

DIPARTIMENTO DI INFORMATICA CORSO DI LAUREA IN INFORMATICA

TESI DI LAUREA IN

MODELLI E METODI PER LA SICUREZZA DELLE APPLICAZIONI

Phishing: Sviluppo di una estensione Chrome per il riconoscimento di siti malevoli mediante AI ​

RELATORE:

Prof. Donato Impedovo

CORRELATORI:

Dott. Francesco Castro

LAUREANDO:

Nicolò Orsi

ANNO ACCADEMICO 2023 - 2024

Abstract

La presente ricerca si propone di sviluppare un'estensione per il browser Chrome dedicata al riconoscimento degli URL malevoli tramite l'utilizzo di algoritmi di machine learning. L'approccio adottato coinvolge la progettazione e l'implementazione di tecniche di apprendimento automatico e di analisi avanzate, mirate a identificare con efficacia e tempestività i siti di phishing e altre minacce online. L'obiettivo principale è migliorare la sicurezza degli utenti durante la navigazione su internet, consentendo una rilevazione rapida e accurata degli URL pericolosi nelle fasi iniziali dell'accesso.

La metodologia di ricerca comprende la raccolta di un dataset rappresentativo di URL malevoli e benigni, seguito dall'addestramento di modelli predittivi utilizzando algoritmi sofisticati come le reti neurali, le macchine a vettori di supporto (SVM) e gli alberi decisionali. I risultati attesi includono un sistema di rilevazione altamente preciso e reattivo, in grado di contribuire significativamente alla protezione degli utenti durante la navigazione web.

Questa ricerca aspira a fornire una base robusta per ulteriori sviluppi nell'ambito della sicurezza informatica, promuovendo una gestione proattiva delle minacce e una difesa avanzata contro i tentativi di phishing e altre forme di attacco online.

Sommario

[Introduzione 5](#_Toc170403731)

[Stato dell’arte 15](#_Toc170403732)

[1.1 Titolo Paragrafo 15](#_Toc170403733)

[1.1.1 Titolo Sottoparagrafo 16](#_Toc170403734)

[Capitolo 2 17](#_Toc170403735)

[2.1 Introduction 17](#_Toc170403736)

[2.2 Titolo Paragrafo 19](#_Toc170403737)

[2.2.1 Titolo Sottoparagrafo 19](#_Toc170403738)

[Capitolo 3 21](#_Toc170403739)

[3.1 Titolo Paragrafo 21](#_Toc170403740)

[3.1.1 Titolo Sottoparagrafo 22](#_Toc170403741)

[Capitolo 4 23](#_Toc170403742)

[4.1 Titolo Paragrafo 23](#_Toc170403743)

[4.1.1 Titolo Sottoparagrafo 24](#_Toc170403744)

[Capitolo 5 Conclusioni 25](#_Toc170403745)

[Bibliografia 27](#_Toc170403746)

[Ringraziamenti 28](#_Toc170403747)

# Introduzione

Nell'era digitale attuale, caratterizzata da una crescente interconnessione e dipendenza dai servizi online, la sicurezza informatica rappresenta un fattore chiave. Le minacce online, in particolare gli attacchi di phishing, sono diventate sempre più sofisticate, mettendo a rischio la privacy e la sicurezza degli utenti. Secondo il rapporto dell'Anti-Phishing Working Group (APWG), il numero di attacchi di phishing ha raggiunto livelli record nel 2022, con oltre 1 milione di attacchi segnalati solo nel primo trimestre dell'anno.

Tradizionalmente, il riconoscimento degli URL malevoli avveniva tramite tecniche basate su liste nere (blacklists) e regole statiche. Le blacklists contengono elenchi di URL noti per essere dannosi, e qualsiasi richiesta a un URL presente in questa lista viene bloccata. Parallelamente, le regole statiche si basano su pattern fissi e caratteristiche predefinite degli URL per identificare quelli sospetti. Sebbene queste tecniche abbiano fornito una difesa iniziale contro molte minacce, presentano diversi limiti significativi:

1. Aggiornamenti Manuali: Le blacklists richiedono aggiornamenti costanti e manuali per includere nuovi URL malevoli. Questo processo è spesso lento e non riesce a tenere il passo con la rapida evoluzione delle minacce.
2. Incapacità di Generalizzare: Le regole statiche si basano su caratteristiche specifiche che possono essere facilmente aggirate dai malintenzionati. Gli attacchi di phishing moderni utilizzano tecniche di offuscamento e mutazione per evitare il rilevamento.
3. Alto Tasso di Falsi Positivi e Negativi: Le tecniche tradizionali possono spesso classificare erroneamente URL benigni come malevoli (falsi positivi) e viceversa (falsi negativi), riducendo l'efficacia complessiva del sistema di sicurezza.

In questo contesto, l'approccio basato sull’utilizzo di algoritmi AI offre significativi vantaggi per il riconoscimento degli URL malevoli.

L’intelligenza artificiale (IA) è un campo dell’informatica che si occupa dello sviluppo di sistemi e tecnologie capaci di simulare alcune capacità cognitive umane. Come afferma Pedro Domingos, autore del libro "The Master Algorithm" [1], l’IA è la scienza di fare macchine intelligenti, specialmente programmi software intelligenti. È l’arte di creare macchine che eseguono attività che richiedono intelligenza quando fatte dall’uomo, come il ragionamento, l’apprendimento, il riconoscimento di pattern, la comprensione del linguaggio naturale e il problem solving. Questa disciplina è diventata un argomento di grande rilevanza nella società moderna, influenzando profondamente molti aspetti della vita umana, dall’economia alla salute, dall’istruzione alla politica.

Come sottolinea Max Tegmark nel suo libro "Life 3.0: essere umani nell’era dell’intelligenza artificiale" [2], le tecnologie cognitive sono la più grande invenzione dell’umanità che amplificherà l’umanità come nessuna tecnologia ha fatto fino ad ora. Queste tecniche avanzate sono utilizzate in molti settori della nostra vita quotidiana, tra cui la medicina, dove aiutano nella diagnosi e nel trattamento di malattie. Secondo un articolo pubblicato sulla rivista Nature Medicine [3], le tecnologie intelligenti sono fondamentali per l’evoluzione della medicina di precisione, l’analisi predittiva e la modellazione dei dati possono migliorare la diagnosi precoce e la personalizzazione dei trattamenti.

Nel settore del marketing, le tecniche di apprendimento automatico analizzano i dati dei consumatori per personalizzare le offerte; infatti, secondo un rapporto di McKinsey [4], queste tecnologie possono aumentare la precisione delle previsioni di vendita fino al 30% grazie all’analisi dei big data e all’apprendimento automatico. Inoltre, si sono dimostrate un alleato prezioso nella protezione delle applicazioni web, offrendo capacità avanzate di rilevamento e risposta alle minacce. Come afferma Stephen Hawking nel suo libro "Brief Answer to the Big Questions" [5], l’intelligenza artificiale potrebbe diventare il più grande evento nella storia della nostra civiltà, ma come ogni avanzamento tecnologico, ci sono anche dei rischi, perciò è fondamentale investire nella ricerca di sistemi di intelligenza artificiale sicuri e affidabili.

