequenza e che si basano sugli avvenimenti. [35]
- **Veles Fake News:** Uno dei casi più significativi e redditizi grazie alla diffusione di Fake News. Come detto precedentemente, durante le elezioni presidenziali del 2016, gli americani sono stati esposti a una spaventosa ondata di disinformazione sui social media. Molti dei post di fake news più virali hanno avuto origine da Veles, una piccola città della Macedonia centrale. Si tratta di una città piuttosto povera, con un salario medio mensile netto di circa 380 dollari (ben al di sotto della media nazionale), un tasso di povertà del 22% e un tasso di disoccupazione del 24% [19]. Per potersi arricchire, alcuni cittadini hanno iniziato a pubblicare appositamente fake news sui loro siti web sponsorizzandoli tramite i social, arrivando a guadagnare nel tempo cifre piuttosto significative se si pensa al salario minimo della città. [14] [19]



Figura 1.3. Veles Fake News

Per quanto riguarda invece l'aspetto **Psicologico e Sociale**, per le fake news è relativamente più facile ottenere la fiducia e farsi strada tra le persone grazie a vari aspetti come:

- **Euristica della disponibilità:** denota la tendenza a fidarsi delle fake news dopo ripetute esposizioni di tali elementi e notizie che riprendono situazioni già presenti nella memoria del soggetto, come situazioni analoghe a quelle già lette. Alcuni esempi che riprendono tale euristica sono le notizie riportate dai mass media, le quali fanno leva sull'impatto emotivo e mediatico di alcuni avvenimenti, come ad esempio un incidente aereo. E' noto infatti che gli incidenti aerei siano giudicati come molto più frequenti di quanto non lo siano realmente; questo evento ha un impatto mediatico notevole che condiziona la percezione che i soggetti hanno del rischio portandoli così ad influenzare la decisione futura di effettuare un viaggio con quel mezzo di trasporto. Al contrario, la frequenza degli incidenti stradali mortali è solitamente sottostimata.
- **Bias di conferma:** rappresenta la tendenza a cercare informazioni che supportino, piuttosto che rifiutare, i propri preconcetti, tipicamente interpretando le prove per confermare le credenze esistenti, mentre si rifiutano o si ignorano i dati contrari; questo atteggiamento denota infatti la tendenza nel credere a una fake news quando questa mira a confermare i preconcetti della persona. Ad esempio, nei social media, le informazioni che ci vengono presentate non riflettono solo ciò che gli utenti vogliono vedere, ma anche le credenze e i valori di coloro che li hanno progettati. Oggi, le persone sono esposte a un numero elevato di fonti di notizie, ognuna delle quali varia nella sua credibilità. Per formulare conclusioni, le persone tendono a leggere quelle notizie che meglio si allineano con le loro prospettive. Per esempio, i nuovi canali forniscono informazioni (anche le stesse notizie) in modo diverso l'uno dall'altro su questioni complesse (come razzismo, partiti politici, ecc.), alcuni usando addirittura titoli/immagini sensazionali e informazioni di parte. A causa della copertura parziale degli argomenti, le persone utilizzano solo certi canali/siti per ottenere le loro informazioni e trarne così le conclusioni.
- **Bandwagon effect (effetto carrozzone)** : rappresenta la considerazione secondo cui le persone compiono alcuni atti o credono in alcune cose solo perché la maggioranza della gente crede o fa quelle stesse cose. Ad esempio, in ambito politico, alcune persone tendono a farsi condizionare ed orientare verso la scelta di un candidato politico di cui ne hanno sentito più spesso il nome o che abbia maggiore probabilità di vittoria anche se questo non porta necessariamente nessun beneficio diretto.

Inoltre, le persone fanno spesso uso di "**scorciatoie cognitive**", come ad esempio il fatto di pensare che se una notizia viene ripetuta da molte persone allora sicuramente sarà vera, oppure se proviene da una fonte ritenuta autorevole deve essere necessariamente vera.

1.4 Diffusione

"Le fake news sono sbagliate e difficili da correggere"

Ai giorni d'oggi, grazie all'ascesa e la popolarità dei Social Media che ne hanno accelerato la diffusione, le fake news hanno sempre di più un impatto significativo sulle nostre vite. Data la prevalenza di reti di blog, social network e app che vengono usate come fonte di notizie per la maggior parte degli individui, le fake news riescono a diffondersi a ritmo velocissimo. Tutto ciò rende il loro rilevamento una sfida estremamente difficile.

Essendo arduo correggere le percezioni degli utenti dopo che le fake news hanno guadagnato la loro fiducia, diventa fondamentale il rilevamento in anticipo delle fake news. Malgrado l'impegno di uno dei Social Network più diffusi (Facebook) che ha annunciato *"l'adozione di misure straordinarie per limitarne la diffusione"* [43], sembra che la disinformazione sia nettamente più popolare rispetto agli anni passati con una percentuale che si è addirittura triplicata dal 2016 ai giorni d'oggi. Molto preoccupante è anche la percentuale di siti che non verifica fonti e pubblica notizie in modo irresponsabile, cresciuta del 293% [11].

Un sondaggio condotto nel marzo 2019 ha valutato se i cittadini adulti negli Stati Uniti avessero mai condiviso consapevolmente o inconsapevolmente notizie o informazioni false online. Una grande percentuale di intervistati ha riferito di aver diffuso inconsapevolmente notizie false. Il 49% ha invece riferito di aver condiviso notizie online che hanno poi scoperto essere inventate. Mentre, solo il 10% degli intervistati ha ammesso di aver condiviso informazioni online che sapevano essere false.

Ma il web non è l'unico posto dove si diffondono. Ancora oggi infatti, molte forme di fake news sono presenti anche nei giornali cartacei. Il 47% degli intervistati, ha affermato di aver trovato fake news su giornali e riviste a gennaio 2019. Come già osservato, se non correttamente contrastate, queste notizie possono anche provocare gravi danni all'incolinità delle persone. E' il caso di alcuni uomini in India che sono stati linchiati dalla folla dopo essere stati ingiustamente accusati di rapire dei bambini grazie ad un video fake condiviso su whatsapp. [20]

In definitiva quindi possiamo affermare che sempre più persone usano i social media come mezzi di informazione, non verificando se le informazioni che leggono sono vere oppure fake. Grazie ad essi abbiamo un'accelerazione della diffusione delle fake news. I social media, inoltre, hanno accelerato l'evoluzione delle fake news favorendo il cosiddetto **effetto Echo Chamber** in virtù del quale le informazioni distorte possono essere amplificate e rafforzate all'interno dei social media, ovvero all'interno di un sistema chiuso [21] portando anche alla conferma dei propri pregiudizi (**"È una bufala, ma sicuramente prima o poi succederà qualcosa di simile o peggio!"**

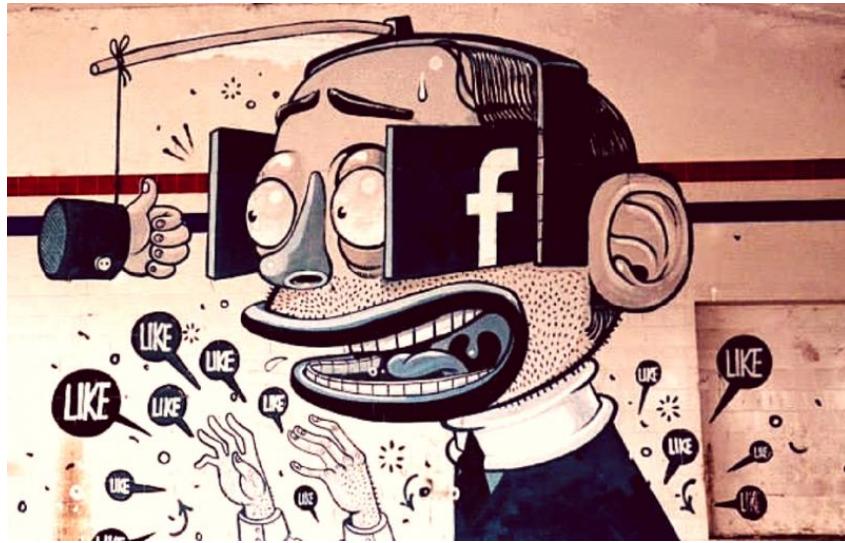


Figura 1.4. Echo Chamber Effect

1.5 Cinque W [44]

Per potersi difendere dalle Fake News sarebbe necessario fermarsi, ragionare sulla notizia stessa e porsi alcune domande specifiche. Ciò è quanto suggerisce la cosiddetta regola delle 5 W, un metodo attraverso il quale rispondendo a determinate domande è possibile individuare le caratteristiche **fondamentali** di un fatto o un problema. Usata anche dai giornalisti per scrivere i propri articoli [47], la regola può essere usata anche per valutare il grado di attendibilità di una notizia.

In particolare, una notizia può essere considerata completa solo se risponde a queste domande:

- **WHO:** E' presente un autore della notizia?
- **WHAT:** Di che tipo di notizia stiamo parlando? Si tratta di un fatto di cronaca, di propaganda elettorale, della definizione di un dogma, una massima, ecc.?
- **WHEN:** Quando è stata scritta la notizia? E quando sono successi i fatti riportati nella notizia?
- **WHERE:** Dov'è il luogo in cui si è verificato l'avvenimento descritto dalla notizia?
- **WHY:** Qual è la motivazione che ha spinto l'autore a pubblicare la notizia? Cronaca, informativa, economica, sociale/politico, ecc.

Nel successivo Capitolo 3 vedremo come tradurre questi elementi in specifici criteri di valutazione volti a misurare il grado di attendibilità di una notizia.

Capitolo 2

Stato dell'Arte

Nel primo capitolo di questa tesi si è ampiamente discusso di come le fake news siano considerate una delle più grandi minacce alla democrazia, alla giustizia, alla fiducia pubblica, alla libertà di espressione, al giornalismo e all'economia. Al giorno d'oggi vi è un'ampia letteratura scientifica sull'argomento, che comprende numerosi articoli volti alla risoluzione del problema attraverso l'utilizzo di quasi tutte le tecnologie emergenti, quali ad esempio il Machine o Deep Learning, le reti neurali, i sistemi multimodali, la blockchain e chi più ne ha più ne metta.

In tutti i progetti finora proposti si delinea però un filo conduttore comune, ovvero l'agire esclusivamente sul contenuto e sulla fonte della notizia, esaminando la veridicità della stessa senza prendere in seria considerazione il destinatario finale, il lettore. Anche per questo a nostro avviso tali iniziative non sono mai diventate progetti commerciali significativi.

Si riporta di seguito l'elenco dei principali lavori esaminati:

- **MVAE:** Multimodal Variational Autoencoder for Fake News Detection che propone, per il rilevamento delle fake news, un autoencoder a variazionale bimodale accoppiato con un classificatore binario multimodale (testuale + visivo); [23]
- **DeHiDe:** Modello ibrido che combina la tecnologia blockchain con un modello intelligente di deep learning per rafforzare la solidità e la precisione nel combattere le fake news; [4]
- **dEFEND:** un sistema per il rilevamento di fake news che sfrutta i commenti degli utenti per verificare se la notizia sia falsa o reale. [36]
- **Fake News Early Detection: A Theory-driven Model:** In questo articolo, viene proposto, partendo dalla teoria, un modello guidato per il rilevamento delle fake news. Il metodo proposto vuole indagare sul contenuto delle notizie a vari livelli: lessicale, sintattico, semantico e del discorso. Le notizie vengono rappresentate ad ogni livello, basandosi sulle teorie consolidate nella psicologia sociale e forense. Il rilevamento delle fake news viene poi condotto all'interno di un quadro di apprendimento automatico supervisionato. Questo lavoro esplora potenziali modelli di fake news, migliorando l'interpretabilità nell'ingegneria delle caratteristiche delle fake news studiandone vari aspetti. [49]

Oltre alla letteratura scientifica, si evidenziano alcuni tool e utility come:

- **Fiskkit**: piattaforma creata da John Pettus che si propone di costruire un luogo per promuovere la coerenza e la neutralità delle informazioni tramite invio da parte dei partecipanti di feedback per aiutare le persone a identificare ciò che sia vero, falso, ben argomentato o logicamente scorretto in articoli o opinioni. [32]
- **TextThresher**: interfaccia web che perfeziona la pratica delle scienze sociali dell'analisi dei contenuti, rendendola più trasparente e scalabile a centinaia di migliaia di documenti. [3]
- **FakeNewsTracker**: strumento per la raccolta, il rilevamento e la visualizzazione di fake news, che utilizza dataset e modelli di ML estraendone le caratteristiche utili. [37]
- **FakeNewsNet**: una repository di dati con informazioni sul contenuto di notizie, contesto sociale e informazioni spazio-temporali per lo studio delle Fake News sui social media. [38]
- **Detecting Fake News in Social Media Networks**: Lo scopo del seguente lavoro è stato quello di trovare una soluzione che possa essere utilizzata dagli utenti per individuare e filtrare i siti che contengono informazioni false e fuorviante. Usando caratteristiche semplici e accuratamente selezionate del titolo e del post è possibile, grazie all'utilizzo del tool, identificare i post falsi. [5]
- Vale la pena segnalare, tra i progetti in via di sviluppo, **SocialTruth** [2]. Questo progetto europeo vuole fornire un modo innovativo e distribuito, grazie alla tecnologia BlockChain e Machine Learning, di ottenere sia la verifica della credibilità del contenuto e dell'autore che il rilevamento di fake news, al fine di aumentare la fiducia nei Social Media. Tuttavia, il progetto iniziato nel 2019, risulta ancora in fase di realizzazione, senza alcun risultato rilevante sinora pubblicato.

Capitolo 3

Fake News Detector

Fake News Detector ha come obiettivo la definizione di un tool automatico per il rilevamento delle Fake News. A differenza dei progetti sopra citati, si è cercato di incentrare e coinvolgere al massimo l'utente senza lavorare solamente sul contenuto o sulla fonte della notizia, cercando così di svilupparne la sua consapevolezza. Il punto di forza dello strumento è la concreta applicazione della regola delle 5W, alla quale abbiamo aggiunto altri tre principi (denominati "How", vedi appresso), attraverso la definizione di otto metriche di valutazione (una per ogni criterio, potremo parlare anche della regola delle 5W e delle 3H), come specificato di seguito:

- **WHO:** questo criterio ha come obiettivo la verifica della presenza o meno dell'autore della notizia. Vuole rispondere alla domanda: "Esiste un autore per la notizia da analizzare?". In base alla risposta, il tool assegna un punteggio 0/1 nel seguente modo:
 - a. [0] Non è presente l'autore della notizia
 - b. [1] La notizia è attribuibile ad un autore esplicitato

Generalmente, una Fake news, non presenta un autore.[1]

- **WHAT:** questo principio ha come obiettivo la ricerca dell'argomento della notizia. In base al tipo di argomento, il tool assegna un punteggio da 0 a 1 nel seguente modo:
 - a. [0] Non è possibile classificare l'argomento della notizia
 - b. [0.5] Dogma, massima, discorso, propaganda politica, propaganda commerciale, satira, provocazione
 - c. [1] Fatto di Cronaca, pubblicazione scientifica

Una notizia considerata attendibile è molto spesso attribuibile ad un argomento ben definito.

- **WHEN:** questo elemento di valutazione si prefigge di identificare la data di avvenimento dei fatti avvenuti nella notizia. Il tool assegna un punteggio da 0 a 1 nel seguente modo:
 - a. [0] Se non è presente nessuna data di riferimento della notizia

- b. [1] Se presente in forma esplicita

Una fake news generalmente non presenta date precise. Pertanto, è possibile attribuire un punteggio elevato ad una notizia solamente se la data è presente in forma esplicita.

Oltre ai punti sopra citati, si aggiunge come fattore per la determinazione del When anche la vicinanza temporale rispetto alla data di pubblicazione. Si assegna il seguente punteggio:

- a. [0] Se i fatti della notizia si riferiscono ad un periodo antecedente 180 giorni (circa 6 mesi) dalla data di valutazione
- b. [0.15] Se i fatti della notizia si riferiscono ad un periodo antecedente 2 anni dalla data di valutazione
- c. [0.35] Se i fatti della notizia si riferiscono ad un periodo antecedente più 2 anni dalla data di valutazione

Altra caratteristica molto importante quando si valuta una notizia è la data di avvenimento dei fatti riportati. Tendenzialmente, più la data di avvenimento dei fatti è vicina alla data odierna, più aumenta la probabilità che si tratti di una fake news costruita ad hoc per ingannare il lettore. Chi legge la notizia tende ad attribuire alla stessa una credibilità maggiore se i fatti presenti in essa sono avvenuti in un periodo relativamente vicino, in quanto il tema è attuale; mentre, più ci allontaniamo temporalmente dalla data odierna, più diminuisce la probabilità che la notizia sia una fake news (ma anche se lo fosse, la stessa non avrebbe più lo stesso effetto mediatico di una notizia recente)

- **WHERE:** questa metrica ha come obiettivo quello di identificare il luogo dove si sono svolti i fatti e attribuire la vicinanza con il luogo dove risiede colui che intende effettuare la verifica. Il tool assegna un punteggio da 0 a 1 nel seguente modo:

- a. [0] Se non esiste o non è possibile identificare il luogo di avvenimento dei fatti
- b. [1] Se è possibile attribuire il luogo di avvenimento dei fatti

Una notizia considerata attendibile avrà un luogo di avvenimento dei fatti ben definito

Inoltre, verrà assegnato un ulteriore punteggio secondo i seguenti parametri:

- a. [0] Se la notizia è distante meno di 1650 KM
- b. [0.15] Se la notizia è a medio raggio dal luogo dove risiede colui che effettua la verifica (distanza compresa tra 1650 KM e 6000 KM)
- c. [0.35] Se la notizia è a lungo raggio dal luogo dove risiede colui che effettua la verifica (distanza maggiore di 6000 KM)

E' importante fare un'analisi della distanza che intercorre tra colui che effettua la ricerca della notizia e il luogo di avvenimento della stessa. Tendenzialmente,

più la distanza di avvenimento dei fatti è vicina al luogo dove risiede colui che sta effettuando la ricerca, più aumenta la probabilità che si tratti di una fake news. Chi la legge avrà maggiore interesse a dare importanza ad una notizia che si svolge in un luogo relativamente vicino al suo.

