

Master Semipresenziale Intelligenza Artificiale nella Programmazione



tech università
tecnologica

Master Semipresenziale Intelligenza Artificiale nella Programmazione

Modalità: Semipresenziale (Online + Tirocinio)

Durata: 12 mesi

Titolo: TECH Università Tecnologica

Crediti: 60 + 4 ECTS

Accesso al sito web: www.techitute.com/informatica/it/master-semipresenziale/master-semipresenziale-intelligenza-artificiale-programmazione

Indice

01

Presentazione

pag. 4

02

Perché iscriversi a questo
Master Semipresenziale?

pag. 8

03

Obiettivi

pag. 12

04

Competenze

pag. 22

05

Direzione del corso

pag. 26

06

Struttura e contenuti

pag. 30

07

Tirocinio

pag. 48

08

Dove posso svolgere il
Tirocinio?

pag. 54

09

Metodologia

pag. 58

10

Titolo

pag. 66

01

Presentazione

Grazie ai continui progressi degli algoritmi di apprendimento automatico e delle reti neurali, gli sviluppatori possono ora automatizzare attività complesse, migliorare l'efficienza del codice e sviluppare applicazioni più intelligenti e adattabili. Secondo dati recenti, l'uso dell'Intelligenza Artificiale (IA) nella programmazione ha ridotto significativamente il tempo di sviluppo del software e migliorato sostanzialmente il rilevamento degli errori e l'ottimizzazione delle risorse. Pertanto, TECH ha creato questo programma completo, diviso in due parti essenziali. La prima sezione è interamente teorica e viene insegnata completamente online, utilizzando l'innovativa metodologia di apprendimento nota come *Relearning*. La seconda parte prevede un tirocinio pratico di 3 settimane presso una società specializzata in servizi di programmazione utilizzando l'Intelligenza Artificiale.



“

Grazie a questo Master Semipresenziale di TECH, acquisirai una conoscenza approfondita dei fondamenti essenziali dell'Intelligenza Artificiale, come l'algoritmia, il data mining e i sistemi intelligenti"

Secondo dati recenti, l'uso dell'Intelligenza Artificiale (IA) nella programmazione ha permesso di automatizzare attività ripetitive e complesse, riducendo i tempi di sviluppo e aumentando la precisione nel rilevamento degli errori. Le aziende leader nel settore tecnologico hanno integrato l'IA nei loro flussi di lavoro di sviluppo software, migliorando l'efficienza e facilitando la creazione di applicazioni più intelligenti e adattabili.

Nasce così questo Master Semipresenziale, che stabilirà le basi dell'Intelligenza Artificiale, fornendo agli informatici una solida base nella teoria e nei concetti chiave. Inoltre, analizzeremo i tipi di dati e la loro gestione efficiente per le applicazioni IA, nonché il ruolo dei dati nell'IA, incluso come possono essere ottimizzati e utilizzati in modo efficace per migliorare i risultati.

Inoltre, i professionisti saranno immersi in tecnologie all'avanguardia come TensorFlow e applicazioni specializzate come l'elaborazione del linguaggio naturale e le reti neurali ricorrenti. In questo modo, saranno preparati ad affrontare sfide complesse in settori come l'informatica bioispirata e il miglioramento della produttività. Inoltre, sarà indagata l'architettura del software per il QA *Testing* e lo sviluppo di applicazioni web e mobile.

Infine, il piano di studi si concentrerà sull'implementazione dell'IA per migliorare la qualità dei test software, un approccio completo che garantirà agli studenti di essere preparati a comprendere in profondità l'Intelligenza Artificiale, applicarla in modo efficace nei progetti del mondo reale.

In questo contesto, TECH ha sviluppato un programma universitario che integra la teoria, completamente online, con un tirocinio pratico di 3 settimane presso le principali aziende del settore. In questo modo, la prima parte del percorso accademico si adatta agli orari di lavoro e personali dello studente, che avrà bisogno solo di un dispositivo elettronico con accesso a Internet. Si basa anche sulla rivoluzionaria metodologia *Relearning*, che enfatizza la revisione dei concetti chiave per facilitare una comprensione profonda e naturale dei contenuti.