Secondo il libro "Artificial Intelligence: A Modern Approach" di Stuart Russel e Peter Norvig [6], queste applicazioni sono supportate da algoritmi di apprendimento automatico e reti neurali profonde, che consentono alle macchine di apprendere dai dati e migliorare le loro prestazioni nel tempo. Ci sono diversi approcci nell’ambito delle tecnologie cognitive, tra cui:

* Machine Learning: una branca dell’intelligenza artificiale che si occupa dello sviluppo di algoritmi e modelli capaci di apprendere dai dati e migliorare le loro prestazioni nel tempo, senza richiedere una programmazione esplicita per svolgere determinati compiti. Questo processo di apprendimento è fondamentale per consentire ai sistemi di adattarsi a nuove informazioni e di prendere decisioni autonome basate su di esse. Una delle caratteristiche centrali del Machine Learning è la capacità di identificare pattern e relazioni nei dati, permettendo al sistema di fare previsioni o prendere decisioni in base a nuove informazioni. Questo processo può essere suddiviso in diverse tecniche, tra cui:
  1. Apprendimento Supervisionato: un approccio in cui il modello viene addestrato utilizzando un insieme di dati di input e le rispettive etichette di output corrispondenti. L’obiettivo è insegnare al modello la relazione tra gli input e gli output desiderati, in modo che possa generalizzare correttamente su nuovi dati non visti durante il processo di addestramento.
  2. Apprendimento Non Supervisionato: coinvolge l’addestramento del modello utilizzando solo dati di input, senza etichette di output corrispondenti. L’obiettivo principale è scoprire pattern, strutture o relazioni nascoste nei dati, senza una guida esterna. Questo approccio trova applicazioni in ambiti come il clustering e l’analisi delle componenti principali.
  3. Apprendimento Semi Supervisionato: combina elementi dell’apprendimento supervisionato e non supervisionato, dove il modello viene addestrato utilizzando un insieme di dati di input, di cui solo una parte è etichettata.

L’adozione diffusa del Machine Learning è stata resa possibile da una serie di fattori, tra cui l’aumento della potenza di calcolo, la disponibilità di grandi quantità di dati e l’avanzamento degli algoritmi e delle tecniche di ottimizzazione. Inoltre, la crescente consapevolezza sull’importanza dei dati e la loro corretta gestione ha contribuito a rendere il Machine Learning una priorità per molte aziende e organizzazioni. Queste considerazioni sono supportate dalle conclusioni di diversi autori nel campo dell’apprendimento automatico, tra cui il libro "Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow" di Aurelien Geron [7] che fornisce un’ampia panoramica delle tecniche e degli strumenti utilizzati nel Machine Learning, offrendo esempi pratici e casi di studio. Allo stesso modo, "Pattern Recognition and Machine Learning" di Christopher M. Bishop [8] approfondisce i principi fondamentali del Machine Learning esplorando le sue applicazioni in vari settori e contesti.

* Deep Learning: una branca dell’intelligenza artificiale che si basa sull’uso di reti neurali profonde per apprendere da dati complessi. È chiamato "profondo" perché coinvolge reti neurali con molti strati, che possono comprendere decine, centinaia o addirittura migliaia di strati di neuroni artificiali. L’evoluzione del Deep Learning è stata trainata dall’enorme disponibilità di dati e dalla potenza di calcolo crescente, che ha permesso la creazione e l’addestramento di reti neurali con migliaia, se non milioni, di parametri. Questo ha reso possibile l’analisi di dati astronomici, la diagnosi medica avanzata, il riconoscimento di immagini e volti, la traduzione automatica e molte altre applicazioni che solo pochi anni fa sembravano fantascienza. Per comprendere al meglio il fenomeno del Deep Learning è fondamentale esplorare le sue radici e i suoi fondamentali teorici. Il libro "Deep Learning" di Goodfellow, Bengio e Courville [9] rappresenta una guida esaustiva e accessibile, fornendo un’analisi dettagliata dei principi fondamentali, degli algoritmi e delle applicazioni del Deep Learning. Da qui possono essere appresi i concetti base delle reti neurali profonde, come l’architettura delle reti, il funzionamento dell’addestramento tramite backpropagation e l'importanza delle funzioni di attivazione. Tuttavia, è necessario esaminare le sue applicazioni pratiche e le sfide che incontra nel mondo reale. L’articolo "Understanding Deep Learning Requires Rethinking Generalization" di Chiyuan Zhang [10] offre una prospettiva su uno degli aspetti più cruciali del Deep Learning: la capacità di generalizzazione. Questo concetto si riferisce alla capacità di un modello di adattarsi efficacemente a nuovi dati non visti durante l’addestramento, ed è un tema centrale nell’evitare il sovradattamento e nel garantire la robustezza dei modelli di Deep Learning. È importante anche considerare le sue limitazioni; nell’articolo "Deep Learning: A Critical Appraisal" di Gary Marcus [11], viene fornita una valutazione critica delle capacità attuali del Deep Learning, evidenziando le sue lacune nell’affrontare la comprensione del linguaggio naturale, il ragionamento casuale e la capacità di astrazione. Un'altra area dell’intelligenza artificiale che il Deep Learning sta trasformando è la visione artificiale. Nel libro "Deep Learning for Computer Vision" di Rajalingappaa [12] viene offerta una panoramica dettagliata su come implementare e utilizzare reti neurali convoluzionali per una varietà di applicazioni di visione artificiale, come il riconoscimento di immagini e la segmentazione.
* Computer Vision: una branca delle tecnologie cognitive che unisce concetti e metodologie provenienti dall’informatica, dall’ingegneria e dalla psicologia cognitiva al fine di sviluppare sistemi informatici in grado di comprendere, interpretare e analizzare le informazioni visive, simili all’abilità umana. Uno dei concetti fondamentali della visione artificiale è il riconoscimento degli oggetti, che consiste nell’identificare e classificare gli oggetti presenti in un’immagine o in un video. Un'altra area importante della visione artificiale è la segmentazione delle immagini, che consiste nel dividere un’immagine in regioni significative o oggetti individuali. Negli ultimi anni, l’evoluzione delle tecnologie di intelligenza artificiale, in particolare degli algoritmi di apprendimento automatico, ha rivoluzionato il modo in cui le organizzazioni analizzano i dati e prendono decisioni. Gli algoritmi di machine learning sono strumenti potenti che consentono di estrarre insight significativi da grandi quantità di dati, identificare pattern complessi e fare previsioni accurate.