- **WHY:** questo criterio si prefigge di ricercare l'obiettivo per il quale la notizia è stata pubblicata. Intende rispondere alla domanda: "Perchè colui che ha pubblicato o condiviso la notizia lo ha fatto? Quale scopo ha ottenuto? Quali emozioni mi ha suscitato?". In base al tipo di risposta il tool assegna un punteggio da 0 a 1 nel seguente modo:
 - a. [0] Se lo scopo di colui che ha pubblicato la notizia è suscitare emozioni al lettore o per indurlo a comprare un bene/effettuare un'attività
 - b. [1] Se lo scopo di colui che ha pubblicato la notizia è informare in maniera disinteressata

Se una notizia suscita nel lettore delle forti emozioni (paura, rabbia, sgomento, ecc) o lo induce a cambiare la propria idea/opinione o ad effettuare un'azione che prima non avrebbe fatto, allora la notizia sarà molto probabilmente una fake news creata con uno scopo ben specifico. Generalmente, una notizia reale che ha il solo scopo di informare il lettore, dovrebbe suscitare in colui che la legge un sentimento neutro.

Prima di passare in rassegna le tecnologie utilizzate (cfr. capitolo successivo), di seguito una descrizione sintetica delle specifiche tecniche della POC del sistema realizzata nel presente lavoro.

- **WHO:** Per verificare l'esistenza dell'autore si utilizza uno scraper web con l'obiettivo di estrarre lo stesso dal codice xml della pagina.
- **WHAT:** Per riconoscere l'argomento della notizia, il tool nell'attuale versione chiede all'utente di selezionare una delle opzioni da un menù a tendina appositamente progettato.
- **WHEN:** Attraverso un algoritmo di "String Matching" il sistema cerca la data alla quale la notizia si riferisce all'interno del suo contenuto; se questa non è esplicitamente presente, attraverso un analizzatore semantico si cercano nomi riconducibili a periodi noti dell'anno (es. Natale, Capodanno, Inverno, Estate, ecc). Una volta determinata la data, si calcola la distanza dalla data di valutazione e si assegnano i parametri come sopra definito.
- **WHERE:** Grazie ad un analizzatore semantico allenato su un dataset di luoghi geografici, si estrae la presenza del luogo. Una volta estratto, attraverso le API di Nominatim, si calcola la distanza in km rispetto all'indirizzo dichiarato da colui che effettua la ricerca.
- **WHY:** Al fine di risolvere questo punto, viene proposto un menù a tendina appositamente progettato.

Oltre ai parametri forniti dalle 5W, sono stati presi in esame altri tre criteri, come di seguito specificato:

- **HOW 1 - Analisi del rapporto tra Maiuscole e Minuscole:** Attraverso un analizzatore sintattico il sistema effettua l'analisi del rapporto tra lettere maiuscole e minuscole presenti nel titolo in modo da attribuire un valore da utilizzare come ulteriore parametro per la valutazione della notizia. Il tool assegna un punteggio 0/1 nel seguente modo:
 - [0] Se la notizia ha un rapporto tra maiuscole e minuscole superiore al 10%
 - [1] Se la notizia ha un rapporto tra maiuscole e minuscole inferiore al 10%

E' il caso del cosiddetto "Titolo Urlato", che ha come solo scopo quello di richiamare l'attenzione del lettore. Una notizia reale non ha la necessità di inserire nel testo del titolo una percentuale elevata di lettere Maiuscole

- **HOW 2 - Analisi dei riferimenti esterni ai fatti descritti nella notizia:** Attraverso l'automatizzazione della ricerca su Google, Fake News Detector propone all'utente una serie di link in riferimento ai fatti della notizia. Se l'utente ne riconosce una corrispondenza con i fatti presenti nella notizia stessa allora il tool assegna 1 alla notizia, altrimenti assegna 0.
 - [0] Se la notizia non ha nessun riferimento oltre al sito stesso
 - [1] Se la notizia ha un riferimento su una fonte diversa rispetto al sito stesso

Una fake news, generalmente, non viene riportata da altre fonti ma solo dal sito di pubblicazione.

- **HOW 3 - Presenza di immagini fuorvianti:** Attraverso l'applicazione della c.d. *reverse image search* di un motore di ricerca, es. Google, si effettua l'analisi delle immagini presenti nella notizia in modo da verificare se queste siano coerenti con la stessa. Il tool assegna un valore compreso nell'intervallo [0,1] nel seguente modo:

$$scoreImmagini = \left(\frac{\sum_{i=1}^n scorePar(i)}{n} \right) \text{ dove}$$

- n = numero totale delle immagini
- $scorePar(i)$ = score assegnato all'immagine i avente dominio $\{0,1\}$ attraverso il seguente algoritmo: Per ogni immagine con indice i da $i = 1$ a n : se l'utente ritrova il riferimento corretto all'immagine i nella ricerca correlata identificata da Google, allora $scorePar(i) = 1$; altrimenti: se l'utente ritrova un link correlato all'immagine (tra i link mostrati da Google nella Reverse Image Search) che rispecchia i fatti della notizia, allora $scorePar(i) = 1$; altrimenti: $scorePar(i) = 0$.

Sono numerose le volte in cui all'interno della notizia vengono inserite delle immagini non pertinenti o modificate; tutto ciò non avrebbe alcun senso se la notizia fosse reale

3.1 Specifica dei requisiti

3.1.1 Requisiti e Swim-Lane Diagram

Il sistema implementa le funzioni descritte nello Swim-Lane Diagram seguente:

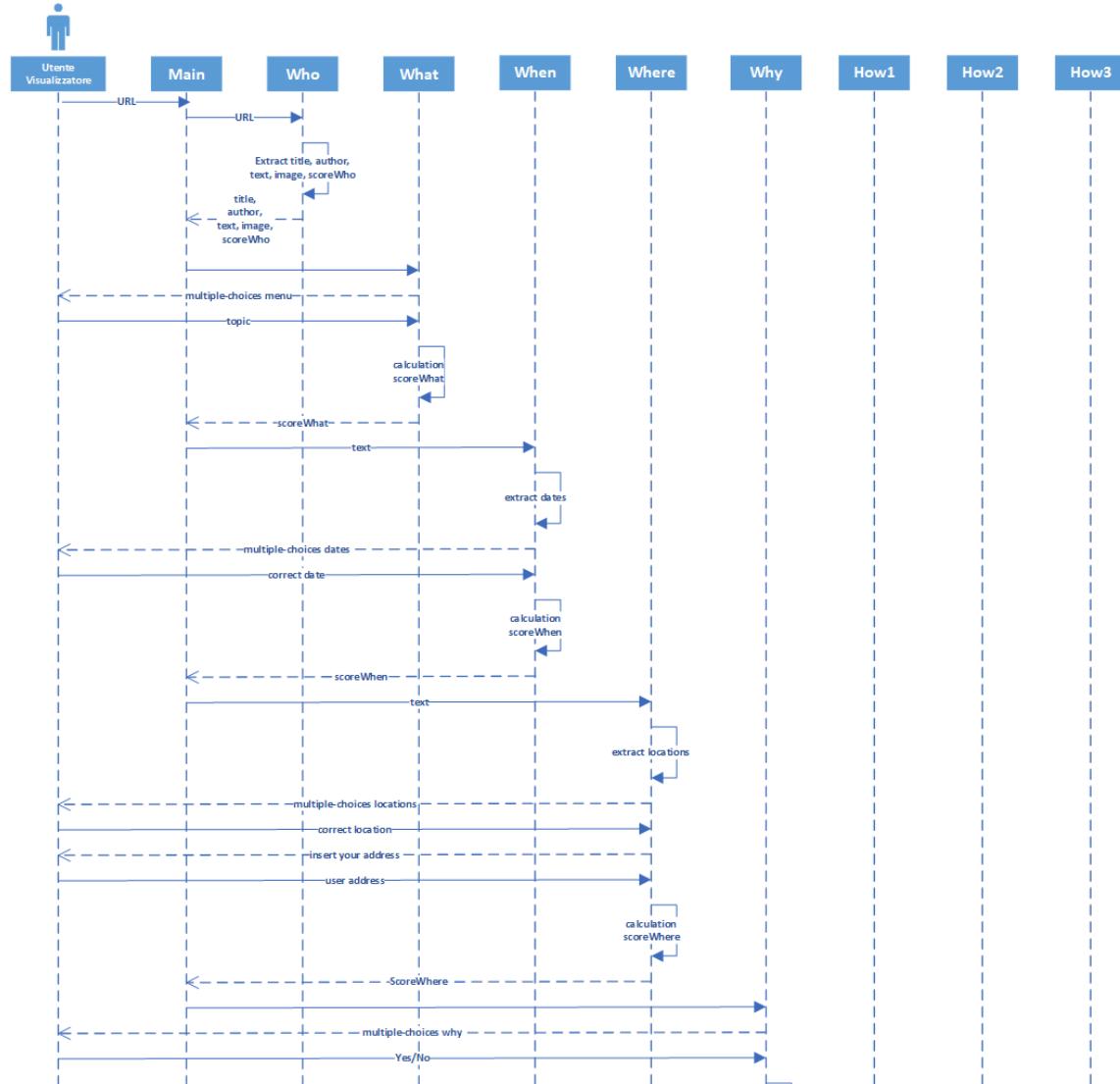


Figura 3.1. Swim-Lane Diagram - Parte 1

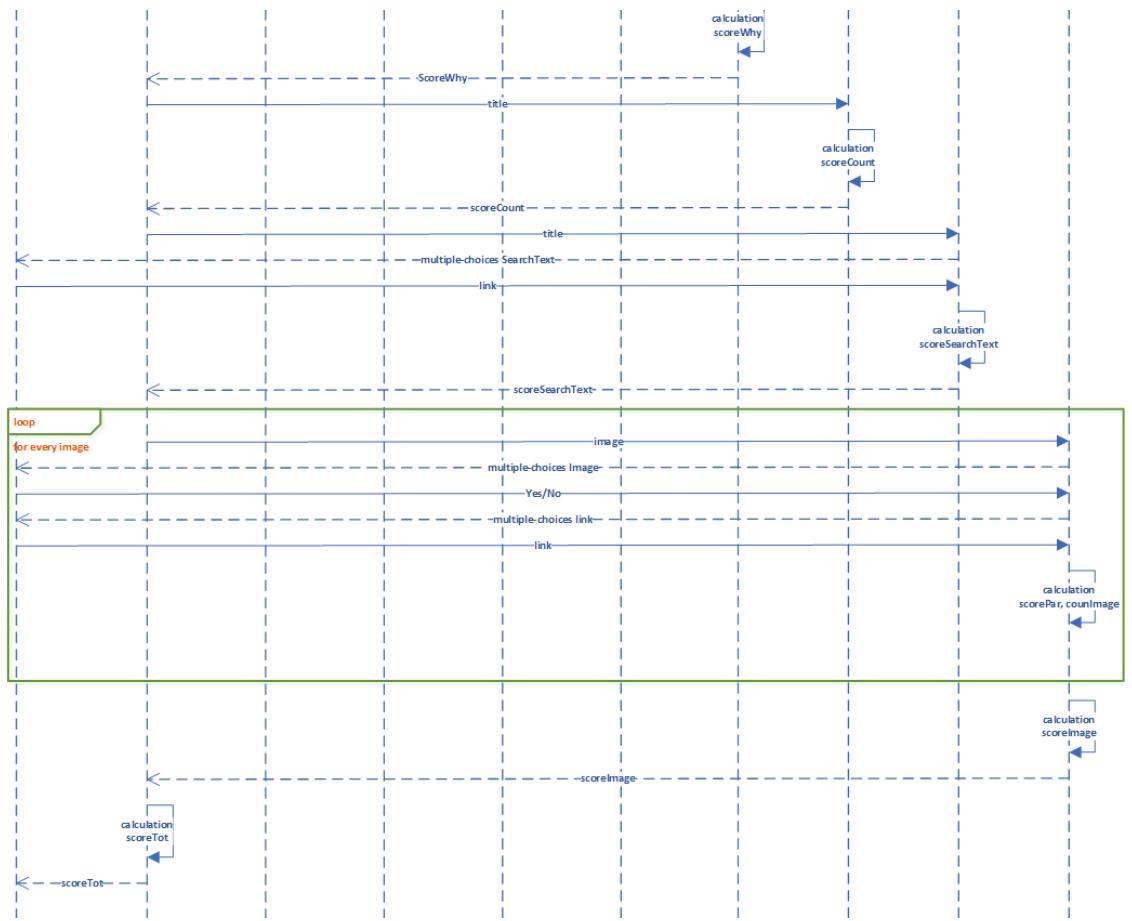


Figura 3.2. Swim-Lane Diagram - Parte 2

3.2 Use Case

In questa sezione viene fornito l'elenco dei casi d'uso e degli attori che interagiscono con il sistema. Ognuno di essi è associato ad un attore che è la persona che interagisce con la funzione. Ogni caso d'uso è rappresentato con la notazione UML. Gli elementi rappresentati in questa notazione sono:

1. "Stickman" per gli attori.
2. Ovali per casi d'uso e relativi identificatori. Ogni attore può comunicare con uno o più di questi.
3. Linee di connessione per gli attori associati ai casi d'uso o casi d'uso ai casi d'uso che determinano una relazione.

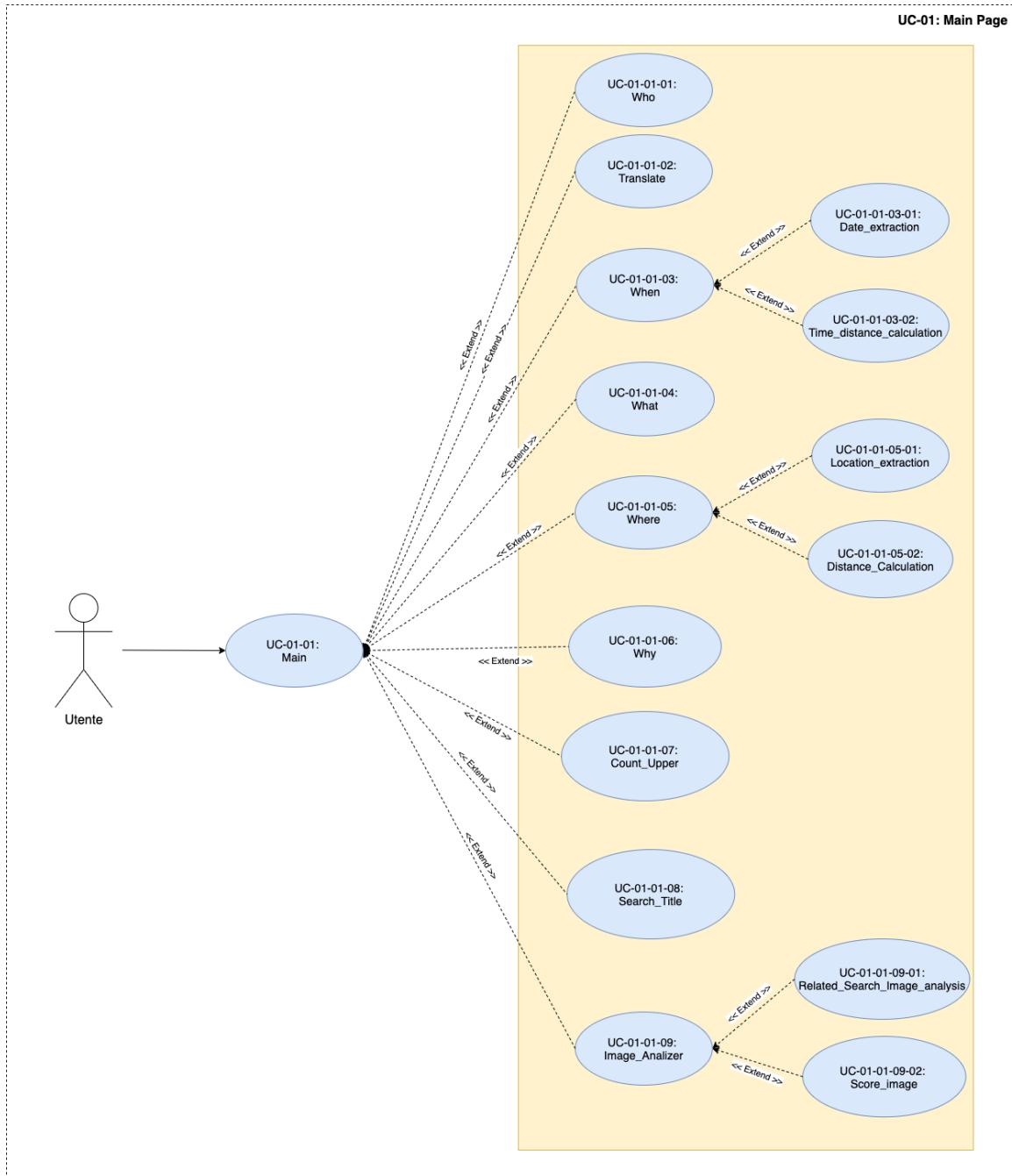
3.2.1 Attori

E' stato individuato un'unica tipologia di attore che interagisce con il sistema. L'attore del sistema è definito come:

- Utente Verificatore

ID	ACT-01
Nome dell'attore	Utente Verificatore
Genitore	-
Descrizione	Colui che intende verificare una notizia presente sotto forma di URL calcolandone un valore di veridicità

3.2.2 Diagramma UML



3.2.3 Specifica Use Case

3.2.4 UC-01-01

ID	UC-01-01
Nome	Main
Attori	Utente Verificatore
Precondizioni	L'utente entra nel tool
Flusso Principale	<ol style="list-style-type: none"> 1. L'utente inserisce l'URL della notizia da verificare 2. Il tool verifica il formato dell'URL 3. Il sistema chiama gli altri UC 4. Il sistema calcola lo score totale della notizia
Post Condizioni	Viene restituito lo score totale della notizia
Flusso Alternativo	In ogni momento l'utente può interrompere l'operazione
Post Condizioni	Nessun dato viene salvato
Flusso Alternativo	

3.2.5 UC-01-01-01

ID	UC-01-01-01
Nome	Who
Attori	Utente Verificatore
Precondizioni	L'utente entra nel tool ed ha eseguito lo UC-01-01
Flusso Principale	<ol style="list-style-type: none"> 1. Se URL è corretto: <ol style="list-style-type: none"> a) Lo scraper estrae dalla pagina web il Titolo, l'autore, il testo e scarica le immagini b) Se viene rilevato un autore: <ol style="list-style-type: none"> a1) Il tool assegna il valore 1 allo score dell'autore b1) else: <ol style="list-style-type: none"> a1) Il tool assegna il valore 0 allo score dell'autore 4. else: <ol style="list-style-type: none"> 4.1 il sistema notifica un errore
Post Condizioni	Viene aggiornato lo score totale della notizia, vengono scaricate le immagini presenti nell'url e vengono restituiti titolo, autore e testo.
Flusso Alternativo	In ogni momento l'utente può interrompere l'operazione
Post Condizioni	Nessun dato viene salvato
Flusso Alternativo	

3.2.6 UC-01-01-02

ID	UC-01-01-02
Nome	Translate
Attori	Utente Verificatore
Precondizioni	<ol style="list-style-type: none"> 1. L'utente ha inserito l'url ed è stato eseguito il caso d'uso UC-01-01-01 2. E' stato restituito il Testo dell'URL
Flusso Principale	<ol style="list-style-type: none"> 1. Il tool verifica il testo 2. Se il testo è in lingua inglese: <ol style="list-style-type: none"> a) il testo non viene tradotto 3. else: b) il testo viene tradotto in inglese
Post Condizioni	Viene restituito il testo in lingua inglese
Flusso Alternativo	In ogni momento l'utente può interrompere l'operazione
Post Condizioni	Nessun dato viene salvato
Flusso Alternativo	

3.2.7 UC-01-01-03

ID	UC-01-01-03
Nome	When
Attori	Utente Verificatore
Precondizioni	<ol style="list-style-type: none"> 1. L'utente ha inserito l'url ed è stato eseguito il Caso D'uso UC-01-01-01 2. Il testo dell'url è in lingua inglese
Flusso Principale	<ol style="list-style-type: none"> 1. Il tool esegue il caso d'uso UC-01-01-03-01 2. Il tool esegue il caso d'uso UC-01-01-03-02 se viene estratta e selezionata una data
Post Condizioni	La funzione aggiorna lo score totale
Flusso Alternativo	In ogni momento l'utente può interrompere l'operazione
Post Condizioni	Nessun dato viene salvato
Flusso Alternativo	

3.2.7.1 UC-01-01-03-01

ID	UC-01-01-03-01
Nome	Date extraction
Attori	Utente Verificatore
Precondizioni	<ul style="list-style-type: none"> 1. L'utente ha inserito l'url ed è stato eseguito il Caso D'uso UC-01-01-01 2. Il testo dell'url è in lingua inglese
Flusso Principale	<ul style="list-style-type: none"> 1. Il tool elimina punteggiatura e caratteri speciali nel testo 2. Il Tool avvia l'estrazione di date e forme lessicali che riconducono a periodi temporali 3. L'utente verificatore seleziona il periodo temporale più appropriato rispetto ai fatti della notizia. 4. Se l'utente verificatore sceglie una data: <ul style="list-style-type: none"> a1) Il tool assegna il valore 1 allo score della data a2) Il tool assegna il valore 0 allo score della data 5. else:
Post Condizioni	La funzione aggiorna lo score della data e ritorna la data estratta e scelta dall'utente
Flusso Alternativo	In ogni momento l'utente può interrompere l'operazione
Post Condizioni	Nessun dato viene salvato
Flusso Alternativo	

3.2.7.2 UC-01-01-03-02

ID	UC-01-01-03-02
Nome	Time distance calculation
Attori	Utente Verificatore
Precondizioni	L'utente ha eseguito il caso d'uso UC-01-01-03-01 ed è stata restituita una data
Flusso Principale	<ul style="list-style-type: none"> 1. Il tool calcola la distanza in giorni tra la data odierna e la data restituita nelle precondizioni 2. Se la distanza in giorni è minore di 180 giorni: <ul style="list-style-type: none"> a1) Il tool aggiunge il valore 0 allo score della data 3. Se la distanza in giorni è compresa tra 180 e 730: <ul style="list-style-type: none"> a2) Il tool aggiunge il valore 0,15 allo score della data 4. else <ul style="list-style-type: none"> a3) Il tool aggiunge il valore 0,35 allo score della data
Post Condizioni	La funzione aggiorna lo score della data
Flusso Alternativo	In ogni momento l'utente può interrompere l'operazione
Post Condizioni	Nessun dato viene salvato
Flusso Alternativo	

3.2.8 UC-01-01-04

ID	UC-01-01-04
Nome	What
Attori	Utente Verificatore
Precondizioni	-
Flusso Principale	<ol style="list-style-type: none"> 1. L'utente verificatore seleziona uno degli argomenti proposti dal menù a tendina che gli viene mostrato 2. Se l'argomento selezionato è uno tra "News Item" o "Scientific Publication": <ul style="list-style-type: none"> a1) Il tool assegna il valore di 1 allo score dell'argomento 3. Se l'argomento selezionato è "No topic can be identified": <ul style="list-style-type: none"> a2) Il tool assegna il valore di 0 allo score dell'argomento 4. else: <ul style="list-style-type: none"> a3) il tool assegna il valore di 0,5 allo score dell'argomento
Post Condizioni	La funzione aggiorna lo score totale
Flusso Alternativo	In ogni momento l'utente può interrompere l'operazione
Post Condizioni	Nessun dato viene salvato
Flusso Alternativo	

3.2.9 UC-01-01-05

ID	UC-01-01-05
Nome	Where
Attori	Utente Verificatore
Precondizioni	<ol style="list-style-type: none"> 1. L'utente ha inserito l'url ed è stato eseguito il Caso D'uso UC-01-01-01 2. Il testo dell'url è in lingua inglese
Flusso Principale	<ol style="list-style-type: none"> 1. Il tool esegue il caso d'uso UC-01-01-05-01 2. Il tool esegue il caso d'uso UC-01-01-05-02 se viene estratta e selezionata una località
Post Condizioni	La funzione aggiorna lo score totale
Flusso Alternativo	In ogni momento l'utente può interrompere l'operazione
Post Condizioni	Nessun dato viene salvato
Flusso Alternativo	

3.2.9.1 UC-01-01-05-01

ID	UC-01-01-05-01
Nome	<u>Location extraction</u>
Attori	Utente Verificatore
Precondizioni	<ol style="list-style-type: none"> 1. L'utente ha inserito <u>l'url</u> ed è stato eseguito il Caso D'uso UC-01-01-01 2. Il testo <u>dell'url</u> è in lingua inglese
Flusso Principale	<ol style="list-style-type: none"> 1. Il <u>Tool</u> avvia l'estrazione delle località o luoghi di interesse 2. L'utente verificatore seleziona il luogo più appropriato rispetto ai fatti della notizia. 3. Se l'utente ha selezionato almeno un luogo relativo ai fatti della notizia: <ol style="list-style-type: none"> a1) assegna il valore 1 allo score del luogo b1) estrae latitudine e longitudine della località selezionata 5. else: <ol style="list-style-type: none"> a2) Il <u>tool</u> assegna il valore 0 allo score della località
Post Condizioni	La funzione aggiorna lo score della località e ritorna la latitudine e longitudine della località estratta e scelta dall'utente
Flusso Alternativo	In ogni momento l'utente può interrompere l'operazione
Post Condizioni	Nessun dato viene salvato
Flusso Alternativo	

3.2.9.2 UC-01-01-05-02

ID	UC-01-01-05-02
Nome	<u>Distance calculation</u>
Attori	Utente Verificatore
Precondizioni	L'utente ha eseguito il caso d'uso UC-01-01-05-01 e sono state restituite latitudini e longitudini di una località
Flusso Principale	<ol style="list-style-type: none"> 1. L'utente inserisce la località di dove si trova 2. Il <u>tool</u> calcola la <u>latitudine</u> e <u>longitudine</u> della località dell'utente inserita al punto 1 3. Il <u>tool</u>, attraverso la formula di <u>Haversine</u>, calcola la distanza in KM tra la latitudine e longitudine della località calcolata al punto 2 e la latitudine e longitudine della località estratta dal caso d'uso UC-01-01-05-01 4. Se la distanza in KM è minore di 1650: <ol style="list-style-type: none"> a1) Il <u>tool</u> aggiunge il valore 0 allo score della località 3. Se la distanza in KM è compresa tra 1650 e 6000: <ol style="list-style-type: none"> a2) Il <u>tool</u> aggiunge il valore 0,15 allo score della località 4. else <ol style="list-style-type: none"> a3) Il <u>tool</u> aggiunge il valore 0,35 allo score della località
Post Condizioni	La funzione aggiorna lo score totale
Flusso Alternativo	In ogni momento l'utente può interrompere l'operazione
Post Condizioni	Nessun dato viene salvato
Flusso Alternativo	

3.2.10 UC-01-01-06

ID	UC-01-01-06
Nome	Why
Attori	Utente Verificatore
Precondizioni	-
Flusso Principale	<ol style="list-style-type: none"> 1. L'utente verificatore seleziona una risposta alla domanda "Does the article would like to convince you to buy some product or to change your opinion about some fact/person?" dal menù a tendina che gli viene mostrato 2. Se la risposta alla domanda è "si" <ol style="list-style-type: none"> a1) Il tool assegna il valore 0 allo score del why 3. Se la risposta alla domanda è "no" <ol style="list-style-type: none"> a2) il tool assegna il valore di 1 allo score del why
Post Condizioni	La funzione aggiornerà lo score totale della notizia
Flusso Alternativo	In ogni momento l'utente può interrompere l'operazione
Post Condizioni	Nessun dato viene salvato
Flusso Alternativo	

3.2.11 UC-01-01-07

ID	UC-01-01-07
Nome	Count_Upper
Attori	Utente
Precondizioni	<ol style="list-style-type: none"> 1. Il programma prende in input il titolo normalizzato dallo UC-01-01-01
Flusso Principale	<ol style="list-style-type: none"> 1. Il programma calcola il rapporto tra il numero di lettere maiuscole ed il numero di lettere totali presenti nel titolo estratto 2. Se questo rapporto è < del 10% del totale: <ol style="list-style-type: none"> a1) il tool assegna il valore 1 allo score del Count_Upper 3. else <ol style="list-style-type: none"> a2) il tool assegna il valore 0 allo score del Count_Upper
Post Condizioni	La funzione aggiornerà lo score totale della notizia
Flusso Alternativo	In ogni momento l'utente può interrompere l'operazione
Post Condizioni	Nessun dato viene salvato
Flusso Alternativo	

3.2.12 UC-01-01-08

ID	UC-01-01-08
Nome	Search Title
Attori	Utente Verificatore
Precondizioni	<ul style="list-style-type: none"> 1. L'utente verificatore ha eseguito il caso d'uso UC-01-01-01 2. Il caso d'uso UC-01-01-01 ha estratto il titolo
Flusso Principale	<ul style="list-style-type: none"> 1. Il tool simula una ricerca sul motore di ricerca "Google" inserendo come campo della ricerca il titolo 2. Il tool chiede all'utente di selezionare un link con il relativo titolo che più si avvicina ai fatti riportati nell'articolo 3. Se l'utente seleziona un link/titolo: <ul style="list-style-type: none"> a1) il tool assegna valore 1 allo score titolo a2) il tool assegna valore 0 allo score titolo
Post Condizioni	La funzione aggiornerà lo score totale della notizia
Flusso Alternativo	In ogni momento l'utente può interrompere l'operazione
Post Condizioni	Nessun dato viene salvato
Flusso Alternativo	

3.2.13 UC-01-01-09

ID	UC-01-01-09
Nome	Image analizer
Attori	Utente Verificatore
Precondizioni	<ul style="list-style-type: none"> 1. L'utente verificatore ha eseguito il caso d'uso UC-01-01-01 e scaricato le immagini presenti nell'articolo nella cartella "articleImage" 2. Il tool prende in input la cartella "articleImage" contenente le immagini dell'articolo estratte nello UC-01-01-01
Flusso Principale	<ul style="list-style-type: none"> 1. Il tool esegue il caso d'uso UC-01-01-09-01 2. Il tool esegue il caso d'uso UC-01-01-09-02
Post Condizioni	La funzione aggiorna lo score totale
Flusso Alternativo	In ogni momento l'utente può interrompere l'operazione
Post Condizioni	Nessun dato viene salvato
Flusso Alternativo	

3.2.13.1 UC-01-01-09-01

ID	UC-01-01-09-01
Nome	<u>Related_Search_Image_analysis</u>
Attori	Utente Verificatore
Precondizioni	<ol style="list-style-type: none"> 1. L'utente verificatore ha eseguito il caso d'uso UC-01-01-01 e scaricato le immagini presenti nell'articolo nella cartella "articleImage" 2. Il <u>tool</u> prende in input la cartella "articleImage" contenente le immagini dell'articolo estrapolate nello UC-01-01-01
Flusso Principale	<ol style="list-style-type: none"> 1. Il <u>tool</u> effettua l'upload delle immagini simulando una ricerca per immagine sul motore di ricerca "Google". 2. Per ogni immagine presente nella cartella "articleImage", il <u>tool</u> chiede all'utente di selezionare se la ricerca correlata di Google (estratta dall'immagine) è relativa ai fatti della notizia. 3. Se è relativa ai fatti della notizia: <ol style="list-style-type: none"> a1) il <u>tool</u> assegna il valore 1 allo score dell'immagine 4. else <ol style="list-style-type: none"> a2) il <u>tool</u> estrae i link relativi all'immagine dalla ricerca Google e chiede all'utente di selezionarne uno se relativo ai fatti della notizia: <ol style="list-style-type: none"> b1) il <u>tool</u> assegna il valore 1 allo score dell'immagine c) else <ol style="list-style-type: none"> b2) il <u>tool</u> assegna il valore 0 allo score dell'immagine
Post Condizioni	La funzione restituisce lo score per ogni singola immagine presente nella cartella "articleImage"
Flusso Alternativo	In ogni momento l'utente può interrompere l'operazione
Post Condizioni Flusso Alternativo	Nessun dato viene salvato

3.2.13.2 UC-01-01-09-02

ID	UC-01-01-09-02
Nome	<u>Score_image</u>
Attori	Utente Verificatore
Precondizioni	<ol style="list-style-type: none"> 1. L'utente verificatore ha eseguito il caso d'uso UC-01-01-09-01
Flusso Principale	<ol style="list-style-type: none"> 1. Il <u>tool</u> conta le immagini 2. Il <u>tool</u> calcola il rapporto tra la somma degli score delle singole immagini (estratte dal caso d'uso UC-01-01-09-01) e il numero delle immagini contate
Post Condizioni	La funzione aggiorna lo score totale della notizia
Flusso Alternativo	In ogni momento l'utente può interrompere l'operazione
Post Condizioni Flusso Alternativo	Nessun dato viene salvato

Capitolo 4

Tecnologie utilizzate nel progetto

In questo capitolo saranno descritte le principali tecnologie utilizzate per lo sviluppo del prototipo afferente a questo progetto.

4.1 Web Scraping

Con il termine "web scraping" ci si riferisce all'estrazione di dati da un sito web. Queste informazioni vengono raccolte e poi esportate in un formato utile ad un'analisi. Anche se il web scraping può essere fatto manualmente, si preferisce automatizzare il processo grazie a numerosi tool o programmi che risultano essere meno costosi e permettono di lavorare ad un ritmo più veloce. Il funzionamento di massima di un web scraper può essere riassunto nei seguenti punti:

1. Viene dato in input al web scraper un URL da caricare. Lo scraper carica quindi l'intero codice HTML, CSS e Javascript della pagina in questione.
2. Il web scraper estrae tutti i dati (o alcuni dati specifici selezionati dall'utente) dalla pagina.
3. Il web scraper converte tutti i dati estratti nel formato specificato dall'utente.