Questo **Master Semipresenziale in Intelligenza Artificiale nella Programmazione** possiede il programma più completo e aggiornato del mercato. Le sue caratteristiche principali sono:

- ♦ Sviluppo di oltre 100 casi clinici presentati da professionisti della Intelligenza Artificiale e da professori universitari con una vasta esperienza nella sua applicazione alla Programmazione
- ♦ Contenuti grafici, schematici ed eminentemente pratici che forniscono informazione tecnica riguardo alle discipline essenziali per l'esercizio della professione
- ♦ Lezioni teoriche, domande all'esperto, forum di discussione su argomenti controversi e lavori di riflessione individuale
- ♦ Contenuti disponibili da qualsiasi dispositivo fisso o mobile dotato di connessione a internet
- ♦ Possibilità di svolgere un tirocinio presso una delle migliori aziende del settore



Ti focalizza sulle applicazioni pratiche dell'IA in diversi contesti, come progetti web, app mobili e QA Testing, con il supporto della migliore università digitale del mondo, secondo Forbes: TECH

“

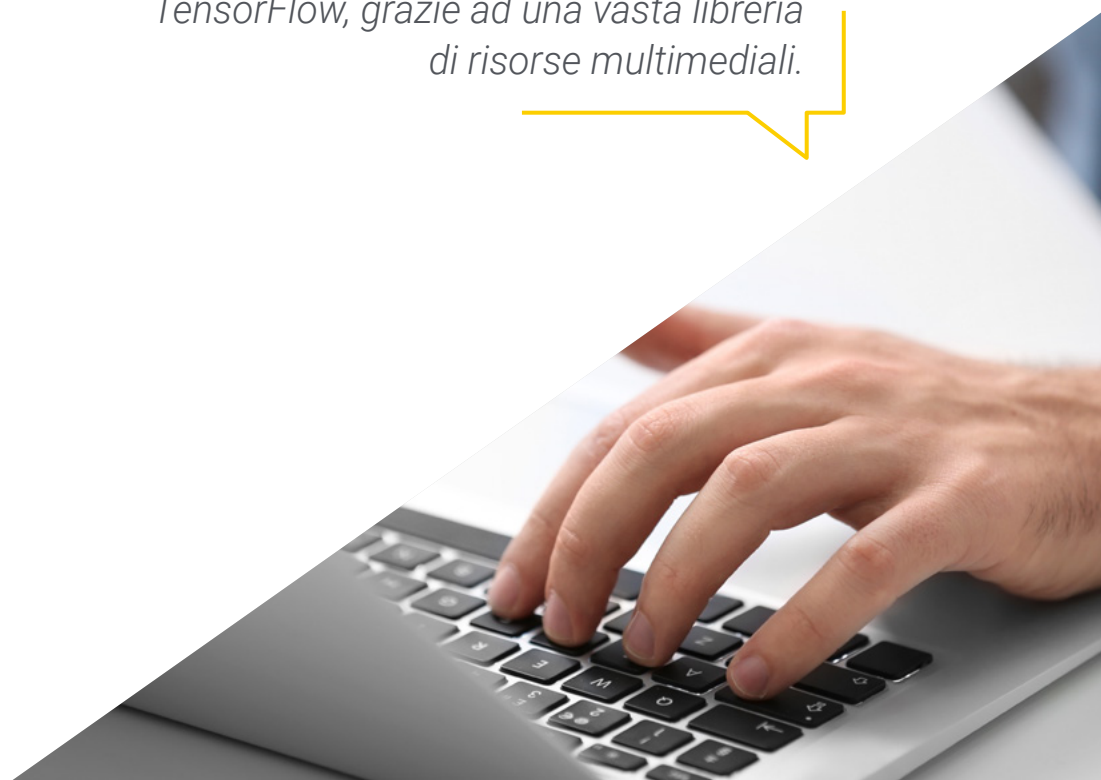
Analizzerai i tipi e il ciclo di vita dei dati, nonché la loro applicazione diretta nell'estrazione di dati, attraverso i migliori materiali didattici, all'avanguardia tecnologica ed educativa"

In questa proposta di Master, di carattere professionalizzante e modalità semipresenziale, il programma è diretto all'aggiornamento dei professionisti dell'Intelligenza Artificiale nella Programmazione, che svolgono le loro funzioni in aziende specializzate in questo campo e che richiedono un alto livello di qualificazione. I contenuti sono basati sulle ultime evidenze scientifiche, orientati in modo didattico per integrare le conoscenze teoriche nella pratica informatica, e gli elementi teorico-pratici faciliteranno l'aggiornamento delle conoscenze e permetteranno di prendere decisioni in situazioni complesse.