Sono molteplici gli usi degli algoritmi di apprendimento automatico:

* Classificazione e Predizione: Gli algoritmi di classificazione sono cruciali per suddividere i dati in categorie o gruppi in base alle loro caratteristiche distintive. Questa suddivisione consente alle organizzazioni di prendere decisioni informate in vari settori. Ad esempio, possono essere utilizzati per classificare transazioni finanziarie come fraudolente o legittime, migliorando l’efficacia dei sistemi di rilevamento delle frodi finanziarie. La predizione, d’altra parte, si basa sull’elaborazione di modelli utilizzando dati storici per fare previsioni su eventi futuri.
* Raccomandazioni personalizzate: Analizzando i comportamenti passati e le preferenze degli utenti, questi algoritmi possono suggerire prodotti, contenuti o servizi che potrebbero interessare l’utente. Ad esempio, piattaforme di streaming come Netflix utilizzano algoritmi di raccomandazione per suggerire film e serie TV in base alle preferenze di visualizzazione degli utenti.
* Riconoscimento di pattern e analisi dei dati: Questi algoritmi possono individuare correlazioni complesse o tendenze nascoste nei dati che possono sfuggire all’analisi umana. Possono essere utilizzati per analizzare dati non strutturati come testo, immagini o suoni. Ad esempio, gli algoritmi di riconoscimento facciale utilizzati nei sistemi di sicurezza o nelle applicazioni di identificazione personale si basano sull’analisi dei pattern presenti nelle immagini.
* Ottimizzazione dei processi decisionali: Questi algoritmi possono essere utilizzati per automatizzare decisioni ripetitive o complesse, riducendo il tempo e gli errori umani. Ad esempio, possono essere impiegati per ottimizzare la gestione della catena di approvvigionamento, il routing del traffico nelle reti di trasporto o le strategie di marketing online.

Gli algoritmi utilizzati sono:

* K-Nearest Neighbors (K-NN): È uno degli algoritmi più intuitivi e semplici nel campo dell’apprendimento automatico. La sua idea fondamentale si basa sull'assunzione che i punti dati con caratteristiche simili si trovino vicini nello spazio delle caratteristiche. Quando si tratta di classificare un nuovo punto dati, K-NN cerca i K punti più vicini all'interno dei dati di addestramento e determina l'etichetta di classe più comune tra questi punti (per la classificazione) o calcola la media dei valori di questi punti (per la regressione). Questo approccio rende K-NN particolarmente adatto per problemi in cui la struttura dei dati è complessa e non linearmente separabile. Secondo "Pattern Recognition and Machine Learning" di Christopher M. Bishop [13], la scelta del parametro K è cruciale per le prestazioni di K-NN, un valore troppo piccolo può rendere l'algoritmo sensibile al rumore e alle fluttuazioni dei dati, mentre un valore troppo grande può portare a una perdita di informazioni. Questo parametro può essere ottimizzato utilizzando tecniche come la cross-validazione, che aiuta a trovare il giusto equilibrio tra bias e varianza. Inoltre, la scelta della misura di distanza è un aspetto importante di K-NN; sebbene la distanza euclidea sia la misura più comunemente utilizzata, in alcuni contesti altre misure come la distanza di Manhattan o la distanza di Mahalanobis possono essere più appropriate in base alla natura dei dati. Secondo "Introduction to Machine Learning with Python" di Andreas C. Müller e Sarah Guido [14], una limitazione di K-NN è la sua computazionalità, poiché deve memorizzare tutti i dati di addestramento, diventando inefficiente su grandi insiemi di dati. Inoltre, il calcolo delle distanze può diventare computazionalmente costoso su dati ad alta dimensionalità, poiché l'aumento delle dimensioni rende difficile definire la 'vicinanza' tra i punti nel grande spazio delle caratteristiche.
* Support Vector Machine (SVM): È un algoritmo di apprendimento supervisionato utilizzato per la classificazione e la regressione. L'obiettivo del SVM è trovare un iperpiano nello spazio delle caratteristiche che meglio separi le diverse classi dei dati di addestramento. Questo iperpiano è definito in modo tale da massimizzare il margine tra le classi, cioè la distanza tra l'iperpiano e i punti dati più vicini di ciascuna classe, chiamati vettori di supporto. Secondo "Pattern Recognition and Machine Learning" di Christopher M. Bishop [15], SVM è un algoritmo che cerca di massimizzare il margine tra le classi durante il processo di addestramento, il che lo rende particolarmente efficace per dataset con una chiara separazione lineare tra le classi. Un aspetto chiave di SVM è l'uso di funzioni kernel, che consentono di trasformare lo spazio delle caratteristiche in uno spazio di dimensioni superiori, dove la separazione lineare dei dati può essere più facilmente ottenuta. Questo è particolarmente utile quando i dati non sono linearmente separabili nello spazio delle caratteristiche originale. Secondo "Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow" di Aurélien Géron [16], SVM è anche noto per la sua capacità di generalizzazione, il che significa che è in grado di gestire bene anche dataset con dimensioni elevate e complessità, riducendo il rischio di overfitting. Tuttavia, la scalabilità di SVM può essere un problema su grandi insiemi di dati, poiché il tempo di addestramento e la complessità computazionale aumentano con il numero di campioni e di dimensioni delle caratteristiche.
* Random Forest: È un algoritmo di apprendimento supervisionato che opera creando un insieme di alberi decisionali durante il processo di addestramento. Ogni albero decisionale viene costruito utilizzando un sottoinsieme casuale dei dati di addestramento e un sottoinsieme casuale delle caratteristiche. Durante la fase di predizione, l'output del modello è determinato dalla media (per regressione) o dalla modalità (per classificazione) delle previsioni dei singoli alberi. Secondo "Pattern Recognition and Machine Learning" di Christopher M. Bishop [17], il Random Forest è noto per la sua capacità di ridurre l'overfitting e migliorare la generalizzazione rispetto agli alberi decisionali singoli. Questo è dovuto alla combinazione di molteplici alberi, ognuno dei quali può imparare da una prospettiva diversa dei dati. Un vantaggio significativo di Random Forest è la sua robustezza ai dati mancanti e al rumore nei dati di addestramento, poiché l'algoritmo considera solo un sottoinsieme casuale delle caratteristiche per ogni albero, è in grado di gestire efficacemente la presenza di rumore e outlier nei dati. Secondo "Introduction to Machine Learning with Python" di Andreas C. Müller e Sarah Guido [18], il Random Forest è anche facilmente interpretabile poiché fornisce stime di importanza delle caratteristiche che indicano quali variabili sono più influenti nel processo decisionale del modello. Tuttavia, Random Forest può essere meno efficace su dataset con relazioni complesse o non lineari tra le caratteristiche e l'output desiderato. Inoltre, poiché l'algoritmo è basato sull'aggregazione di molti alberi decisionali, può essere computazionalmente costoso da addestrare e richiedere più risorse computazionali rispetto ad altri modelli più semplici.
* Decision Tree: È un modello di apprendimento supervisionato utilizzato per la classificazione e la regressione. L'obiettivo di un albero decisionale è quello di creare una struttura ad albero in cui ogni nodo interno rappresenta una caratteristica del dataset, ogni arco rappresenta una decisione basata su quella caratteristica e ogni foglia rappresenta l'etichetta di classe o il valore di output. Secondo "Pattern Recognition and Machine Learning" di Christopher M. Bishop [8], gli alberi decisionali sono noti per la loro capacità di rappresentare decisioni complesse in modo interpretabile e intuitivo. Questo li rende particolarmente utili per problemi in cui è importante comprendere e spiegare il processo decisionale del modello. Un vantaggio significativo degli alberi decisionali è la loro capacità di gestire sia dati numerici che categorici, senza la necessità di normalizzazione dei dati. Inoltre, sono robusti a dati mancanti e a rumore nei dati di addestramento. Secondo "Introduction to Machine Learning with Python" di Andreas C. Müller e Sarah Guido [13], gli alberi decisionali sono anche noti per la loro velocità di addestramento e di predizione, poiché la complessità dell'albero è controllata dal numero di divisioni e dalla profondità dell'albero, possono essere addestrati e utilizzati in tempo ragionevole anche su grandi insiemi di dati. Tuttavia, un problema comune degli alberi decisionali è la tendenza all'overfitting, soprattutto su dataset complessi o ad alta dimensionalità. Questo può essere limitato attraverso tecniche come la potatura dell'albero, che limita la sua profondità o complessità per migliorare le prestazioni di generalizzazione.
* Reti Neurali: Sono modelli computazionali ispirati dal funzionamento del cervello umano. Comprendono un insieme di neuroni artificiali organizzati in strati, ognuno dei quali è connesso agli strati precedenti e successivi attraverso pesi sinaptici. L'apprendimento avviene aggiustando iterativamente questi pesi sinaptici durante il processo di addestramento, al fine di ridurre l'errore tra le previsioni del modello e gli obiettivi desiderati. Secondo "Deep Learning" di Ian Goodfellow, Yoshua Bengio e Aaron Courville [9], le reti neurali sono notevoli per la loro capacità di apprendere rappresentazioni complesse e stratificate dei dati. Grazie alla loro architettura multistrato, le reti neurali possono catturare automaticamente caratteristiche gerarchiche dai dati, rendendole adatte per una vasta gamma di problemi di apprendimento automatico. Un vantaggio significativo delle reti neurali è la loro flessibilità e capacità di adattarsi a una vasta gamma di problemi di machine learning, compresi problemi di classificazione, regressione, clustering, generazione di testo e altro ancora. Le reti neurali possono essere progettate e addestrate in modo da soddisfare le esigenze specifiche del problema in questione, utilizzando una varietà di architetture, funzioni di attivazione, ottimizzatori e altri iperparametri. Secondo "Neural Networks and Deep Learning: A Textbook" di Charu C. Aggarwal [15], le reti neurali sono anche notevoli per la loro capacità di generalizzazione. Se adeguatamente addestrate su un set di dati rappresentativo, possono generalizzare bene su nuovi dati non visti durante il processo di addestramento, evitando l'overfitting. Tuttavia, un limite delle reti neurali è che possono richiedere una grande quantità di dati di addestramento e risorse computazionali per ottenere buone prestazioni, specialmente su architetture più complesse come le reti neurali profonde. Inoltre, le reti neurali possono essere soggette al fenomeno dell'overfitting, soprattutto su dataset di piccole dimensioni o con rumore nei dati.
* Regressione Lineare: È un metodo statistico utilizzato per modellare la relazione tra una variabile dipendente continua e una o più variabili indipendenti. In particolare, la regressione lineare cerca di trovare la migliore linea retta (o iperpiano in spazi multidimensionali) che si adatta ai dati osservati, minimizzando la somma dei quadrati degli errori tra i valori predetti dal modello e i valori effettivi della variabile dipendente. Secondo "Introduction to Statistical Learning" di Gareth James, Daniela Witten, Trevor Hastie e Robert Tibshirani [16], la regressione lineare è apprezzata per la sua semplicità e interpretabilità. La relazione lineare tra le variabili rende il modello facile da comprendere e spiegare, il che lo rende particolarmente adatto per l'analisi esplorativa dei dati e la comunicazione dei risultati. Un vantaggio significativo della regressione lineare è la sua efficienza computazionale e la relativa semplicità nell'implementazione. Poiché il modello è lineare, l'addestramento e l'interpretazione dei risultati sono più rapidi rispetto a modelli più complessi come le reti neurali o gli alberi decisionali. Secondo "Applied Predictive Modeling" di Max Kuhn e Kjell Johnson [17], la regressione lineare offre anche una buona base per la comprensione di concetti statistici fondamentali come l'importanza delle variabili, il test di ipotesi e l'intervallo di confidenza. Tuttavia, un limite della regressione lineare è che assume una relazione lineare tra le variabili, il che può non essere sempre realistico nei dati del mondo reale. Inoltre, la regressione lineare può non essere in grado di catturare relazioni complesse o non lineari tra le variabili, richiedendo l'uso di modelli più complessi quando necessario.
* Regressione logistica: È un metodo statistico utilizzato per modellare la relazione tra una variabile dipendente binaria o di tipo dummy e una o più variabili indipendenti. Contrariamente al nome, la regressione logistica è utilizzata principalmente per la classificazione piuttosto che per la regressione. L'obiettivo della regressione logistica è stimare la probabilità che un'osservazione appartenga a una determinata classe, utilizzando una funzione logistica per trasformare un'equazione lineare delle variabili indipendenti. Secondo "Introduction to Statistical Learning" di Gareth James, Daniela Witten, Trevor Hastie e Robert Tibshirani [16], la regressione logistica è apprezzata per la sua interpretabilità e la sua capacità di modellare relazioni non lineari tra le variabili predittive e la variabile di risposta. La funzione logistica utilizzata nella regressione logistica produce una curva a forma di sigmoide che varia tra 0 e 1, rappresentando la probabilità predetta di appartenenza alla classe positiva. Un vantaggio significativo della regressione logistica è la sua interpretazione diretta dei coefficienti stimati, che rappresentano il cambio nella log-odds della variabile dipendente associato a un cambiamento unitario nelle variabili indipendenti. Questo rende la regressione logistica utile per identificare l'importanza relativa delle variabili predittive e comprendere il modo in cui influenzano la probabilità di appartenenza alla classe. Inoltre, la regressione logistica è computazionalmente efficiente e facile da implementare, rendendola adatta anche per dataset di grandi dimensioni. Rispetto ai modelli più complessi come le reti neurali, la regressione logistica è meno suscettibile all'overfitting e richiede meno dati per ottenere stime affidabili dei parametri. Tuttavia, la regressione logistica assume una relazione lineare tra le variabili indipendenti e la log-odds della variabile dipendente, il che può essere limitante in alcune situazioni in cui le relazioni sono non lineari. Inoltre, la regressione logistica è sensibile alla presenza di dati outlier e multicollinearità tra le variabili predittive.