La maggior parte dei web scraper estrapola i dati in un foglio di calcolo CSV o Excel, mentre gli scraper più avanzati supportano anche altri formati come il JSON. Qualsiasi pagina web presente in internet può essere scansionata per ottenerne le informazioni necessarie e, allo stesso tempo, qualsiasi oggetto visibile su una pagina web può essere estratto attraverso l'uso di un web scraper. In particolare, ogni pagina web ha una sua struttura con i propri elementi caratteristici. Per questo è necessario in taluni casi tarare in via preliminare lo strumento prima di poter effettuare una estrazione efficace delle informazioni. [31]

4.1.1 Goose3

Originariamente, Goose era un web scraper scritto in Java ora riprogettato completamente in Python. Lo scopo principale del software è quello di estrarre da un

qualsiasi pagina web contenente un articolo di notizie, il corpo principale dello stesso ma anche tutti i meta dati ed eventuali immagini presenti. Goose estrae le seguenti informazioni:

- Testo principale di un articolo
- Immagine principale dell'articolo
- Qualsiasi filmato YouTube/Vimeo incorporato nell'articolo
- Meta description e tag

Per estrarre le seguenti informazioni è necessario passare in input a Goose un url. Il programma, in base al tipo di lingua che viene rilevata o che è stata precedentemente impostata, pulisce il testo in input da eventuali punteggiature, tag, ecc e restituisce il testo normalizzato. [18]

4.1.2 Scrapy

Scrapy è un potente framework di web crawling basato su Python. Scrapy utilizza dei crawler chiamati spider, che possono estrarre, elaborare e salvare i dati. Poiché Scrapy è costruito su Twisted, un framework di rete asincrono, le sue prestazioni sono estremamente veloci grazie al meccanismo non bloccante (gestione della concorrenza) per inviare le richieste. Quando si fa qualcosa in modo sincrono, bisogna aspettare prima che un processo finisca per poterne iniziare un altro; mentre quando si fa qualcosa in modo asincrono, i processi possono essere eseguiti contemporaneamente senza che uno aspetti la fine dell'altro per poter essere eseguito. [29]

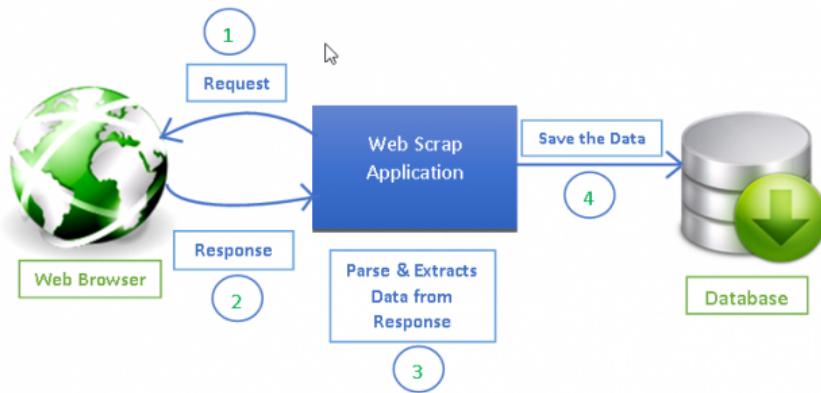


Figura 4.1. Scrapy Web Crawling

I Vantaggi nell'uso di questo framework sono molteplici [10]:

- Grazie ad meccanismo incorporato chiamato Selectors, esso individua ed estrae dati da una pagina web utilizzando XPath e CSS.
- Non ha bisogno di una codifica estesa come altri framework; basta solamente definire il sito web e i dati da estrarre.

- E' gratuito, open-source e multipiattaforma.
- È veloce, potente e facilmente estensibile grazie alla sua gestione asincrona delle richieste.
- Essendo veloce, potente e facilmente estensibile grazie alla sua gestione asincrona delle richieste, viene usato in grandi progetti per costruire e scalare crawler.
- Grazie a Scrapy è possibile scansionare qualsiasi pagina web indipendentemente dalla disponibilità di dati grezzi.
- Minore consumo di memoria e CPU rispetto ad altre librerie.
- Supporta l'esportazione dei dati in diversi formati come CSV, XML, JSON e JSON.

Per la realizzazione del tool è stato scelto Scrapy in quanto è stato più funzionale al nostro scopo. Inoltre presenta le seguenti caratteristiche [48]:

- E' estremamente funzionale
- Rappresenta il pacchetto completo per scaricare pagine web, elaborarle e salvarle in file e database.
- "Learning Curve": In quanto Scrapy non si occupa solo dell'estrazione di contenuti ma anche di molti altri compiti come il download di HTML, la curva di apprendimento di Scrapy è molto più elevata, ma se usato a dovere esso rappresenta una macchina da guerra per il web scraping.
- Scrapy può eseguire un'ingente mole di lavoro molto facilmente. Può scansionare un gruppo di URL in non più di un minuto, a seconda delle dimensioni del gruppo, e lo fa in modo molto fluido poiché utilizza Twister che funziona in modo asincrono (non bloccante) per la concorrenza. Ad esempio, BeautifulSoup è usato per semplici lavori di scraping ed è estremamente più lento di Scrapy in quanto non usa il multiprocessing.
- L'architettura di Scrapy è ben progettata ed è possibile sviluppare facilmente middleware o pipeline personalizzate per aggiungere funzionalità personalizzate.
- Grazie a Scrapy, lo spider può inviare molte richieste contemporaneamente ed è quindi necessario impostare download_delay nella maggior parte dei casi per evitare di essere bannati. Ad esempio, altri framework di web crawling come BeautifulSoup non hanno questa caratteristica. [48]

Purtroppo, nelle ultime versioni del programma, è stata rimossa la funzionalità per lo scaricamento delle immagini; per questo motivo si è scelto di integrare nel progetto anche BeautifulSoup

4.1.3 BeautifulSoup

Beautiful Soup è una libreria Python usata per estrarre dati da pagine HTML e XML. Essa usa il parser html/xml preinstallato e converte la pagina web/html/xml in un albero composto da tag, elementi, attributi e valori. Per essere più precisi, l'albero consiste di quattro tipi di oggetti: Tag, NavigableString, BeautifulSoup e Comment. Questo albero può poi essere "interrogato" utilizzando i metodi/proprietà dell'oggetto BeautifulSoup che viene creato dalla libreria parser. Per la realizzazione del nostro tool, come è stato spiegato precedentemente, si è scelto di usare questa libreria (beautifulsoup4) per il download delle immagini presenti nella pagina web.

```
soup = BeautifulSoup(data, 'html.parser') #Setup a "soup" which BeautifulSoup can search
links = []

for link in soup.find_all('img'): #Cycle through all 'img' tags
    imgSrc = link.get('src') #Extract the 'src' from those tags
    links.append(imgSrc) #Append the source to 'links'
```

Figura 4.2. Estratto di codice BeautifulSoup per immagini

Ciò che rende Beautiful Soup così utile è la miriade di funzioni che fornisce per estrarre dati dall'HTML. L'immagine in figura 4.3 illustra alcune delle funzioni che possiamo usare.

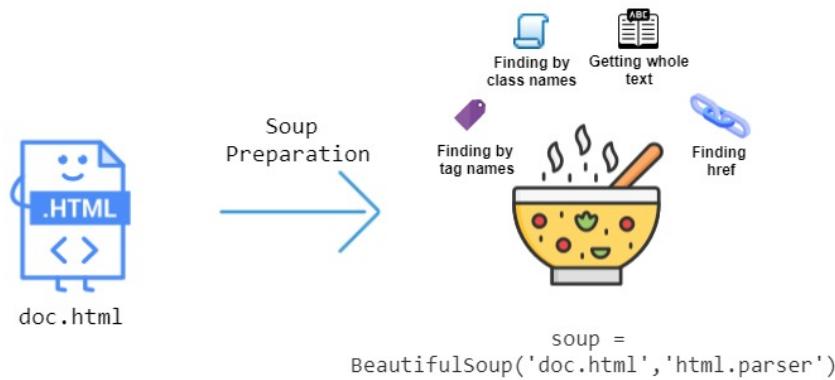


Figura 4.3. BeautifulSoup Functions

4.1.3.1 Come usare correttamente BeautifulSoup

Per usare correttamente la libreria BeautifulSoup è necessario eseguire i seguenti passi:

- Importare la libreria Beautiful Soup
- Aprire una pagina web o un testo html con la libreria BeautifulSoup, indicando quale parser deve essere usato. Il risultato di questo passo è un oggetto

BeautifulSoup. (Nota: il nome del parser menzionato deve essere già installato come parte del tuo pacchetto Python. Per esempio, `html.parser`, è un pacchetto in-built, 'with-batteries' spedito con Python. Potreste installare altri parser come `lxml` o `html5lib`.)

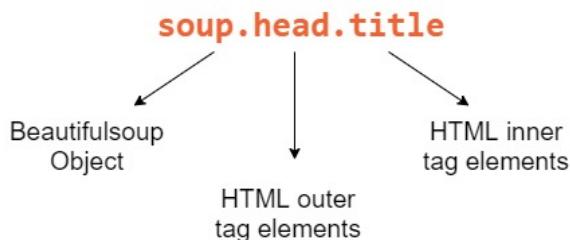


Figura 4.4. BeautifulSoup Object

- "Interrogare" o cercare l'oggetto BeautifulSoup usando la sintassi `object.method` e ottenere il risultato in una collezione, come un dizionario Python. Per alcuni metodi, l'output sarà un semplice valore.
- Usa il risultato del passo precedente per fare qualsiasi cosa tu voglia fare con esso, nel proprio codice Python. È possibile modificare i valori degli elementi o degli attributi nell'oggetto albero. Le modifiche non influenzano il sorgente del codice html, ma è possibile chiamare i metodi di formattazione dell'output (come `prettify`) per creare un nuovo output dall'oggetto BeautifulSoup. Metodi comunemente usati: Tipicamente, i metodi `.find` e `.find_all` sono usati per cercare nell'albero, dando gli argomenti di input.

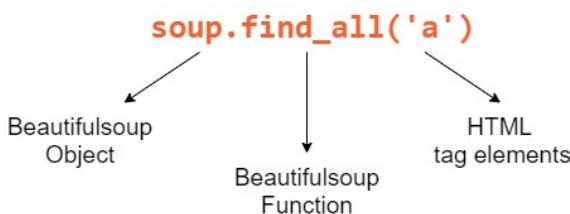


Figura 4.5. BeautifulSoup `find_all`

- Gli argomenti di input sono: il nome del tag che si sta cercando, i nomi degli attributi e altri argomenti correlati. Questi argomenti potrebbero essere presentati come: una stringa, un'espressione regolare, una lista o anche una funzione. Gli usi comuni dell'oggetto BeautifulSoup includono:
 - Ricerca per classe CSS
 - Ricerca per indirizzo di collegamento ipertestuale

- Ricerca per Id dell'elemento, tag
- Ricerca per nome dell'attributo. Valore dell'attributo.
- Se hai bisogno di filtrare l'albero con una combinazione dei criteri di cui sopra, potresti anche scrivere una funzione che valuti vero o falso, e cercare per quella funzione.

4.2 Estrazione delle località utilizzando NLP

Il **Natural Language Processing (NLP)** è una branca dell'intelligenza artificiale che aiuta i calcolatori a capire, interpretare e manipolare il linguaggio umano. NLP attinge da molte discipline, tra cui l'informatica e la linguistica computazionale, nel suo tentativo di colmare il divario tra la comunicazione umana e la comprensione del computer. L'elaborazione del linguaggio naturale include molte tecniche diverse per interpretare il linguaggio umano, che vanno dai metodi statistici e di apprendimento automatico agli approcci algoritmici basati sull'applicazione di tecniche specifiche. I compiti di base di NLP includono la tokenizzazione ed il parsing, la lemmatizzazione/stemming, il part-of-speech tagging, il rilevamento della lingua e l'identificazione delle relazioni semantiche; tutti processi che ad un essere umano sono stati insegnati già nelle scuole elementari. [13] In termini generali, il compito primario di NLP è quello di scomporre il linguaggio in pezzi più brevi ed elementari, cercando di capire le relazioni tra i pezzi ed esplorando come questi lavorano insieme per creare significato. Questi compiti di base sono spesso utilizzati in NLP per ottenere dei risultati come:

- **Categorizzazione dei contenuti.** Un riassunto di documenti basato sulla linguistica, inclusa la ricerca e l'indicizzazione, gli avvisi di contenuto e il rilevamento di eventuali duplicati.
- **Scoperta e modellazione degli argomenti.** Catturare accuratamente il significato e i temi nelle collezioni di testo, e applicare analisi avanzate al testo, come l'ottimizzazione e la previsione.
- **Estrazione contestuale.** Estrarre automaticamente informazioni strutturate da fonti basate sul testo.
- **Sentiment Analysis.** Identificare l'umore o le opinioni soggettive all'interno di grandi quantità di testo, compreso il sentimento medio e l'estrazione delle opinioni.
- **Conversione da voce a testo e da testo a voce.** Trasformazione dei comandi vocali in testo scritto e viceversa.
- **Riassunto di documenti.** Generazione automatica di sinossi di grandi corpi di testo.
- **Traduzione automatica.** Traduzione automatica di testo o discorso da una lingua all'altra.

Attraverso l’NLP, il nostro sistema è in grado, quindi, di estrarre le località presenti nel testo in input, in modo da poter calcolare la distanza tra il luogo di avvenimento dei fatti della notizia ed quello in cui il lettore sta effettuando la ricerca. Per fare ciò sono state utilizzate alcune librerie e formule matematiche specificate di seguito. [42]

4.2.1 spaCy

spaCy è una libreria gratuita e open-source per l’elaborazione avanzata del linguaggio naturale (NLP) in Python. Essa è stata progettata specificamente per aiutare a costruire applicazioni che elaborano e "capiscono" grandi volumi di testo. Può essere usata per costruire sistemi di estrazione di informazioni o di comprensione del linguaggio naturale, o per pre-elaborare il testo per il deep learning. [41] Grazie all’utilizzo di questa libreria è infatti possibile effettuare [39]:

- **Tokenizzazione.** Segmentazione del testo in parole, segni di punteggiatura ecc.
- **Part-of-speech (POS) Tagging.** Assegnazione di tipi di parole ai token, come verbo o nome.
- **Dependency Parsing.** Assegnazione di etichette di dipendenza sintattica, che descrivono le relazioni tra i singoli token, come soggetto o oggetto.
- **Lemmatizzazione.** Assegnazione delle forme base delle parole.
- **Sentence Boundary Detection (SBD)** Individuazione e segmentazione di singole frasi.
- **Named Entity Recognition (NER)** Etichettare oggetti nominati del "mondo reale", come persone, aziende o luoghi.
- **Entity Linking (EL).** Disambiguazione di entità testuali a identificatori unici in una knowledge base.
- **Similarità.** Confrontare parole, parti di testo e documenti e quanto sono simili tra loro.
- **Classificazione del testo.** Assegnazione di categorie o etichette a un intero documento o a parti di esso.
- **Corrispondenza basata su regole.** Trovare sequenze di token basate sui loro testi e annotazioni linguistiche, simili alle espressioni regolari.
- **Training.** Aggiornamento e miglioramento delle previsioni di un modello statistico.
- **Serializzazione.** Salvataggio di oggetti in file o stringhe di byte.

Per capire più in dettaglio il suo funzionamento è necessario studiare attentamente le varie fasi di processing.

4.2.1.1 Spacy Pipeline

Per processare una stringa di testo, quando si lavora con spaCy, è necessario passarla a un oggetto NLP. Questo oggetto è essenzialmente una pipeline di diverse operazioni di pre-elaborazione del testo attraverso cui la stringa di testo in ingresso deve passare. [22]

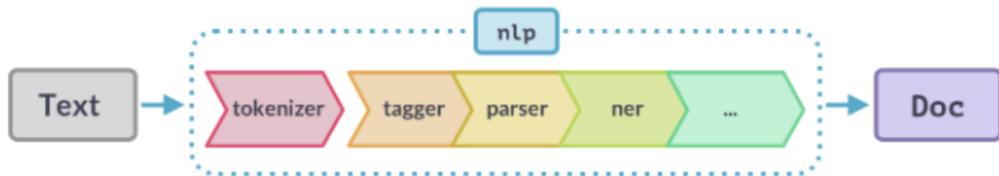


Figura 4.6. Spacy pipeline

In quanto la pipeline NLP ha più componenti, come tokenizer, tagger, parser, ner, ecc, una stringa di testo data in input deve passare attraverso tutti questi componenti prima di poter lavorare su di essa. [22] Grazie a spaCy è possibile eseguire i vari compiti di NLP di Part-of-Speech Tagging, Dependency Parsing, e Named Entity Recognition.