I contenuti multimediali, sviluppati in base alle ultime tecnologie educative, forniranno al professionista un apprendimento coinvolgente e localizzato, ovvero inserito in un contesto reale. La creazione di questo programma è incentrata sull'Apprendimento Basato su Problemi, mediante il quale lo specialista deve cercare di risolvere le diverse situazioni che gli si presentano durante il corso. Lo studente potrà usufruire di un innovativo sistema di video interattivi creati da esperti di rinomata fama.

Svolgi un tirocinio di 3 settimane presso un centro prestigioso e acquisisci tutte le conoscenze di cui hai bisogno per crescere personalmente e lavorativamente.

Ti immergerai in argomenti avanzati, come le reti neurali profonde e la loro formazione utilizzando strumenti come TensorFlow, grazie ad una vasta libreria di risorse multimediali.



02

Perché iscriversi a questo Master Semipresenziale?

Questo Master Semipresenziale offre una combinazione unica di flessibilità e profondità accademica che è molto attraente per i professionisti che cercano di avanzare nella loro carriera tecnologica. Questo tipo di programma permetterà agli informatici di beneficiare della comodità dell'apprendimento online, adattandosi ai loro orari lavorativi e personali, per poi effettuare un completo tirocinio pratico di 3 settimane. Inoltre, l'attenzione all'Intelligenza Artificiale e alla Programmazione prepara i professionisti ad affrontare le sfide tecnologiche più impegnative di oggi e del futuro, dotandoli delle competenze necessarie per innovare e guidare in un campo in continua evoluzione.



“

La combinazione di teoria rigorosa e l'esperienza pratica faciliterà lo sviluppo di competenze avanzate e la capacità di guidare progetti innovativi nel campo della tecnologia"

1. Aggiornarsi sulla base delle più recenti tecnologie disponibili

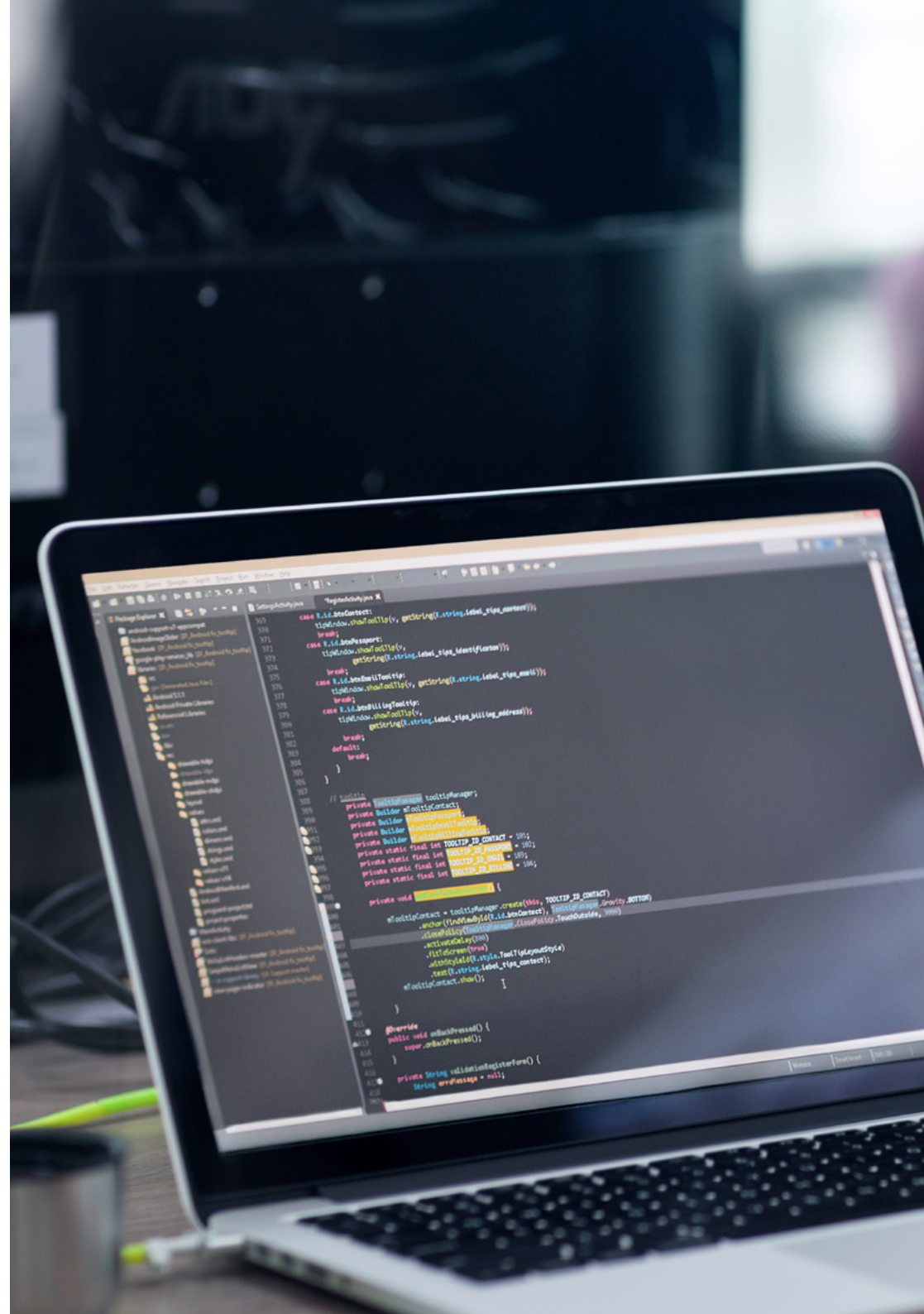
In questo ambito, i sistemi di apprendimento automatico e le reti neurali profonde hanno rivoluzionato la capacità delle macchine di elaborare e comprendere autonomamente i dati. Strumenti come *TensorFlow* e *PyTorch* hanno democratizzato lo sviluppo di modelli complessi, consentendo a ricercatori e sviluppatori di esplorare applicazioni in settori quali l'elaborazione del linguaggio naturale, la visione computerizzata e la generazione di contenuti creativi.