# Stato dell’arte

Nell’articolo “Real-time phishing detection using deep learning methods by extensions”[18] si propone un metodo per la rilevazione in tempo reale del phishing attraverso l'uso di estensioni per browser che impiegano modelli di deep learning, in particolare reti neurali convoluzionali (CNN). L'estensione protegge gli utenti verificando, avvisando e prevenendo connessioni non affidabili. I risultati ottenuti mostrano un'accuratezza del 98.4% nella rilevazione di URL maligni utilizzando una CNN.

Processo di Addestramento:

Il processo di addestramento del modello è suddiviso in più fasi:

1. Preelaborazione dei Dati:
   * Raccolta di un dataset di URL contenente 651,191 campioni, suddivisi in URL benigni, defacement, phishing e malware.
   * Estrazione delle caratteristiche degli URL come protocollo, sottodominio, nome di dominio, porta, percorso, query, parametri e frammento.
   * Tokenizzazione degli URL utilizzando un dizionario di 270 caratteri per convertire gli URL in vettori numerici.
2. Architettura del Modello:
   * CNN: La rete neurale convoluzionale è composta da livelli convoluzionali per l'estrazione delle caratteristiche, livelli di pooling per ridurre la dimensionalità e un livello completamente connesso per la classificazione finale.
   * CNN-LSTM: Una variante che include un livello LSTM per catturare le dipendenze temporali nei dati, utile per analizzare sequenze di caratteri negli URL.
3. Fasi di Valutazione:
   * Metriche di Valutazione: Accuratezza, precisione, richiamo e F1-score.
   * Configurazione Hardware: Gli esperimenti sono stati condotti utilizzando server con configurazioni hardware avanzate, tra cui CPU a 36 core, 64 GB di RAM e GPU NVIDIA® GeForce RTX™ 3060.

Estensione del Browser:

L'estensione software, sviluppata in JavaScript, è installata sui browser con kernel Chromium (es. Google Chrome e Microsoft Edge). Le sue principali funzioni includono:

* Invio dell'URL al Server: L'URL inserito dall'utente viene inviato al server per la classificazione.
* Classificazione in Tempo Reale: Il server utilizza il modello CNN per classificare l'URL come benigno o phishing.
* Notifica all'Utente: Se l'URL è maligno, l'estensione notifica immediatamente l'utente con un avviso.

Risultati Sperimentali:

* Accuratezza del Modello: La CNN ha raggiunto un'accuratezza del 98.4%, superando altri modelli come la regressione logistica (LR), gli alberi decisionali (DT), le foreste casuali (RF) e le macchine a vettori di supporto (SVM).
* Confronto tra Modelli: La CNN ha dimostrato un'elevata precisione e richiamo nella rilevazione degli URL di phishing grazie alla sua architettura e all'uso di un set di caratteri esteso.

L'implementazione di un'estensione per browser basata su deep learning, in particolare CNN, offre un metodo efficace per la rilevazione in tempo reale degli URL maligni. Questo approccio migliora significativamente la sicurezza degli utenti durante la navigazione web. Il modello CNN ha dimostrato superiorità rispetto ad altri modelli di machine learning testati, evidenziando la sua efficacia nell'identificazione delle minacce informatiche.

L'articolo “Classification of Malicious Websites Using Machine Learning Based on URL Characteristics”[19] presenta un approccio basato su machine learning per la classificazione di siti web maligni utilizzando le caratteristiche degli URL. Utilizzando un dataset di URL aggiornato, l'algoritmo Random Forest ha raggiunto un'accuratezza del 95.68% nel rilevare siti web maligni. Il software sviluppato per la rilevazione e l'avviso di siti web maligni è implementato sia come applicazione web che come estensione del browser.

Il processo di addestramento del modello di rilevamento dei siti web maligni è suddiviso in due fasi principali:

1. Fase di Addestramento:
   * Dataset: Il dataset di addestramento include 213,345 URL suddivisi in cinque categorie: siti benigni, defacement, phishing, malware e spam.
   * Estrazione delle Caratteristiche: Vengono estratte 20 caratteristiche principali dagli URL, come la presenza di un indirizzo IP, la lunghezza dell'URL, l'uso di servizi di accorciamento, la presenza del protocollo HTTPS, il numero di cifre e l'anomalia dell'URL.
   * Algoritmi di Machine Learning: Gli algoritmi utilizzati includono Decision Tree, Random Forest, K-Nearest Neighbors (K-NN) e Adaptive Boost (AdaBoost).
2. Fase di Rilevamento:
   * Estrazione delle Caratteristiche: Gli URL vengono monitorati ed estratti utilizzando il classificatore addestrato.
   * Classificazione: Gli URL vengono classificati come benigni, defacement, phishing, malware o spam.

Algoritmi di Classificazione:

* Decision Tree: Utilizza un diagramma ad albero per prendere decisioni basate su eventi casuali, spese di risorse e benefici.
* Random Forest: Genera alberi decisionali su campioni di dati selezionati casualmente, predice per ogni albero e seleziona la migliore decisione tramite votazione.
* K-Nearest Neighbors (K-NN): Algoritmo di apprendimento supervisionato che classifica gli oggetti basandosi sulla vicinanza ai vicini più prossimi nel dataset.
* Adaptive Boost (AdaBoost): Algoritmo potente che accelera la creazione di un forte classificatore selezionando buone caratteristiche da una famiglia di deboli classificatori e combinandoli linearmente.

Dataset:

Il dataset utilizzato comprende 213,345 URL, suddivisi in cinque categorie: Benign (0), Defacement (1), Phishing (2), Malware (3) e Spam (4). Gli URL maligni sono stati raccolti da fonti come OpenPhish, Phishtank, Zone-H e WEBSPAM-UK2007. Il processo di pulizia dei dati ha rimosso gli URL duplicati e quelli che non includevano tutte le caratteristiche selezionate.

Risultati e Discussione:

* Valutazione delle Performance: Gli algoritmi di machine learning sono stati valutati utilizzando 10-fold cross-validation. Le metriche di performance utilizzate includono Precision, Recall, accuratezza e F1-Score.
* Random Forest: L'algoritmo Random Forest ha ottenuto i migliori risultati con un'accuratezza del 94.8388% e un F1-Score del 95%. La matrice di confusione normalizzata ha mostrato un'alta precisione di riconoscimento dei dati etichettati, con un tasso di falsi riconoscimenti relativamente basso, il più alto essendo l'1.59%.

Implementazione Software:

* Applicazione Web: Implementata utilizzando JavaScript e Python, l'interfaccia permette di rilevare e avvisare l'utente su siti web maligni.
* Estensione del Browser: Sviluppata utilizzando HTML, JavaScript e CSS, l'estensione utilizza il modello Random Forest implementato in Python per rilevare i siti web maligni in tempo reale.

L'articolo dimostra che l'uso di metodi di machine learning per la classificazione di siti web maligni basati sulle caratteristiche degli URL è efficace. Il metodo Random Forest ha mostrato le migliori performance. Il software sviluppato, sia come applicazione web che come estensione del browser, offre una soluzione pratica per rilevare e avvisare gli utenti di siti web maligni.