1. Tagging Part-of-Speech (POS) usando spaCy: Nella grammatica inglese, le parti del discorso ci dicono qual è la funzione di una parola e come viene usata in una frase. Alcune delle parti comuni del discorso in inglese sono Nome, Pronome, Aggettivo, Verbo, Avverbio, ecc. Il POS tagging è il compito di assegnare automaticamente i tag POS a tutte le parole di una frase. È utile in vari compiti a monte in NLP, come la feature engineering, la comprensione del linguaggio e l'estrazione di informazioni.
2. Parsing delle dipendenze usando spaCy: Ogni frase ha una struttura grammaticale e con l'aiuto del parsing delle dipendenze, è possibile estrarre questa struttura. Può anche essere pensato come un grafo diretto, dove i nodi corrispondono alle parole nella frase e gli archi tra i nodi sono le dipendenze corrispondenti tra le parole. [28]
3. Riconoscimento di entità nominate con spaCy. Le entità sono parole o gruppi di parole che rappresentano informazioni su cose comuni come persone, luoghi, organizzazioni, ecc. Queste entità hanno nomi propri. Ad esempio nella frase "Donald Trump will meet the chairman of Google in New York City", spaCy riconoscerà in questa frase le entità "Donald Trump", "Google" e "New York City".
4. Corrispondenza basata su regole con spaCy. La corrispondenza basata su regole è una nuova aggiunta alle funzionalità di spaCy. Con questo spaCy matcher, è possibile trovare parole e frasi nel testo utilizzando regole definite dall'utente. Mentre le Espressioni Regolari usano modelli di testo per trovare parole e frasi, il matcher di spaCy non usa solo i modelli di testo ma le proprietà lessicali della parola, come i tag POS, i tag di dipendenza, il lemma, ecc.

Dep tree	Token	Dep type
	The	det
	programmer	nsubjpass
	was	auxpass
	pleased	ROOT
	by	agent
	the	det
	nicely	advmod
	formatted	amod
	parse	compound
	tree	pobj
	.	punct

Figura 4.7. Albero delle dipendenze

4.2.2 Geopy

Longitudine e latitudine sono ottime misure per ottenere posizioni relativamente precise delle osservazioni. La trasformazione di conversione degli indirizzi in coordinate geografiche (come la latitudine e longitudine) o il processo contrario, è chiamata "**Geocodifica**". [24] Geopy consente agli sviluppatori Python di localizzare le coordinate di indirizzi, città, paesi e punti di riferimento in tutto il mondo utilizzando geocoder di terze parti e altre fonti di dati per la conversione in coordinate geografiche. Geopy, in particolare, può calcolare la distanza tra diversi punti ed effettuare chiamate ad API esterne per ottenere informazioni su queste coordinate. Dopo che è stata stabilita una connessione API impostando il geocoder, è possibile ottenere informazioni su un indirizzo usando le coordinate. [8]

```
location = geocode.reverse((lat, long))
# returns geopy Location object
```

Figura 4.8. Codice Geopy Indirizzo

Gli oggetti location hanno istanze di address, altitude, latitude, longitude, point. "address" restituisce una stringa concatenata di tutte le informazioni che appartengono all'indirizzo. Questo può includere o meno informazioni come una località, una strada, un quartiere, un isolato, un sobborgo, una contea, una città, uno stato, un codice postale, un paese, un codice di avviamento postale. Per ottenere latitudine e longitudine, è necessario usare la funzione inversa del nostro oggetto geocoder. Questo perché l'uso generale di questo geocoder è quello di ottenere le coordinate reali dall'indirizzo. Questo può essere utile se vogliamo valutare l'effetto geografico o quantificare le posizioni categoriche. [17]

Un'altra variabile importante che è possibile estrarre dalle coordinate è la distanza tra luoghi diversi. Geopy offre sia la distanza a great-circle che la distanza geodetica.

```
# after initiating geocoder
location = geocode(address)
# returns location object with longitude, latitude and altitude
instances
(location.latitude, location.longitude)
```

Figura 4.9. Ottenere Latitudine e Longitudine

[7]

```
from geopy.distance import geodesic

distance_in_miles = geodesic(coordinate1, coordinate2).miles
distance_in_km = geodesic(coordinate1, coordinate2).km
# note: coordinates must be in a tuple of (lat, long)
```

Figura 4.10. Ottenere Distanza

4.2.3 Formula di Haversine

Grazie alle librerie precedentemente citate, che hanno permesso l'estrazione delle località da un testo e la relativa geocodifica, la formula di Haversine ha permesso, data un sfera e presi due punti in essa aventi latitudine e longitudine ben definiti, di calcolare la distanza tra essi. La formula di Haversine calcola quindi la distanza più breve tra due punti su una sfera usando le loro latitudini e longitudini misurate lungo la superficie. Tale distanza può essere espressa in funzione trigonometrica come:

$$d = rhav^{-1}(h) = 2rsin^{-1}(\sqrt{h}) \quad (4.1)$$

dove d è la distanza tra due punti ed r è il raggio della terra (6.371 km), e dove,

$$hav(\theta) := haversine(\theta) = \sin^2(\theta/2) \quad (4.2)$$

In cui, per calcolare l'haversine dell'angolo centrale (che è d/r) si usa la seguente formula:

$$(d/r) = haversine(\phi_2 - \phi_1) + \cos(\phi_1)\cos(\phi_2)haversine(\lambda_2 - \lambda_1) \quad (4.3)$$

dove, ϕ_1, ϕ_2 è la latitudine dei due punti e λ_1, λ_2 è la longitudine dei due punti rispettivamente.

Da cui, infine, risolvendo in d e applicando l'inverso dell'haversine o utilizzando la funzione seno inversa, si ottiene la formula di cui sopra.

Calcolando la distanza tra due punti in una sfera, la misura risulta essere in realtà un'approssimazione, dato che la Terra è uno sferoide oblato e non una sfera perfetta.
[46]

4.3 Estrazione Temporale

4.3.1 Pytime extractor

Grazie a questa utilissima libreria presente Python, è possibile trovare ed estrarre informazioni relative a tempo/data da documenti testuali. [16] L'obiettivo principale è quello di identificare frammenti di testo che siano legati al tempo/data/periodo (data esatta, ora del giorno, giorno della settimana, mesi, stagioni, intervalli di tempo, ecc). Questa libreria è costruita su:

- joda-time: Libreria Java per le classi di data e ora
- opencsv Parser Library
- JUnit Testing Framework
- Log4j Logging Service
- Gson Json Serialization/Deserialization library

La classe DateTimeExtractor è la classe principale per usare Timeextractor. DateTimeExtractor in particolare è usato costruendo prima un'istanza di DateTimeExtractor e poi invocando il metodo extract() su di essa, utile estrarre frammenti di data/ora dal testo inserito. [27]

4.4 Selenium

Selenium WebDriver è uno degli strumenti più popolari quando si tratta di effettuare automazione Web UI.

4.4.0.1 Cos'è Selenium WebDriver?

Una pagina web è composta da diversi elementi web, come caselle di testo, caselle di controllo, pulsanti, ecc. Un test di automazione web implica l'automazione di quei compiti che devono essere eseguiti su questi elementi web. Selenium WebDriver è un popolare framework di test di automazione basato sul web che viene utilizzato principalmente per automatizzare i compiti relativi al test dell'interfaccia utente del web. Esso non interagisce direttamente con gli elementi web su una pagina. Un Selenium WebDriver specifico per il browser funge da ponte tra lo script di test e il browser web; esso è il componente principale che comunica con il browser web.

Selenium WebDriver supporta la maggior parte dei linguaggi di programmazione popolari usati da sviluppatori e tester, vale a dire Python, Java, C#, Ruby e altri. Supporta sistemi operativi popolari come Windows, Mac OS, Linux e Solaris.

Mozilla Firefox è il browser web predefinito di Selenium WebDriver, ma per la realizzazione del tool è stato scelto di usare google Chrome in quanto uno dei browser più utilizzati.

4.4.0.2 Architettura di Selenium WebDriver

Comprendere la comunicazione tra i diversi blocchi di Selenium è essenziale prima di esaminare Selenium WebDriver con Python. Le API di Selenium WebDriver sono utilizzate per comunicare tra linguaggi di programmazione e browser web.

L'architettura di Selenium WebDriver comprende i seguenti blocchi:

- Librerie client di Selenium
- Protocollo JSON Wire
- Driver del browser
- Browser



Figura 4.11. Selenium Architecture

4.4.0.3 Librerie client di Selenium

Come accennato in precedenza, gli sviluppatori possono utilizzare Selenium per eseguire test di automazione con i linguaggi di programmazione più diffusi. Le Selenium Client Libraries o Selenium Language Bindings rendono possibile questo supporto multilingue in Selenium.

4.4.0.4 Protocollo JSON Wire

JSON sta per JavaScript Object Notation. JSON Wire Protocol è usato per il trasferimento di dati tra il server e il client sul web. È un'API REST (Representational State Transfer) che facilita il trasferimento di informazioni tra il server HTTP. Ogni browser web, vale a dire - Chrome, Firefox, Internet Explorer, ecc, ha il proprio driver del browser (o server HTTP).

4.4.0.5 Driver del browser

I driver del browser sono principalmente responsabili della comunicazione con il browser web corrispondente. Ogni browser ha il suo driver per il browser, e lo stesso deve essere installato sulla macchina dove verranno eseguiti i test di automazione. Poiché la comunicazione con il browser web avviene tramite il driver del browser, la logica interna del browser non viene rivelata. Il driver del browser aggiunge il tanto necessario livello di astrazione all'interazione con il browser.

Quando il driver del browser riceve un qualsiasi comando (o richiesta), questo viene eseguito sul rispettivo browser, e la risposta dell'esecuzione viene inviata al driver web come risposta HTTP.

4.4.0.6 Browser

Selenium può essere utilizzato con i browser più popolari come Chrome, Firefox, Internet Explorer, Microsoft Edge, ecc. Il framework non può essere utilizzato per i browser il cui driver non è disponibile.

Capitolo 5

Implementazione

5.1 Ambiente di sviluppo

Per poter procedere all'implementazione del tool "Fake News Detector", di cui alla specifica dei requisiti presente nel capitolo 4, è stato necessario innanzitutto settare l'ambiente di sviluppo. Per la realizzazione del tool si è scelto di usare il linguaggio di programmazione Python alla versione 3.7

5.2 Python 3.7

Python è un linguaggio di programmazione interpretato, orientato agli oggetti, di alto livello con una semantica dinamica, creato da Guido van Rossum e rilasciato nel 1991. Le sue strutture dati, combinate con la tipizzazione ed il binding dinamico, lo rendono estremamente adatto allo sviluppo di applicazioni web. Può essere usato insieme al software per creare flussi di lavoro, può connettersi a sistemi di database e permette di leggere e modificare i file. È molto usato anche in ambito matematico a causa della presenza di librerie specializzate. [34] Grazie alla sua sintassi, Python enfatizza la leggibilità e l'usabilità del codice riducendo il costo della manutenzione del programma. Inoltre supporta moduli, pacchetti e librerie che permettono ai programmi scritti in questo linguaggio modularità e riutilizzo del codice. Essendo multi piattaforma, può essere eseguito su diverse piattaforme (Windows, Mac, Linux, Raspberry Pi, ecc.), il che gli permette di essere uno dei linguaggi di programmazione più usato nel mondo. [45]

Per la realizzazione del prototipo è stato usato Python versione 3.7 a causa della sua compatibilità con le librerie descritte precedentemente.

5.3 Spyder

È sempre necessario avere ambienti interattivi quando si creano applicazioni software e questo diventa molto importante soprattutto quando si lavora nei campi di Data Science, dell'ingegneria e della ricerca scientifica. [15] L'IDE Python Spyder è stato creato per questo scopo. Spyder è un IDE open-source multipiattaforma ed è scritto completamente in Python. È progettato da scienziati ed è esclusivamente

per scienziati, data analysts e ingegneri. È anche conosciuto come Scientific Python Development IDE e ha un enorme set di caratteristiche:

- Customizzazione della sintassi
- Disponibilità di breakpoint (debug e breakpoint condizionali)
- Esecuzione interattiva che consente di eseguire righe, file, celle, ecc.
- Configurazioni di esecuzione per selezioni di directory di lavoro, opzioni della linea di comando, console corrente/ dedicata/ esterna, ecc.
- Configurazioni di esecuzione per selezioni di directory di lavoro, opzioni della linea di comando, console corrente/ dedicata/ esterna, ecc.
- Può cancellare automaticamente le variabili (o entrare nel debug)
- Permette la navigazione attraverso celle, funzioni, blocchi, ecc. può essere realizzata attraverso l'Outline Explorer
- Fornisce l'introspezione del codice in tempo reale (la capacità di esaminare quali funzioni, parole chiave e classi sono, cosa stanno facendo e quali informazioni contengono)
- Inserimento automatico dei due punti dopo if, while, ecc.
- Supporta tutti i comandi di IPython
- Permette la visualizzazione per i grafici prodotti usando Matplotlib
- Fornisce anche caratteristiche come aiuto, esplorazione di file, funzioni di trova file, ecc.

L'IDE Python Spyder viene fornito come implementazione predefinita insieme alla distribuzione Python di Anaconda. Questo non è solo il metodo raccomandato, ma anche il più semplice. Per l'installazione di Spyder sono stati effettuati i seguenti passaggi:

1. Dal sito ufficiale (www.anaconda.com) è stata scaricata la versione per mac contenente Python 3.7
2. Una volta che il programma di installazione è stato scaricato, si è provveduto a lanciare il file per l'installazione.
3. Al termine dell'installazione si è provveduto a lanciare Anaconda-navigator e successivamente Spyder

The screenshot shows the Spyder Python IDE interface. On the left, the code editor displays the `main.py` file with approximately 80 lines of Python code. The code imports various modules like `openpyxl`, `Who`, `WNER`, `WAT`, `PyPDF2`, `searchgoole`, `Scrapy`, `lxml`, `pd`, `openpyxl`, `pathlib`, `langdetect`, and `deep_translator`. It reads an Excel sheet, iterates through rows, and performs operations such as URL validation, WHO analysis, and language detection. The right side of the interface includes a 'Usage' help box, a 'Console' window showing the Python environment, and a status bar at the bottom.

```

1  #!/usr/bin/python3
2  # -*- coding: utf-8 -*-
3
4  from who import WHO
5  from wner import WNER
6  from wnat import WAT
7  from PyPDF2 import PdfFileReader
8  from searchgoole import searchGO
9  from scrapy import Spider, Item, Field
10 from scrapy_splash import SplashRequest
11 from checker import validate_url
12 from checker import Ingestable
13 from checker import validate_domain
14 import openpyxl
15 import re
16 from pathlib import Path
17 from langdetect import detect
18 from deep_translator import GoogleTranslator
19
20
21 df_test = pd.read_excel('link_site.xlsx')
22 df_test.head()
23
24 book = openpyxl.load_workbook('link_site.xlsx')
25 sheet = book.active
26
27 for i in range(1, len(sheet['A'])):
28     for j in range(1, 3):
29         cell = sheet.cell(row=i, column=j)
30         url = cell.value
31         url3 = validate_url(url)
32         if url3:
33             score_who_tot += score_who(url)
34             score_ner_tot += score_ner(url)
35             score_wat_tot += score_wat(url)
36             score_pdf_tot += score_pdf(url)
37             print("Please enter the url to analyze: ")
38             url = validate_url(input())
39             score_tot += score_tot(score_who(url))
40             score_tot += score_tot(score_ner(url))
41             score_tot += score_tot(score_wat(url))
42             score_tot += score_tot(score_pdf(url))
43
44             text, auth, scoreWho, title, image, summary = WHO(url)
45             print(auth)
46             print(text)
47             print(scoreWho)
48             score_who_tot+=scoreWho
49
50             lang = detect(text)
51             if lang == 'en':
52                 text = text
53             else:
54                 to_translate = text
55                 to_translate = GoogleTranslator(source='auto', target='en').translate(to_translate)
56                 text = translated
57
58             score_diff = score_who_min(text)
59             score_who_tot+=scoreWho
60             score_ner_min()
61             score_ner_tot+=scoreWho
62             score_wat_min()
63             score_wat_tot+=scoreWho
64             score_pdf_min()
65             score_pdf_tot+=scoreWho
66
67             distance, score_ner_min(text)
68             print("The distance between your location and the place where the events took place is: ", distance, "KM")
69             score_tot+=scoreWho
70
71             score_myWho()
72             score_myWho+=scoreWho
73             score_who_tot+=scoreWho
74             print(title)
75             print()
76             rep_scoreCount=COUNT_UPPER(title)
77             score_ner_tot+=scoreCount
78             print(scoreCount)
79             print(score_ner_tot)
80
81             search, scoreSearch=titleSearch(title)
82             score_who_tot+=scoreSearch
83             score_ner_tot+=scoreSearch
84             scoreImage = ImageSimilarity()
85             scoreImage = scoreImage()
86             scoreImage_tot+=scoreImage
87             print(scoreImage_tot)
88
89
90

```

Figura 5.1. IDE Python Spyder

5.4 Main.py

Questo è il file principale di tutto il programma che eredita le varie funzioni del tool. Come è possibile vedere dalla figura 5.2 una volta che viene chiamato il `main.py` vengono eseguite in cascata i vari moduli che portano all'implementazione dei vari casi d'uso. Questo file permette anche il calcolo dello score totale delle varie notizie ed il valore probabilistico della veridicità, associato a queste.