Nell’articolo “Development of a malicious website detection system using web browser extension” si discute lo sviluppo di un sistema per la rilevazione di siti web dannosi attraverso un'estensione del browser web. Il sistema proposto combina blacklist e tecniche di machine learning tramite estrazione di caratteristiche per rendere il software ancora più preciso e robusto nel riconoscimento di URL malevoli.

Il sistema è composto da tre livelli principali:

1. User Layer: Interfaccia che permette agli utenti di interagire con il browser e agli amministratori di aggiornare il database delle blacklist.
2. Web Browser Extension Layer: Nucleo del sistema che esegue la previsione della pericolosità del sito web utilizzando due tecniche:
   * Blacklist Feature: Controlla se l'URL visitato è presente nel database delle blacklist.
   * Feature Extractor: Estrae caratteristiche dall'HTML, JavaScript e contenuti lessicali del sito web per determinare se è dannoso.
3. Database Layer: Contiene informazioni di login degli amministratori e la lista delle blacklist.

Metodologie e Implementazione

Il sistema utilizza JavaScript per l'estensione del browser e MySQL per il database. La rilevazione avviene in due fasi:

* Controllo della Blacklist: Se l'URL è presente nella blacklist, viene segnalato come dannoso.
* Estrazione delle Caratteristiche: Se l'URL non è presente nella blacklist, il sistema analizza le caratteristiche del sito (HTML, JavaScript, contenuti lessicali) per determinarne la pericolosità.

L'articolo fornisce una soluzione pratica ed efficace per la rilevazione dei siti web dannosi. La combinazione di blacklist e tecniche di estrazione di caratteristiche garantisce una protezione robusta e in tempo reale contro le minacce web. Questo approccio è particolarmente utile in un contesto in cui le minacce evolvono rapidamente e i metodi tradizionali non sono sufficienti. La facile implementazione dell'estensione del browser e la sua capacità di aggiornarsi automaticamente tramite il database delle blacklist rendono questa soluzione praticabile e scalabile per un'ampia gamma di utenti e contesti di utilizzo.

L’articolo “Phishing Detection and Prevention using Chrome Extension” tratta lo sviluppo di un'estensione Chrome per la rilevazione e la prevenzione del phishing. Durante la pandemia di COVID-19, gli attacchi di phishing sono aumentati significativamente, evidenziando la necessità di soluzioni tecniche più efficaci. L'approccio proposto utilizza algoritmi di machine learning, in particolare il Support Vector Machine (SVM), per migliorare l'accuratezza e l'adattabilità nel rilevare nuove minacce di phishing.

L'estrazione delle caratteristiche degli URL è fondamentale per determinare se un URL è dannoso. Il dataset utilizzato contiene 30 caratteristiche, ridotte a 16 per l'implementazione nel browser. Queste caratteristiche includono:

* isIPInURL: Identifica la presenza di un indirizzo IP nell'URL.
* isLongURL: Controlla se l'URL è più lungo di 75 caratteri.
* isTinyURL: Controlla se l'URL è più corto di 20 caratteri.
* isAlphaNumericURL: Cerca caratteri alfanumerici nell'URL.
* isRedirectingURL: Verifica la presenza di più "//" nell'URL.
* isHypenURL: Cerca "-" adiacenti al nome di dominio.
* isMultiDomainURL: Limita il nome di dominio al livello superiore e al codice del paese.
* isFaviconDomainUnidentical: Controlla se i link sulla pagina web provengono da un altro dominio.
* isIllegalHttpsURL: Identifica la presenza multipla di 'https' in una stringa URL.
* isImgFromDifferentDomain: Verifica se le immagini sono caricate da altri domini.
* isAnchorFromDifferentDomain: Controlla se i link sulla pagina web provengono da un altro dominio.
* isScLnkFromDifferentDomain: Verifica se gli script sulla pagina web provengono da un altro dominio.
* isFormActionInvalid: Cerca sottomissioni di form invalidi o vuoti.
* isMailToAvailable: Cerca tag anchor che incorporano mailto.
* isStatusBarTampered: Controlla se "onmouseover" controlla la visualizzazione della barra di stato.
* isIframePresent: Identifica le pagine che visualizzano iframes nel DOM.

Le caratteristiche sono estratte e codificate in valori {-1, 0, 1} (Legittimo, Sospetto, Phishing) in tempo reale durante il caricamento della pagina.

L'algoritmo di rilevamento e prevenzione del phishing utilizza tre modelli di machine learning supervisionati:

1. Random Forests
2. Artificial Neural Networks (ANN)
3. Support Vector Machines (SVM)

I modelli sono addestrati utilizzando Python con librerie come Scikit-learn, NumPy e Pandas. SVM è stato selezionato come classificatore appreso per la sua elevata accuratezza e specificità. L'algoritmo embedded nel browser predice la validità dei siti visitati e avverte l'utente in caso di URL dannosi.

Il dataset utilizzato include 11,055 URL, con 6157 casi di phishing e 4898 casi legittimi. I dati di phishing sono stati ottenuti da PhishTank e dal repository di UCI Machine Learning.

L'algoritmo è stato testato in un ambiente di laboratorio controllato utilizzando un'estensione Chrome. Le metriche di performance valutate includono accuratezza, sensibilità e specificità. I risultati mostrano che SVM supera gli altri classificatori con un'accuratezza del 90.05% e una specificità del 93%.

Performance Matrix dei Classificatori

* Accuratezza: SVM (90.05%), Random Forest (89.63%), ANN (89.03%)
* Sensibilità: Random Forest (89%), SVM (86%), ANN (86%)
* Specificità: SVM (93%), ANN (92%), Random Forest (90%)

SVM ha mostrato il tasso più basso di falsi positivi tra i tre classificatori, rendendolo il più efficace per classificare URL di phishing.

L'estensione Chrome è progettata con un'interfaccia utente semplice utilizzando HTML e CSS. Permette agli utenti di scaricare e utilizzare l'estensione per rilevare phishing. In caso di rilevamento, l'estensione blocca il sito e avvisa l'utente, fornendo opzioni per ulteriori dettagli o per ignorare l'avviso.

I risultati dimostrano che l'algoritmo SVM implementato nell'estensione Chrome è efficace nel rilevare URL di phishing con alta accuratezza e bassa incidenza di falsi positivi. L'approccio basato su machine learning migliora significativamente rispetto ai metodi tradizionali, adattandosi rapidamente alle nuove minacce.

Nell’articolo “PhishShield: ML Based-Powered Phishing [Client-Side Protection Against Web Spoofing Attacks]” si discute la progettazione e lo sviluppo di PhishShield, un'estensione per Google Chrome progettata per rilevare e prevenire attacchi di phishing basati su spoofing web. La motivazione alla base di questo lavoro è l'aumento degli attacchi di phishing, soprattutto durante la pandemia di COVID-19, che hanno messo in pericolo la sicurezza e la privacy degli utenti. PhishShield utilizza tecniche di machine learning, in particolare il classificatore Random Forest, per valutare la legittimità degli URL e classificare le pagine di login come sospette o affidabili.