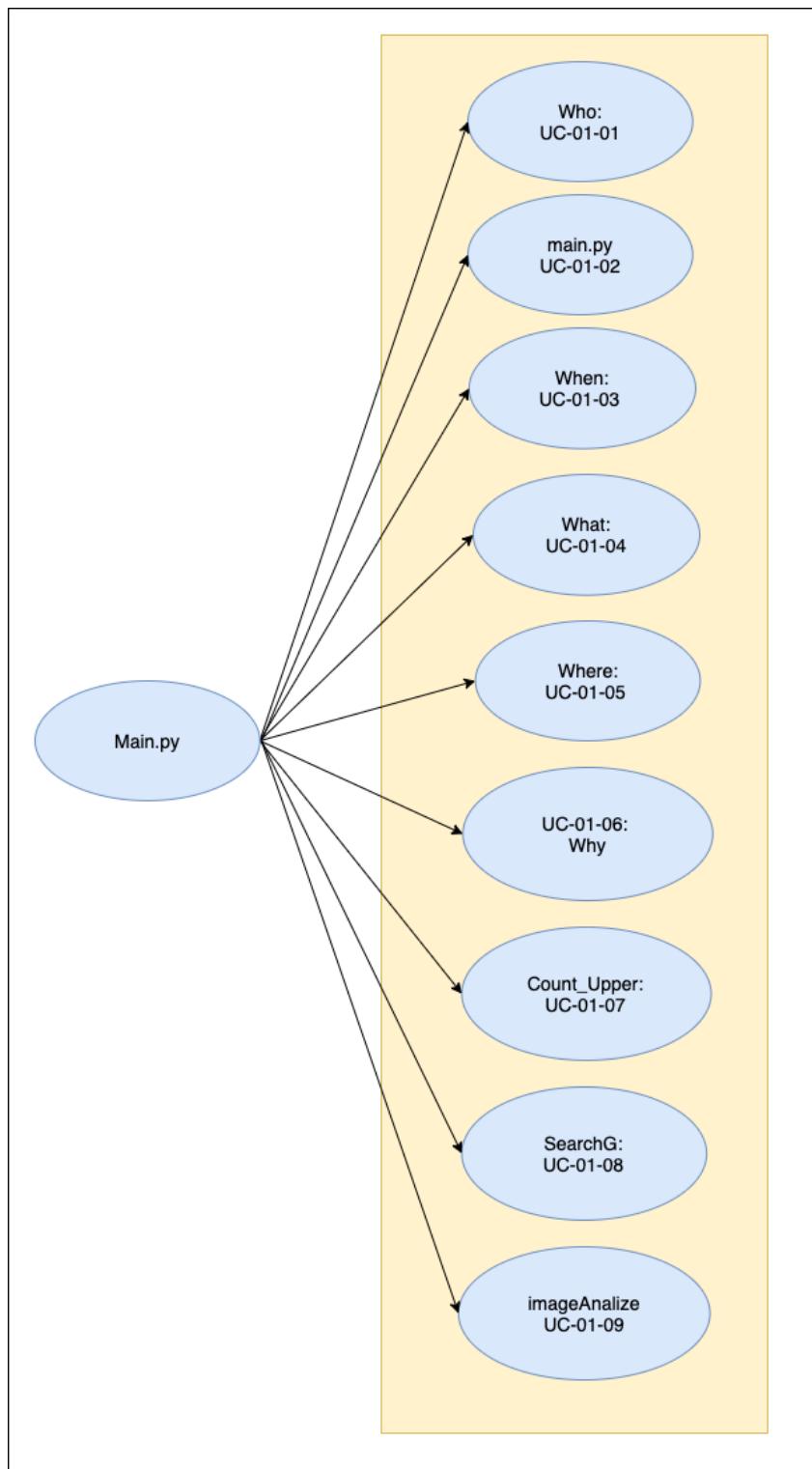


Figura 5.2. Struttura ad albero del *main.py*

5.5 Who - UC-01-01-01

L'obiettivo di questa funzione è quello di estrarre autore, titolo, testo ed immagini presenti in una notizia. A seguito dell'inserimento di un url da parte dell'utente, questo viene passato ad una funzione di nome checkURL che ne controlla il formato e notifica all'utente verificatore eventuali problemi. Dopo questa verifica, l'URL viene poi passato ad uno scraper di nome "Goose", il quale, attraverso l'analisi del codice html e xml, permette con le funzioni "*cleaned_text*", "*title*", l'estrazione del testo e del titolo in formato string.

```
def WHO(url):
    g = Goose()
    article = g.extract(url=url)
    text=article.cleaned_text
    title=(article.title)
```

Figura 5.3. Who - Impostazione dello scraper Goose

A seguito dell'estrazione del testo e del titolo si è provveduto allo scaricamento di tutte le immagini presenti nella pagina dell'url definito. Per fare ciò è stato integrato lo scraper web *BeautifulSoup* per lo scaricamento delle immagini ed il modulo "*os*" di Python per la gestione delle directory. Questa parte del codice verifica se è presente una cartella denominata "articleImage" nel path del programma e se questa è vuota. Se la cartella non esiste viene creata; in caso contrario la cartella viene svuotata dei file, eliminata, e poi creata nuovamente. Una volta creata la cartella, vengono scaricate al suo interno le immagini presenti nei tag "img" del codice html relativo all'url passato in input.

```
'Scaricamento immagini'
soup = BeautifulSoup(data, 'html.parser')
links = []

for link in soup.find_all('img'):
    imgSrc = link.get('src')
    links.append(imgSrc)

if not os.path.isdir("articleImage/"):
    os.mkdir("articleImage/")
else:
    dir_path=Path("articleImage/")
    shutil.rmtree(dir_path)
    os.mkdir("articleImage/")

countImage=1
for el in links:
    if el.startswith("http") == True:
        response = requests.get(el)
        fi=open("articleImage/image"+str(countImage)+".jpg", "wb")
        fi.write(response.content)
        fi.close()
        countImage+=1
    else:
        continue
```

Figura 5.4. Who - Scaricamento Immagini

La funzione termina con l'estrazione dell'autore ed assegna lo score relativo nel seguente modo (come definito nella specifica dei requisiti nel paragrafo 3):

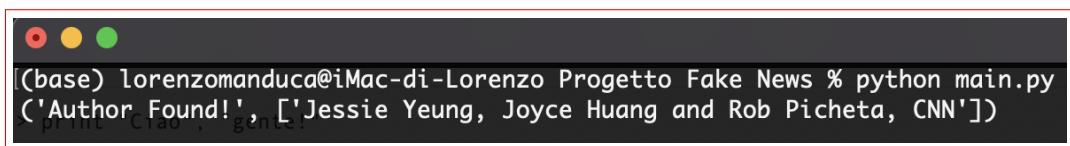
- Assegna il valore 1 se viene rilevato un autore
- Assegna il valore 0 altrimenti

```

if (article.authors == []):
    auth="Unrecognised author"
    score=0
else:
    auth=("Author Found!", article.authors)
    score=1

```

Figura 5.5. Who - Codice



```
(base) lorenzomanduca@iMac-di-Lorenzo Progetto Fake News % python main.py
('Author Found!', ['Jessie Yeung, Joyce Huang and Rob Picheta, CNN'])
```

Figura 5.6. Who - Rilevamento autore

5.6 Translater - UC-01-01-02

L’obiettivo di questa funzione è quello di tradurre in lingua inglese (se non lo è) il testo dell’URL estratto nel caso d’uso UC-01-01-01. La funzione effettua, in una prima fase, il riconoscimento della lingua del testo passato in input. Nel caso in cui il testo riconosciuto presenti una lingua diversa dall’inglese, questo permette, grazie alla libreria di google translate, la traduzione.

E’ necessario effettuare la traduzione in inglese in quanto il caso d’uso UC-01-03 (When), effettuando un nlp sul testo, riconosce solamente periodi temporali scritti in lingua inglese.

```

lang = detect(text)
if lang == 'en':
    text = text
else:
    to_translate = text
    translated = GoogleTranslator(source='auto', target='en').translate(to_translate)
    text = translated

```

Figura 5.7. Translate - Traduttore testo estratto

5.7 When - UC-01-01-03

L’obiettivo di questa funzione è quello di estrarre date ed eventuali periodi temporali dal testo estratto nel caso d’uso UC-01-01-01. Come prima cosa, la funzione elimina i caratteri speciali presenti nel testo e successivamente estraе i periodi temporali dal testo attraverso un classificatore NLP basandosi sulle espressione temporali contenute nel dataset della libreria JodaTime e di un dataset presente nella cartella "data" della libreria stessa contenente le principali espressioni temporali prese da

testi in lingua inglese e divisi in categorie richiamabili attraverso settings. Questa libreria è costruita su:

- joda-time: Libreria per le classi di data e ora di Java
- opencsv: Libreria parser
- JUnit: testing framework
- Log4j: Servizio di logging
- Gson: Libreria Json per la Serializzazione/Deserializzazione

L'algoritmo sfrutta un dataset di espressioni temporali note (incluso nella libreria utilizzata) ed espressioni regolari formattate ad-hoc per il riconoscimento di date. Poichè l'estrazione dei periodi temporali include anche espressioni come "By 20 minutes", "From 12 to 20", ecc. si sono dovuti impostare dei settings (precedentemente impostati) che escludessero questi periodi temporali per l'estrazione delle sole espressioni riportanti date. Questi settings, che hanno permesso l'esclusione di questi periodi temporali, sono:

- .excludeRules("durationRule"): esclude periodi di tempo come "last two days", "last 30 minutes", ecc
- .excludeRules("repeatedRule"): esclude i periodi temporali che presentano espressioni di eventi ripetuti come "every Sunday at 5 pm", "weekly", ecc.
- .excludeRules("timeIntervalRule"): esclude i periodi temporali come "from 19 till 20" e "2:00 pm and 4:00 pm"

A seguito dell'estrazione, grazie al modulo Pandas, le date estratte vengono inserite in un dataframe.

```
df_date=pd.DataFrame(columns = ['TemporalExpression', 'day', 'month', 'year'])
df_date.head()

df_date1=pd.concat([pd.DataFrame([elem['temporalExpression']], columns=['TemporalExpression']) for elem in result], ignore_index=True)
df_date2=pd.concat([pd.DataFrame([(elem['temporal'][0])['endDate']]['date'][1], columns = ['day']) for elem in result], ignore_index=True)
df_date3=pd.concat([pd.DataFrame([(elem['temporal'][0])['endDate']]['date'][1]['month'], columns = ['month']) for elem in result], ignore_index=True)
df_date4=pd.concat([pd.DataFrame([(elem['temporal'][0])['endDate']]['date'][1]['year'], columns = ['year']) for elem in result], ignore_index=True)
df_date=df_date1.join(df_date2['day'])
df_date=df_date.join(df_date3['month'])
df_date=df_date.join(df_date4['year'])
```

Figura 5.8. When - Creazione DataFrame

Le date inserite nel dataframe vengono passate in output all'utente attraverso il modulo inquirer. Se all'interno del testo sono presenti più di una data viene chiesto all'utente di selezionare la data più corretta corrispondente ai fatti dell'articolo; se non viene rilevata nessuna data, viene invece chiesto all'utente di selezionare la voce "*No time Identified*"

Successivamente alla scelta dell'utente, la funzione termina con l'assegnazione dello score relativo al When nel seguente modo (come definito nella specifica dei requisiti nel paragrafo 3):

- Se non è stata scelta nessuna data, la funzione assegna il valore 0 allo score

```
"Scegliere un tempo per evitare disambiguazione"

questions = [
    inquirer.List('tempo',
        message="Which of these is the time of the events in the article?",
        choices=df_date['fullDate'].tolist(),
    ),
]
answers = inquirer.prompt(questions)

giornoDataEstratta=int(df_date.loc[df_date['fullDate'] == answers["tempo"]], 'day'))
mesaDataEstratta=int(df_date.loc[df_date['fullDate'] == answers["tempo"]], 'month'))
annoDataEstratta=int(df_date.loc[df_date['fullDate'] == answers["tempo"]], 'year'))
```

Figura 5.9. When - Scelta dell'utente

- Se la differenza tra la data odierna e quella scelta è minore di 180 giorni, la funzione assegna valore 0 allo score
- Se la differenza tra la data odierna e quella scelta è compresa tra 180 giorni e 730 giorni, la funzione assegna valore 0,15 allo score
- In caso contrario, la funzione assegna valore 0,35 allo score

[?] Which of these is the time of the events in the article?: 2-4-2021
> 2-4-2021
4-4-2021
2-4-2018
No time identified

Figura 5.10. When - Rilevamento delle date

5.8 What - UC-01-01-04

L'obiettivo di questa funzione è quello di permettere all'utente di selezionare l'argomento più adatto per l'articolo da lui sottomesso al tool nello UC-01-01-01. Grazie al modulo Pandas, viene creato un dataframe contenente gli argomenti della notizia. Attraverso il modulo inquirer, viene chiesto all'utente di selezionare l'argomento più appropriato per i fatti contenuti nell'articolo.

In seguito alla scelta dell'utente, la funzione termina con l'assegnazione dello score relativo al What nel seguente modo (come definito nella specifica dei requisiti nel paragrafo 3):

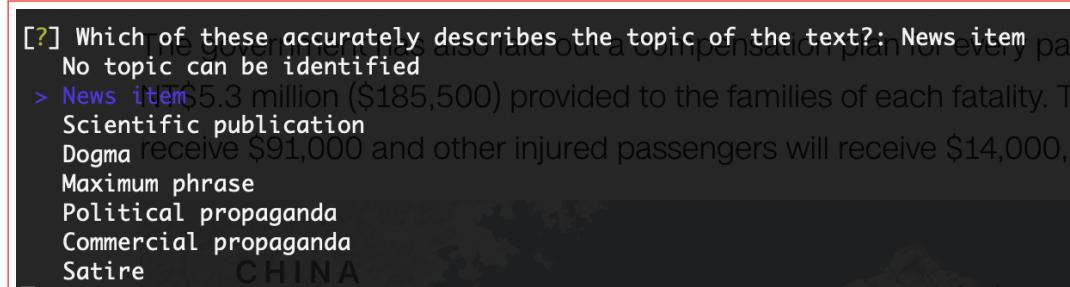
- Se non è possibile individuare un argomento per la notizia "No topic can be identified", la funzione assegna il valore 0 allo score
- Se l'argomento della notizia è "News Item" o "Scientific Publication", la funzione assegna valore 1 allo score
- Se l'argomento della notizia fa parte dei restanti argomenti selezionabili ("Dogma", "Maximum phrase", "Political propaganda", "Satire") la funzione assegna valore 0 allo score

```
"Creazione del dataframe con gli argomenti"
def WHAT():
    argomenti = {'Topic': ['No topic can be identified', 'News item',
                           'Scientific publication', 'Dogma', 'Maximum phrase',
                           'Political propaganda', 'Commercial propaganda',
                           'Satire']}
    df_Argomenti = pd.DataFrame(data=argomenti)
    df_Argomenti.head()

    "Scegliere un argomento per il What"

    import inquirer
    questions = [
        inquirer.List('topic',
                      message="Which of these accurately describes the topic of the text?",
                      choices=df_Argomenti['Topic'].tolist(),
                      ),
    ]
    answers = inquirer.prompt(questions)

    if (answers["topic"] == 'No topic can be identified'):
        score=0
    elif (answers["topic"] == 'News item' or answers["topic"] == 'Scientific publication'):
        score=1
    else:
        score=0.5
    return(score)
```

Figura 5.11. What - Codice**Figura 5.12.** What - Scelta Argomento

5.9 Where - UC-01-01-05

L’obiettivo di questa funzione è quello di estrarre località o luoghi di interesse dal testo estratto nel caso d’uso UC-01-01; questo viene fatto grazie al modulo "Spacy", il quale permette di effettuare NLP sul testo per estrarre luoghi o località. SpaCy estrae le caratteristiche di localizzazione da documenti di testo usando il Named Entity Recognition (NER).

5.9.1 Riconoscimento di entità nominate (NER)

Il riconoscimento di entità nominate (NER) è un subtask dell’estrazione di informazioni che cerca di localizzare e classificare le entità nominate menzionate in un testo non strutturato in categorie predefinite come nomi di persone, organizzazioni, luoghi, codici medici, espressioni temporali, quantità, valori monetari, percentuali, ecc. Il Named Entity Recognition può essere realizzato con diversi metodi e una semplice espressione regolare sarebbe una buona opzione. Tuttavia, alcune altre tecniche utilizzano modelli statistici o reti neurali per estrarre le entità. Spacy è una delle librerie Python più utilizzate per l’elaborazione del linguaggio naturale. Con il

Named Entity Recognition di Spacy, è possibile estrarre due tipi di caratteristiche di localizzazione: Entità geopolitiche (GPE) e località non GPE.

La libreria applica quindi un classificatore basato su un dizionario di località note incluso nella libreria. Anche in questa funzione le località estratte vengono inserite in un dataframe assieme alle loro coordinate. Per estrarre le coordinate si utilizza il modulo "geopy" che si occupa di geocodificare indirizzi, città, luoghi o località estratte in coordinate geografiche.

```
locator = geopy.geocoders.Nominatim(user_agent="mygeocoder")
geocode = RateLimiter(locator.geocode, min_delay_seconds=1)
df["address"] = df["Location"].apply(geocode)
df[['coordinates']] = df['address'].apply(lambda loc: tuple(loc.point) if loc else None)
df[['latitude', 'longitude', 'altitude']] = pd.DataFrame(df['coordinates'].tolist(), index=df.index)
df.latitude.isnull().sum()
df = df[pd.notnull(df['latitude'])]
df=df.drop_duplicates(subset=['Location'], keep='first', inplace=False)
df=df.append(df2)
```

Figura 5.13. Where - Geocodifica ed inserimento nel Dataframe

Viene successivamente chiesto all'utente, grazie al modulo "inquirer", di selezionare il luogo più appropriato nel quale si sono svolti i fatti dell'articolo tra quelli estratti dalla funzione; se non è possibile assegnare un luogo di avvenimento dei fatti della notizia l'utente seleziona la voce "No Location Identified". Oltre all'input di scelta, selezionato sopra, viene chiesto all'utente di inserire l'indirizzo o il luogo dal quale intende verificare i fatti della notizia e geocodificato dal tool. Grazie alla funzione di Haversine (Vedi paragrafo 4.2.3), viene calcolata la distanza tra il luogo dove sono avvenuti i fatti della notizia ed il luogo dell'utente verificatore; il calcolo viene effettuato grazie alla libreria "math" di python.

```
lat1=latitudineIndirizzo
lon1=longitudineIndirizzo
lat2=latitudineLocalita
lon2=longitudineLocalita
R = 6371
dLat = (lat2-lat1)*(Math.pi/180)
dLon = (lon2-lon1)*(Math.pi/180)
a = Math.sin(dLat/2) * Math.sin(dLat/2) + Math.cos(lat1*(Math.pi/180)) * Math.cos(lat2*(Math.pi/180)) * Math.sin(dLon/2) * Math.sin(dLon/2)
c = 2 * Math.atan2(Math.sqrt(a), Math.sqrt(1-a))
d = R * c
```

Figura 5.14. Where - Formula di Haversine

La funzione termina con l'assegnazione dello score relativo al Where nel seguente modo (come definito nella specifica dei requisiti nel paragrafo 3):

- Se non è possibile individuare un luogo per i fatti della notizia , la funzione assegna il valore 0 allo score
- Se la distanza tra il luogo dei fatti della notizia e il luogo dell'utente verificatore è minore di 1650 Km, la funzione assegna valore 0 allo score
- Se la distanza tra il luogo dei fatti della notizia e il luogo dell'utente verificatore è compresa tra 1650 Km e 6000 Km, la funzione assegna valore 0,15 allo score
- Altrimenti la funzione assegna valore 0,35 allo score della notizia

```
[?] Which of the following is the location of the events in the article?: Taiwan
Hualien County
Macau
Hulien
> Taiwan
Qingshui Tunnel
Hualien
Taroko National Park
No Location identified
{'localita': 'Taiwan'}
Please enter your location: Via Salaria 113, Rome
The distance between your location and the place where the events took place is: 9651.55947756042 KM
```

Figura 5.15. Where - Scelta del luogo estratto ed inserimento indirizzo

5.10 Why - UC-01-06

Questa funzione ha come obiettivo quello di permettere, grazie al modulo "inquirer", di selezionare una risposta alla domanda mostrata all'utente verificatore: *"Does the article would like to convince you to buy some product or to change your opinion about some fact/person?"* In base al tipo di risposta selezionato, la funzione termina con l'assegnazione dello score relativo al why nel seguente modo (come definito nella specifica dei requisiti nel paragrafo 3):

- Se la risposta è "Yes", la funzione assegna il valore 0 allo score
 - Se la risposta è "No", la funzione assegna il valore 1 allo score

[?] Does the article would like to convince you to buy some product or to change your opinion about some fact/person?: No
Yes
> No * Math.atan2(Math.sqrt(a), Math.sqrt(1-a))

Figura 5.16. Why - Domanda

5.11 Count Upper - UC-01-01-07

L'obiettivo di questa funzione è quello di calcolare il rapporto tra il numero di lettere maiuscole ed il numero di lettere totali presenti nel titolo estratto e normalizzato nel caso d'uso UC-01-01-01.

La funzione conta le lettere totali (sumletter) e le lettere maiuscole (cup) e ne calcola il rapporto. La funzione termina con l'assegnazione dello score relativo al count_upper nel seguente modo (come definito nella specifica dei requisiti nel paragrafo 3):

- Se il rapporto è minore del 10%, la funzione assegna il valore 1 allo score
 - Se il rapporto è maggiore del 10%, la funzione assegna il valore 0 allo score

```
def COUNT_UPPER(text):
    if text != '':
        c_up = sum(1 for c in text if c.isupper())
        sum_letter = sum(1 for c in text)
        rep = c_up / sum_letter
        if (rep < (10 * sum_letter) / 100):
            score = 1
        else:
            score = 0
    else:
        score = 0
        rep = 0

    return(rep, score)
```

Figura 5.17. Count Upper - Codice

```
Passenger train carrying 490 derails in Taiwan, killing at least 50 and injuring dozens
1 distance 1650.
```

Figura 5.18. Count Upper - Score

5.12 SearchG - UC-01-01-08

L'obiettivo di questa funzione è quello di simulare una ricerca sul motore di ricerca Google, tramite il titolo estratto dal caso d'uso UC-01-01-01, per constatare se ci siano o meno risultati affini alla notizia data in input precedentemente. Per fare ciò viene utilizzato il modulo "googlesearch" al quale viene passato in input il title; viene poi utilizzato lo scraper web "BeautifulSoup" per estrarre i risultati (titolo e link) dalla pagina di ricerca.

Grazie al modulo Pandas viene creato un dataset contenente i link ed i titoli precedentemente estratti dalla funzione. Anche in questo caso viene chiesto all'utente di selezionare il titolo ed il link più affine alla notizia da lui sottomessa nel caso d'uso UC-01-01; nel caso in cui non venisse trovata nessuna notizia o non sia affine a quella sottomessa dall'utente verificatore, può selezionare la voce "No Link identified", "No Title Identified".

La funzione termina con l'assegnazione dello score relativo al searchG nel seguente modo (come definito nella specifica dei requisiti nel paragrafo 3):

- Se l'utente seleziona un link/titolo, la funzione assegna il valore 1 allo score
- Altrimenti, la funzione assegna il valore 0 allo score

```
def searchG(textN):
# to search
    query = textN

    j=search(query, num_results=5)

    res_link={}
    link_tot=[]
    title_tot=[]
    for elem in j:
        reqs = requests.get(elem)
        # using the BeautifulSoup module
        soup = BeautifulSoup(reqs.text, 'html.parser')
        # displaying the title
        for title in soup.find_all('title'):
            titolo=(title.get_text())
```

Figura 5.19. SearchG - Codice

```
[] Which of these is closest to the news? https://edition.cnn.com/2021/04/01/asia/taiwan-train-derail-intl-hnk/index.html || Open Menu
https://edition.cnn.com/2021/04/01/asia/taiwan-train-derail-intl-hnk/index.html || No Title Identified
https://www.bbc.com/news/world-asia-56612248 || Taiwan: Dozens killed as train crashes and derails in tunnel - BBC News
https://www.gwinnettdailypost.com/news/world_nation/passenger-train-carrying-490-derails-in-taiwan-killing-at-least-50-and-injuring-dozens/image_613ee76f-8b0e-5d40-9a5e-ac5394573ba1.html
|| Passenger train carrying 490 derails in Taiwan, killing at least 50 and injuring dozens | World/Nation | gwinnettdailypost.com
https://www.aljazeera.com/news/2021/4/2/taiwan-train-de-rails-many-feared || facebook
https://www.albanyherald.com/news/world_nation/passenger-train-carrying-490-derails-in-taiwan-killing-at-least-50-and-injuring-dozens/image_428755aa-6e11-53f0-b692-8427fb964a81.html || Pa
ssenger train carrying 490 derails in Taiwan, killing at least 50 and injuring dozens | World & Nation | albanyherald.com
No Link identified || No Title Identified
```

Figura 5.20. SearchGoogle

5.13 ImageAnalize - UC-01-01-09

L'obiettivo di questa funzione è quello di simulare una ricerca per immagini sul motore di ricerca Google per verificare se le eventuali immagini presenti nell'articolo o nella notizia sono pertinenti o meno ai fatti contenuti in essa. Le immagini prese in input sono quelle scaricate nella cartella "articleImage" dallo UC-01-01. Come prima cosa viene controllato se nella cartella "articleImage" sono presenti o meno delle immagini; questo viene fatto grazie al modulo "os" di Python. Successivamente, viene fatto l'encode delle immagini e viene generato l'url per effettuare la ricerca. Grazie alla libreria "Selenium", la quale istanzia un browser web "Chrome" in background, viene passato l'url del punto precedente e vengono estratte le informazioni relative alla ricerca per url delle immagini contenute nell'articolo.

Anche in questa funzione viene usato il modulo "Pandas" per creare un dataset contenente i link ed i titoli relativi alla ricerca per immagine. Per ogni immagine vengono estratte le informazioni relative alla ricerca correlata di Google tramite il tag "fKDtNb". Questo perché Google utilizza dei suoi algoritmi di intelligenza artificiale che cercano di riconoscere il contenuto delle immagini e di associargli alcune keyword per ricerche sul loro contenuto. Se vengono trovati risultati, viene chiesto all'utente di indicare o meno se l'argomento dell'articolo o della notizia si riferisce al valore della ricerca correlata estratto dall'immagine. In caso di risposta

```

path_1 = "articleImage/"
if len(os.listdir(path_1)) != 0:

    onlyfiles = [i for i in listdir(path_1) if isfile(join(path_1, i))]
    print(onlyfiles)
    contScore=0
    scorePar=0
    for elem in onlyfiles:

        filePath = path_1 + elem
        searchUrl = 'http://www.google.com/searchbyimage/upload'
        multipart = {'encoded_image': (filePath, open(filePath, 'rb')), 'image_content': ''}
        response = requests.post(searchUrl, files=multipart, allow_redirects=False)
        fetchUrl = response.headers['Location']

```

Figura 5.21. ImageAnalize - Codice

```

options = Options()
options.add_argument('--headless')
options.add_argument('--disable-gpu') # Last I checked this was necessary.
driver = webdriver.Chrome(ChromeDriverManager().install(), chrome_options=options)

#Specify Search URL
search_url=fetchUrl
driver.get(search_url.format())

df = pd.DataFrame(columns = ['Link'])
df2 = pd.DataFrame([['No Link identified']], columns=['Link'])

ricResult=driver.find_elements_by_xpath("//a[contains(@class,'fKDtNb')]")
totalricResults=len(ricResult)

```

Figura 5.22. ImageAnalize - Selenium

affermativa la funzione aggiorna lo score parziale delle immagini sommando il valore 1 ed aggiunge il valore 1 anche al conteggio delle immagini. Inoltre, in caso di risposta negativa, la funzione estrae i link relativi all'immagine dalla ricerca e chiede all'utente di selezionarne uno se relativo ai fatti della notizia; se non c'è nessun link relativo, l'utente selezionerà "No Link identified". Se viene scelto un link, la funzione aggiorna lo score parziale delle immagini sommando il valore 1 ed aggiunge il valore 1 anche al conteggio delle immagini; se non viene scelto un link la funzione aggiorna lo score parziale delle immagini sommando il valore 0 ed aggiunge il valore 1 al conteggio delle immagini. Questo processo viene ripetuto per ogni immagine presente nella cartella "articleImage". La funzione termina con l'assegnazione dello score relativo ad imageAnalize nel seguente modo (come definito nella specifica dei requisiti nel paragrafo 3):

- Se nella cartella "articleImage" non sono presenti immagini, la funzione assegna valore 1 allo score
- Altrimenti lo score totale delle immagini viene calcolato dalla formula

$$\text{scoreImmagini} = \left(\frac{\sum_{i=1}^n \text{scorePar}(i)}{n} \right)$$
 dove
 - n = numero totale delle immagini
 - $\text{scorePar}(i)$ = score assegnato all'immagine i

5.13.1 La codifica URL di Google per la ricerca delle immagini

Per poter automatizzare la ricerca inversa delle immagini (Reverse image search) tramite Google con Python è stato necessario utilizzare il modulo "Requests". La ricerca inversa delle immagini consiste in una richiesta POST con un corpo "multipart" ed un URL di caricamento, la cui risposta è un reindirizzamento alla pagina dei risultati effettivi. Il processo di caricamento di un'immagine per la ricerca consiste in 2 fasi. La prima fase carica l'immagine tramite l'url "<http://images.google.com/searchbyimage/upload>" ed il server di Google restituisce la relativa fingerprint. Nella seconda fase il browser viene reindirizzato ad una pagina di ricerca con una query string basata sulla fingerprint restituita nella prima fase. A meno che Google non pubbli l'algoritmo per generare questa fingerprint, non è possibile generare la stringa di ricerca dall'interno della nostra applicazione. Fino ad allora, si può fare in modo che il tool invii l'immagine all'URL di caricamento. Grazie a ciò si è potuto essere in grado di analizzare la risposta e costruire la stringa di ricerca.

```
[WDM] - ===== WebDriver manager =====
[WDM] - Current google-chrome version is 89.0.4389
[WDM] - Get LATEST driver version for 89.0.4389
[WDM] - Driver [/Users/lorenzomaneduca/.wdm/drivers/chromedriver/mac64/89.0.4389.23/chromedriver] found in cache
[?] Is the topic of the article referring to? Elton John?: Yes
> Yes
No
```

Figura 5.23. ImageAnalize - Ricerca

```
[WDM] - ===== WebDriver manager =====
[WDM] - Current google-chrome version is 89.0.4389
[WDM] - Get LATEST driver version for 89.0.4389
[WDM] - Driver [/Users/lorenzomaneduca/.wdm/drivers/chromedriver/mac64/89.0.4389.23/chromedriver] found in cache
[?] Is the topic of the article referring to? formal wear?: No
> Yes
> No

[?] Which of these is closest to the image?: https://www.theguardian.com/music/2021/feb/07/elton-john-brexit-negotiators-screwed-up-deal-for-british-musicians
https://en.wikipedia.org/wiki/Formal\_wear
https://www.cheggall.com/html/formal-wear.html
https://www.theguardian.com/music/2021/feb/07/elton-john-brexit-negotiators-screwed-up-deal-for-british-musicians
https://www.fr24news.com/a/2021/02/elton-john-brexit-cooper-lined-up-to-play-harry-gibb-in-bee-gees-bionic-music/2021/feb/07/elton-john-brexit-negotiators-screwed-up-deal-for-british-musicians
https://teststar.in/1c009a-bradley-cooper-lined-up-to-play-harry-gibb-in-bee-gees-bionic-music/2021/feb/07/elton-john-brexit-negotiators-screwed-up-deal-with-a-british-musician-music/275838/
https://emintra.co.uk/elton-john-britains-brexit-negotiator-messed-up-a-deal-with-a-british-musician-music/275838/
https://shepherdgazette.com/elton-john-brexit-negotiators-screwed-up-deal-for-british-musicians-elton-john/
```

Figura 5.24. ImageAnalize - Ricerca

Capitolo 6

Test e conclusioni

6.1 Obiettivi e Struttura dei test

La sessione di test qui descritta ha l'obiettivo di verificare l'accuratezza dell'algoritmo proposto dal tool implementato. Per fare ciò, si è provveduto a costruire uno script che prende in input un dataset contenente una serie di URL etichettati con label che indicano se ci si riferisce ad una notizia vera o fake e, a seguito dell'applicazione del tool, verifica se queste combaciano con lo score totale ottenuto. I risultati vengono poi riassunti in una tabella in modo da poterli analizzare punto per punto.

6.2 Lo Script creato per effettuare i test

Lo script si avvale della libreria "Openpyxl" che permette di aprire/leggere/modificare i contenuti di un file Excel. Il file dato in input a questo script deve avere la seguente forma:

https://www.theguardian.com/world/2021/mar/30/new-covid-vaccines-needed-within-year-say-scientists	1
https://www.fonteverificata.it/2021/02/21/immune-al-covid-la-cura-definitiva-per-il-virus-potrebbe-nascondersi-nel-dna-di-un-romano-di-33-anni/	0

Figura 6.1. Link Site - Screen

Come si evince dall'immagine 6.1, ogni riga del file è composta dall'URL della notizia e da un tag 0/1 che indica se la notizia è vera (tag = 1) oppure fake (tag = 0).

Il python script prende in input il file descritto precedentemente e cicla il contenuto delle celle contenenti gli URL passandolo in input alla funzione main.py del tool (Figura 6.2).