L'architettura di PhishShield è composta da tre moduli principali:

1. Backend: Ospita il classificatore e gestisce il dataset.
2. Frontend: Contiene il codice eseguibile scritto in JavaScript, CSS e HTML.
3. Modulo Statico: Include file essenziali come testdata.json e classifier.json.

Il cuore dell'estensione è il file manifest (manifest.json) che definisce le informazioni cruciali dell'estensione come nome, versione, descrizione e permessi richiesti.

Le caratteristiche degli URL sono cruciali per determinare se un sito è di phishing. Il dataset utilizzato comprende informazioni da quattro fonti principali:

* Mohammad et al.: Caratteristiche efficaci per rilevare attacchi di phishing.
* Jalalian et al.: Una collezione di 90 siti web compromessi.
* PhishTank: Un set di 310 URL presenti nella blacklist.
* Moz.com/top500: Un set di 310 URL legittimi.

Le caratteristiche selezionate sono divise in quattro gruppi:

1. Caratteristiche basate sulla barra degli indirizzi
2. Caratteristiche basate su anomalie
3. Caratteristiche basate su HTML e JavaScript
4. Caratteristiche basate sul dominio

PhishShield utilizza il classificatore Random Forest, scelto per la sua robustezza ed efficienza nel rilevamento degli attacchi di phishing. Il processo di classificazione avviene direttamente nel browser del client, migliorando la privacy e riducendo la latenza.

Il processo di sviluppo di PhishShield comprende:

1. Selezione del Modello: Utilizzo di tecniche di data mining per differenziare tra pagine web legittime e di phishing.
2. Preparazione dei Dati: Raccolta e preparazione di dataset rilevanti per l'estrazione delle caratteristiche.
3. Estrazione delle Caratteristiche: Analisi e selezione delle caratteristiche migliori per il modello.
4. Addestramento del Modello: Addestramento del classificatore Random Forest utilizzando tecniche di machine learning supervisionato.

L'efficacia di PhishShield è stata valutata utilizzando un dataset di 800 URL, divisi equamente tra URL di phishing e legittimi. I risultati mostrano:

* Accuratezza: 98.5%
* Precisione: 98.5%
* Recall: 98.5%

PhishShield ha mostrato un tempo di risposta medio di 62.5 millisecondi durante i test su 40 URL di phishing. Questa latenza è significativamente inferiore rispetto agli strumenti stateful come SpoofCatch, che ha mostrato una latenza media di 512 millisecondi.

Il sistema proposto dimostra un'elevata efficacia nella rilevazione degli attacchi di phishing, con un'accuratezza del 98.5% e tempi di risposta rapidi. La classificazione eseguita direttamente nel browser del client offre vantaggi in termini di privacy e velocità.

L’articolo “BrowSEcure” discute lo sviluppo di BrowSEcure, un'estensione per Chrome progettata per prevenire il furto di informazioni sensibili come credenziali di accesso, dettagli di carte di credito e dati personali da parte di siti web di phishing. BrowSEcure utilizza avvisi di sicurezza per segnalare agli utenti quando stanno navigando su URL potenzialmente dannosi, prevenendo così gli attacchi di phishing.

BrowSEcure utilizza algoritmi di machine learning per migliorare l'accuratezza e l'efficienza del rilevamento dei domini di phishing. Questi algoritmi analizzano varie caratteristiche degli URL, come quelle lessicali, relative al dominio e all'host, per identificare i modelli che distinguono i domini legittimi da quelli di phishing.

L'algoritmo di machine learning utilizzato per rilevare i domini di phishing è il Random Forest, che combina la potenza di più alberi decisionali per fare previsioni accurate. L'algoritmo viene addestrato su un ampio dataset etichettato contenente sia URL legittimi che di phishing. Durante l'addestramento, ogni albero decisionale apprende diversi aspetti dei dati, migliorando la capacità di generalizzazione e riducendo il rischio di overfitting.

* Raccolta dei Dati: Include la raccolta di domini legittimi e di phishing, con informazioni rilevanti come nomi di dominio, indirizzi IP, certificati SSL e contenuti HTML.
* Preprocessing dei Dati: Pulizia e preparazione dei dati per l'analisi, gestendo valori mancanti e rimuovendo duplicati.
* Estrazione delle Caratteristiche: Analisi delle caratteristiche dei nomi di dominio, reputazione degli indirizzi IP, anomalie nei certificati SSL e struttura del contenuto delle pagine web.

Oltre al Random Forest, sono stati utilizzati e confrontati diversi algoritmi di machine learning per la rilevazione dei phishing:

* Decision Tree Classifier: 95% di accuratezza
* Support Vector Machine (SVM): 54% di accuratezza
* K-Nearest Neighbors: 68% di accuratezza
* Logistic Regression: 91% di accuratezza
* Naive Bayes: 90% di accuratezza
* AdaBoostClassifier: 93% di accuratezza
* Hybrid Ensembler: 94% di accuratezza

Il Random Forest ha dimostrato di essere l'algoritmo più efficace, con un'accuratezza del 97%.

L'estensione BrowSEcure per il browser Chrome integra vari algoritmi di machine learning per analizzare gli URL e classificarli come legittimi o sospetti. Utilizza anche database di phishing esistenti per confrontare gli URL in tempo reale. Quando viene rilevato un URL sospetto, l'estensione visualizza indicatori visivi e avvisi per aiutare gli utenti a prendere decisioni informate.

L'algoritmo di BrowSEcure è stato testato su un dataset di 7900 siti di phishing e 5800 siti legittimi. I risultati mostrano che l'algoritmo Random Forest ha ottenuto il miglior rendimento in termini di accuratezza.

L'estensione è progettata per operare in tempo reale, analizzando e classificando gli URL durante la navigazione degli utenti. La latenza del sistema è stata ottimizzata per garantire una rapida risposta agli utenti, fornendo avvisi tempestivi senza rallentare l'esperienza di navigazione.

BrowSEcure si è dimostrato efficace nel rilevare e prevenire attacchi di phishing utilizzando tecniche di machine learning. L'integrazione dell'estensione nel browser fornisce una difesa proattiva contro le minacce di phishing, migliorando la sicurezza online degli utenti. Il sistema continuerà a evolversi, incorporando feedback degli utenti e aggiornamenti regolari per affrontare nuove tecniche di phishing.

Per migliorare ulteriormente BrowSEcure, si prevede di:

* Ampliare il set di caratteristiche utilizzate per il rilevamento.
* Implementare altri classificatori discriminativi come SVM.
* Addestrare il modello su dataset più grandi.
* Integrare meccanismi di feedback degli utenti per migliorare continuamente le capacità di rilevamento.

Questi miglioramenti mirano a rafforzare le capacità di BrowSEcure e offrire agli utenti una protezione robusta contro le minacce di phishing in continua evoluzione.