Al termine dell'analisi di tutti gli URL del file Excel in input, lo script crea un nuovo file avente estensione .xlsx dove inserisce i valori dei vari score (Who, When, What, Where, Why, Count Upper, Score Text e Score Image) sulla riga con indice corrispondente all'URL analizzato e sintetizza nelle ultime 2 colonne (sempre di ogni riga) il tag 0/1 relativo all'URL e lo Score totale ottenuto dall'analisi.

```

book = openpyxl.load_workbook('link_site.xlsx')
sheet = book.active

for i in range(1,(len(sheet['A']))):
    a3 = sheet.cell(row=i, column=1)
    label=sheet.cell(row=i, column=2)
    url=a3.value
    labels=label.value

```

Figura 6.2. Ciclo Test - Screen

6.3 Risultati dei test

Un esempio di test è stato eseguito sui seguenti URL:

https://www.theguardian.com/world/2021/mar/30/new-covid-vaccines-needed-within-year-say-scientists	1
https://www.fonteverificata.it/2021/02/21/immune-al-covid-la-cura-definitiva-per-il-virus-potrebbe-nascondersi-nel-dna-di-un-romano-di-33-anni/	0
https://www.bbc.com/news/world-us-canada-56452471	1
https://www.bbc.com/news/uk-politics-47571253	1
https://www.bbc.com/news/uk-england-devon-49282509	1
https://theconversation.com/charles-dickens-wrote-about-the-diphtheria-crisis-of-1856-and-it-all-sounds-very-familiar-151718	1
http://actionnews3.com/veterinarian-accused-sex-300-dogs/	0
https://100giornidaleoni.it/blog/kamala-harris-a-breve-presidente-usa/	0
https://beforeitsnews.com/opinion-conservative/2021/03/situation-update-mar-18th-2021-fake-joe-biden-now-confirmed-its-all-just-a-movie-mike-adams-must-video-3571130.html	0
https://beforeitsnews.com/christian-news/2021/03/real-red-alert-washington-insider-blows-the-whistle-on-the-planned-agenda-for-the-new-world-order-insider-was-killed-a-month-aft	0
https://www.theguardian.com/world/2021/mar/19/france-limits-astazeneca-covid-jab-to-over-55s-despite-ema-green-light	1
https://www.lercio.it/roma-rissa-al-pincio-fra-una-gang-di-cinghiali-e-una-di-gabbiani/	0
https://www.lercio.it/led-zeppelin-annunciano-reunion-faremo-un-disco-di-cover-dei-maneskin/	0
https://www.ansa.it/sito/notizie/mondo/2021/03/30/covid-usa-i-morti-superano-quota-550mila_7f933fbf-6684-45a9-91f8-8585b30d2096.html	1
https://www.ilmessaggero.it/roma/news/maddalena_urbani_morta_overdose_soccorsi_respirati_inchiesta_amica_oggi_ultime_notizie-5864379.html	1
https://www.calcofinanza.it/2021/03/29/diritti-tv-sky-ci-riprova-oggi-assemblea-su-pacchetto-2/	1

Figura 6.3. File Excel - Link Site

A seguito dell'applicazione dello script sono stati restituiti i seguenti risultati:

Score_Who	Score_When	Score_What	Score_Where	Score_Why	Score_Count_upper	Score_SearchText	Score_Images	Score_Tot	Orig_Label
0	0	1	1,15	1	1	1	0,5	5,65	1
0	0	1	0	0	1	1	0	3	0
0	0	1	0	1	1	1	0	4	1
0	1,35	1	1,15	1	1	1	1	7,5	1
0	1,35	1	1	1	1	1	0,428571429	6,778571429	1
1	1,35	1	1	1	1	1	0,5	7,85	1
1	0	1	1,35	0	1	1	0	5,35	0
0	0	0,5	1,35	0	1	1	1	4,85	0
0	0	0,5	1,35	0	1	1	0	3,85	0
0	1	0,5	1,15	0	1	1	0	4,65	0
1	1	1	1	1	1	1	0,5	7,5	1
0	1	0,5	0	1	1	1	1	5,5	0
0	0	0,5	0	1	1	1	1	4,5	0
1	1	1	1	1	1	1	1	8	1
0	1	1	1	1	1	1	0,16	6,16	1
1	1	1	1	1	1	1	0,5	7,5	1

Figura 6.4. Risultati per ogni URL

6.4 Conclusioni

Dai test eseguiti su un campione significativo, composto da notizie di genere, argomento e lingua differente prese dal web, è risultato ottimale il valore 5,5 per la soglia (il che significa che sotto o uguale a tale valore la notizia viene considerata come fake). Con questa configurazione è stato sottoposto al sistema un ulteriore gruppo di notizie prese da fonti diverse. La risposta è stata soddisfacente come da grafici, figura 6.3 e 6.4 in quanto il sistema ha correttamente ottenuto un'accuracy pari al 93,75%. Vale la pena ricordare ancora una volta che l'obiettivo del nostro progetto non è quello di distinguere con certezza una notizia fake da una vera, ma segnalare all'utente il grado di "pericolosità" della notizia stessa per sollecitarne un approfondimento critico alla lettura. La tabella è composta di URL con notizie in varie lingue e con formati diversi. Dalla tabella dei risultati è possibile notare che lo score relativo al Who presenta un numero di 6 score aventi valore 0 su notizie che in realtà presentano un autore. Questo avviene in quanto l'autore della notizia è presente all'interno dell'articolo ma non è correttamente inserito nel campo "author" del codice html relativo all'url. Per quanto riguarda l'estrazione delle località, la funzione Where presenta a volte delle difficoltà nell'individuare la corretta località in quanto, quando si effettua la traduzione in lingua inglese, potrebbe erroneamente venire tradotto il nome della località in maniera errata. Ad esempio, se si considerano le località aventi nome composto come "Testa di Lepre" e "Porte di Roma", queste potrebbero essere tradotte con "Hare Head" e "Gates of Rome", cosa che impedisce al tool di individuare la corretta località. Se si considera il punteggio relativo alla ricerca di notizie simili su altre fonti online, questo assume spesso il valore 1 in quanto la notizia viene individuata sui social network o su altri siti che rappresentano una fonte diversa da quella originale ma che pubblicano lo stesso contenuto. Mentre per lo score delle immagini questo è strettamente correlato all'algoritmo di intelligenza artificiale di Google che è spesso fonte di ambiguità. Ad esempio, la foto di un personaggio vestito elegante (che è coerente con il contenuto della notizia) viene inteso ed analizzato da Google come "Formal Wear" creando ambiguità nel riconoscimento dell'immagine e facendo quindi abbassare lo score della notizia. In ogni caso, per gli scopi che il sistema si prefigge, queste incertezze sono perfettamente accettabili, e potrebbero in futuro essere oggetto di ulteriori sviluppi e miglioramenti del software. Abbiamo realizzato uno strumento che può aiutare veramente l'utente a discriminare l'enorme quantità di informazione a cui è sottoposto.

Capitolo 7

Sviluppi Futuri

Fermo restando che l'obiettivo principale è quello di cercare di aumentare l'accuracy dei vari algoritmi utilizzati, sono stati individuati i seguenti sviluppi futuri:

Servizio Web:

- Creazione di una Web-App pubblica che permetta di visualizzare, attraverso un'interfaccia grafica intuitiva ed user-friendly, tutto il ciclo di vita implementato per attribuire il valore di veridicità, a partire da un URL.

Miglioramento della precisione dello strumento:

- Studiare ed implementare uno scraper web per automatizzare la reverse image di Bing che sembra avere un grado di accuratezza maggiore di quello di Google nell'identificazione dei soggetti nelle immagini.
- Creazione di meccanismi automatizzati per la disambiguazione di date e località. In questo momento la disambiguazione sui classificatori usati viene fatta dall'utente stesso, scegliendo tra una lista di vari input, risultato degli algoritmi di classificazione.
- Aggiunta alla specifica dei requisiti ed implementazione di un algoritmo che studia la differenza in punti tra le grandezze del font del titolo e del testo e che definisca un altro parametro per identificare l'affidabilità di una notizia. (Una fake news generalmente presenta un carattere nel titolo molto più grande rispetto al testo).
- Aggiunta alla specifica dei requisiti ed implementazione di un algoritmo che analizza la credibilità di un autore e verifichi se il nome estratto dal classificatore si riferisca ad un autore reale o ad uno pseudonimo.
- Creazione di un dizionario di forme lessicali enfatiche per l'analisi della presenza di queste all'interno di una notizia. Generalmente una fake news, per convincere, ne utilizza moltissime.
- Analisi automatica per l'estrazione del topic di una notizia.

- Creazione di un dataset con i siti contenenti fake news ed automatizzazione della ricerca delle parole chiave della notizia in analisi su questi per verificarne l'affidabilità.

Plug-in Browser:

- Realizzazione di un'estensione per il Browser web che permetta all'utente di calcolare il grado di affidabilità della notizia direttamente all'interno della pagina visitata, ad ogni sua richiesta.

Bibliografia

- [1] L'autore. <https://www.agcom.it/l-autore>.
- [2] Socialtruth. <http://www.socialtruth.eu/>.
- [3] Textthresher software. www.textthresher.org.
- [4] Prashansa Agrawal, Anjana Parwat Singh, and Sathya Peri. Dehide: Deep learning-based hybrid model to detect fake news using blockchain. 10 2020.
- [5] Monther Aldwairi and Ali Alwahedi. Detecting fake news in social media networks. 11 2018.
- [6] Bloomberg. How facebook fights fake news in the world's largest election. <https://www.livemint.com/companies/news/how-facebook-fights-fake-news-in-the-world-s-largest-election-1555903142347.html>.
- [7] Eunjoo Byeon. Introduction to geopy: Using your latitude longitude data in python, 2020. <https://towardsdatascience.com/things-to-do-with-latitude-longitude-data-using-geopy-python-1d356ed1ae30>.
- [8] Eunjoo Byeon. Introduction to geopy: Using your latitude longitude data in python; getting address, postal code, distance and more, 2020. <https://towardsdatascience.com/things-to-do-with-latitude-longitude-data-using-geopy-python-1d356ed1ae30>.
- [9] Jon Danzig. How fake news caused brexit, 2017. <http://europe.ideasoneurope.eu/2017/11/14/fake-news-caused-brexit/>.
- [10] Scrapy developers. Scrapy 2.4 documentation, 2018. <https://docs.scrapy.org/en/latest/>.
- [11] Agenda Digitale. Le fake news online orientano l'opinione pubblica: che dicono gli ultimi studi, 2020. <https://www.agendadigitale.eu/cultura-digitale/disinformazione-online-e-costruzione-dellopinione-pubblica-se-in-pericolo-e-la-tenuta-delle-democrazie/>.
- [12] Giulia Bonelli e Francesca Camilli. Così le fake news hanno aiutato trump a diventare presidente, 2017. https://www.agi.it/data-journalism/cosiefake_newshannoaiutatotrumpadiventarepresidente_1937892/news/2017-07-08.

- [13] Massimo Esposito. Linguaggio naturale e intelligenza artificiale: a che punto siamo, 2019. <https://www.agendadigitale.eu/cultura-digitale/linguaggio-naturale-e-intelligenza-artificiale-a-che-punto-siamo/>.
- [14] Etui.org. In veles, meeting the producers of fake news. <https://www.etui.org/topics/health-safety-working-conditions/hesamag/the-future-of-work-in-the-digital-era/in-veles-meeting-the-producers-of-fake-news>.
- [15] Claudio Davide Ferrara. Spyder: Ide python per lo sviluppo e l'analisi dei dati, 2018. <https://www.html.it/24/04/2018/spyder-ide-python-per-lo-sviluppo-e-lanalisi-dei-dati/>.
- [16] Python Software Foundation. pytimeextractor 0.1.4, 2017. <https://pypi.org/project/pytimeextractor/>.
- [17] geopy team. Project description geopy 2.1.0, 2021. <https://pypi.org/project/geopy/>.
- [18] Xavier Grangier. Project description, 2020. <https://pypi.org/project/goose3/>.
- [19] Heather C Hughes and Israel Waismel-Manor. The macedonian fake news industry and the 2016 us election. *PS: Political Science & Politics*, 54(1):19–23, 2021.
- [20] il Post. In india si continua a uccidere per video e notizie false diffuse su whatsapp. <https://www.ilpost.it/2018/07/04/india-linciaggi-video-virali-bambini-whatsapp/>.
- [21] Kathleen Jamieson and Joseph Cappella. *Echo Chamber: Rush Limbaugh and the Conservative Media Establishment*. 01 2008.
- [22] PRATEEK JOSHI. Spacy tutorial to learn and master natural language processing (nlp), 2020. <https://www.analyticsvidhya.com/blog/2020/03/spacy-tutorial-learn-natural-language-processing/>.
- [23] Dhruv Khattar, Jaipal Singh, Manish Gupta, and Vasudeva Varma. Mvae: Multimodal variational autoencoder for fake news detection. pages 2915–2921, 05 2019.
- [24] Sunny Kumar. A practical guide to geopy, 2019. <https://sunnykrgupta.github.io/a-practical-guide-to-geopy.html>.
- [25] la Repubblica. Papa francesco ai media: Smascherare le fake news e attenti alle insidie del web, 2020. <https://www.repubblica.it/cronaca/2021/01/23/news/papa-283881201/>.
- [26] 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.
- [27] Libraries.io. pytimeextractor, 2017. <https://libraries.io/pypi/pytimeextractor>.

- [28] Conor McDonald. A short introduction to nlp in python with spacy, 2017. <https://towardsdatascience.com/a-short-introduction-to-nlp-in-python-with-spacy-d0aa819af3ad>.
- [29] meritocracy developers. How to scrape amazon reviews with scrapy, 2020. <https://meritocracy.is/blog/2020/02/27/how-to-scrape-amazon-reviews-with-scrapy/>.
- [30] University of Michigan. "fake news," lies and propaganda: How to sort fact from fiction". 2021.
- [31] Parsehub. What is web scraping and what is it used for?, 2018. <https://www.parsehub.com/blog/what-is-web-scraping/>.
- [32] John Pettus. Fiskkit. <https://fiskkit.com/>.
- [33] Julie Posetti and Alice Matthews. A short guide to the history of'fake news' and disinformation. *International Center for Journalists*, 7:2018–07, 2018.
- [34] Python. What is python?, 2020. <https://www.python.org/doc/essays/blurb/>.
- [35] Kenneth Rapoza. Can 'fake news' impact the stock market?, 2017. <https://www.forbes.com/sites/kenrapoza/2017/02/26/can-fake-news-impact-the-stock-market/?sh=29db36d02fac>.
- [36] Kai Shu, Limeng Cui, Suhang Wang, Dongwon Lee, and Huan Liu. *DEFEND: Explainable Fake News Detection*, page 395–405. Association for Computing Machinery, New York, NY, USA, 2019.
- [37] Kai Shu, Deepak Mahudeswaran, and Huan Liu. Fakenewstracker: a tool for fake news collection, detection, and visualization. *Computational and Mathematical Organization Theory*, 25, 03 2019.
- [38] 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.
- [39] Taranjeet Singh. Natural language processing with spacy in python, 2020. <https://realpython.com/natural-language-processing-spacy-python/>.
- [40] Jacob Soll. The long and brutal history of fake news. *POLITICO Magazine*.
- [41] spaCy Team. spacy 101: Everything you need to know, 2020. <https://spacy.io/usage/spacy-101>.
- [42] Sas team. Natural language processing (nlp): What it is and why it matters, 2019. <https://www.sas.com/it/t/insights/analytics/what-is-natural-language-processing-nlp.html>.
- [43] Facebook Teams. Working to stop misinformation and false news, 2020. <https://www.facebook.com/formedia/blog/working-to-stop-misinformation-and-false-news>.

- [44] Marianna Tirrito. La regola delle 5w, 2018.
- [45] w3school. Python introduction, 2020. https://www.w3schools.com/python/python_intro.asp.
- [46] Wikipedia. Haversine formula, 2016. https://en.wikipedia.org/wiki/Haversine_formula.
- [47] workfront. The 5 ws (and 1 h) that should be asked of every project!, 2018. <https://www.workfront.com/blog/project-management-101-the-5-ws-and-1-h-that-should-be-asked-of-every-project>.
- [48] Michael Yin. Scrapy tutorial 1: Scrapy vs beautiful soup, 2021. <https://www.accordbox.com/blog/scrapy-tutorial-1-scrapy-vs-beautiful-soup/>.
- [49] Xinyi Zhou, Atishay Jain, Vir Phoha, and Reza Zafarani. Fake news early detection: A theory-driven model, 04 2019.

Glossario

Disinformazione Creare e/o diffondere INTENZIONALMENTE informazioni false, fuorvianti o non oggettive, distorcendo o alterando la realtà dei fatti con l'intento di ingannare e manipolare il destinatario del messaggio. 1, 3

Fake News Notizia, storia o bufala creata appositamente con lo scopo di disinformare o ingannare deliberatamente i lettori. 3

False News Notizia o storia di autenticità falsa ma di cui non ne conosciamo l'intenzione. 3

Misinformation Condivisione di informazioni false o non corrette ma in cui è assente la componente intenzionale d'inganno. 1, 3

Rumor Notizia, discorso o opinione ampiamente diffusa ma senza una fonte certa. Non si conosce né la sua autenticità, né la sua intenzione. 3

Satire News Notizia presentata in un formato tipico del giornalismo e chiamata satira a causa del suo contenuto. La satira giornalistica si basa molto sull'ironia e sull'umorismo. Non si conosce la sua autenticità ed ha una intenzione non malevola. Funziona solo se si sa che è inventata. 3

Ringraziamenti

Ringrazio il mio relatore, il **Prof. Claudio Cilli**, per avermi dato l'opportunità di sviluppare questo progetto dandomi nuovi spunti per la mia formazione ed ampliare così il mio bagaglio di esperienze e conoscenze.

Ringrazio i miei correlatori **Dott. Ing. Giulio Magnanini** e **Dott. Fabrizio Venettoni** che mi hanno supportato in questo percorso guidandomi sempre verso la giusta direzione.

Ringrazio inoltre i miei colleghi di Università, **A. Magnante**, **A. Berti**, **F. Bianca** e **M. Prandini** senza i quali questo percorso universitario non sarebbe stato lo stesso.