*Titolo capitolo 1*

### Titolo Sottoparagrafo

At vero eos et accusamus et iusto odio dignissimos ducimus qui blanditiis praesen- tium voluptatum deleniti atque corrupti quos dolores et quas molestias excepturi sint occaecati cupiditate non provident, similique sunt in culpa qui officia deserunt mollitia animi, id est laborum et dolorum fuga. Et harum quidem rerum facilis est et expedita distinctio. Nam libero tempore, cum soluta nobis est eligendi optio cumque nihil impedit quo minus id quod maxime placeat facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet ut et voluptates repudian- dae sint et molestiae non recusandae. Itaque earum rerum hic tenetur a sapiente delectus, ut aut reiciendis voluptatibus maiores alias consequatur aut perferendis doloribus asperiores repellat [[3](#_bookmark17)].

# Capitolo 2

Titolo capitolo 2

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum

## Introduction



Figura 2.1: Immagine del Dipartimento

*Titolo capitolo 2*

Listing 2.1: Esempio di codice

x = 1

if x == 1:

*# indented four spaces*

print("x is 1.")

## Titolo Paragrafo

Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium dolo- remque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt. Neque porro quisquam est, qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit, sed quia non numquam eius modi tempora incidunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim ad minima veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur? Quis autem vel eum iure reprehenderit qui in ea voluptate velit esse quam nihil molestiae consequatur, vel illum qui dolorem eum fugiat quo voluptas nulla pariatur?

### Titolo Sottoparagrafo

At vero eos et accusamus et iusto odio dignissimos ducimus qui blanditiis praesen- tium voluptatum deleniti atque corrupti quos dolores et quas molestias excepturi sint occaecati cupiditate non provident, similique sunt in culpa qui officia deserunt mollitia animi, id est laborum et dolorum fuga. Et harum quidem rerum facilis est et expedita distinctio. Nam libero tempore, cum soluta nobis est eligendi optio cumque nihil impedit quo minus id quod maxime placeat facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet ut et voluptates repudian- dae sint et molestiae non recusandae. Itaque earum rerum hic tenetur a sapiente delectus, ut aut reiciendis voluptatibus maiores alias consequatur aut perferendis doloribus asperiores repellat [[3](#_bookmark17)].

# Capitolo 3

Titolo capitolo 3

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum

## Titolo Paragrafo

Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium dolo- remque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt. Neque porro quisquam est, qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit, sed quia non numquam eius modi tempora incidunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim ad minima veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur? Quis autem vel eum iure reprehenderit qui in ea voluptate velit esse quam nihil molestiae consequatur, vel illum qui dolorem eum fugiat quo voluptas nulla pariatur?

*Titolo capitolo 3*

### Titolo Sottoparagrafo

At vero eos et accusamus et iusto odio dignissimos ducimus qui blanditiis praesen- tium voluptatum deleniti atque corrupti quos dolores et quas molestias excepturi sint occaecati cupiditate non provident, similique sunt in culpa qui officia deserunt mollitia animi, id est laborum et dolorum fuga. Et harum quidem rerum facilis est et expedita distinctio. Nam libero tempore, cum soluta nobis est eligendi optio cumque nihil impedit quo minus id quod maxime placeat facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet ut et voluptates repudian- dae sint et molestiae non recusandae. Itaque earum rerum hic tenetur a sapiente delectus, ut aut reiciendis voluptatibus maiores alias consequatur aut perferendis doloribus asperiores repellat [[1](#_bookmark15)].

# Capitolo 4

Titolo capitolo 4

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum

## Titolo Paragrafo

Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium dolo- remque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt. Neque porro quisquam est, qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit, sed quia non numquam eius modi tempora incidunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim ad minima veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur? Quis autem vel eum iure reprehenderit qui in ea voluptate velit esse quam nihil molestiae consequatur, vel illum qui dolorem eum fugiat quo voluptas nulla pariatur?

*Titolo capitolo 4*

### Titolo Sottoparagrafo

At vero eos et accusamus et iusto odio dignissimos ducimus qui blanditiis praesen- tium voluptatum deleniti atque corrupti quos dolores et quas molestias excepturi sint occaecati cupiditate non provident, similique sunt in culpa qui officia deserunt mollitia animi, id est laborum et dolorum fuga. Et harum quidem rerum facilis est et expedita distinctio. Nam libero tempore, cum soluta nobis est eligendi optio cumque nihil impedit quo minus id quod maxime placeat facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet ut et voluptates repudian- dae sint et molestiae non recusandae. Itaque earum rerum hic tenetur a sapiente delectus, ut aut reiciendis voluptatibus maiores alias consequatur aut perferendis doloribus asperiores repellat [[2](#_bookmark16)].

# Capitolo 5 Conclusioni

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

# Bibliografia

[*1] Domingos, P. (2015). \*The Master Algorithm: How the Quest for the Ultimate Learning Machine Will Remake Our World\*. Basic Books.*

*[2] Tegmark, M. (2017). \*Life 3.0: Being Human in the Age of Artificial Intelligence\*. Knopf.*

*[3] Esteva, A., et al. (2019). "A guide to deep learning in healthcare." \*Nature Medicine\*,25(1),24-29.*

*[4] McKinsey & Company. (2018). "Notes from the AI frontier: Applications and value of deep learning."*

*[5] Hawking, S. (2018). \*Brief Answers to the Big Questions\*. Bantam.*

*[6] Russell, S., & Norvig, P. (2010). \*Artificial Intelligence: A Modern Approach\*. Prentice Hall.*

*[7] Géron, A. (2019). \*Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems\*. O'Reilly Media.*

*[8] Bishop, C. M. (2006). \*Pattern Recognition and Machine Learning\*. Springer.*

*[9] Goodfellow, I., Bengio, Y., & Courville, A. (2016). \*Deep Learning\*. MIT Press.*

*[10] Zhang, C., et al. (2017). "Understanding deep learning requires rethinking generalization." \*arXiv preprint arXiv:1611.03530\*.*

*[11] Marcus, G. (2018). "Deep Learning: A Critical Appraisal." \*arXiv preprint arXiv:1801.00631\*.*

*[12] Rajalingappaa, S. (2018). \*Deep Learning for Computer Vision\*. Packt Publishing.*

*[13] Müller, A. C., & Guido, S. (2016). \*Introduction to Machine Learning with Python: A Guide for Data Scientists\*. O'Reilly Media.*

*[14] Chen, T., & Guestrin, C. (2016). "XGBoost: A Scalable Tree Boosting System." In \*Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining\* (pp. 785-794).*

*[15] Aggarwal, C. C. (2018). \*Neural Networks and Deep Learning: A Textbook\*. Springer.*

*[16] James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). \*An Introduction to Statistical Learning with Applications in R\*. Springer.*

*[17] Kuhn, M., & Johnson, K. (2013). \*Applied Predictive Modeling\*. Springer.*

# Ringraziamenti

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.