2. Approfondire nuove competenze dall'esperienza dei migliori specialisti

Il team di professionisti che accompagnerà lo specialista per tutto il periodo di pratica rappresenta un'assicurazione di prima classe e una garanzia di aggiornamento senza precedenti. Con un tutor appositamente designato, lo studente sarà in grado di lavorare su progetti reali, in un ambiente all'avanguardia, che gli permetterà di incorporare nella sua pratica quotidiana le procedure e gli strumenti di Programmazione con Intelligenza Artificiale.

3. Accedere ad ambienti professionali di prim'ordine

TECH seleziona con cura tutti i centri disponibili per lo svolgimento del Tirocinio. Grazie a ciò, lo specialista sarà garantito l'accesso ad un centro tecnologico prestigioso nell'ambito dell'Intelligenza Artificiale nella Programmazione. In questo modo, potrete verificare il lavoro quotidiano di un settore esigente, rigoroso ed esaustivo, applicando sempre i progressi tecnologici più innovativi.



4. Combinare la migliore teoria con la pratica più avanzata

Il mercato accademico è pieno di programmi educativi poco adattabili alle attività quotidiane dello specialista e che richiedono lunghe ore di carico didattico, spesso poco compatibili con la vita personale e professionale. TECH offre un nuovo modello di apprendimento, in modalità 100% pratica, che permette di conoscere le procedure più avanzate in materia e, soprattutto, di metterle in pratica in sole 3 settimane.

5. Accedere a nuove opportunità

Con la capacità di sviluppare algoritmi avanzati di apprendimento automatico

e sistemi intelligenti, gli informatici possono influenzare settori diversi come la sanità, l'e-commerce, l'automotive, ecc. Queste tecnologie non solo ottimizzano i processi esistenti, ma consentono anche la creazione di prodotti e servizi innovativi che migliorano la qualità della vita e l'efficienza aziendale.

“

Avrai l'opportunità di svolgere un tirocinio all'interno di un centro a tua scelta”

03 Obiettivi

Questo programma si propone di formare gli informatici a progettare e sviluppare soluzioni innovative, utilizzando strumenti come l'apprendimento automatico, le reti neurali profonde e l'elaborazione del linguaggio naturale. Inoltre, si concentrerà sulla preparazione dei professionisti per applicare queste conoscenze in contesti reali, sia nel settore industriale che nella ricerca, promuovendo competenze critiche come l'analisi dei dati, l'ottimizzazione degli algoritmi e la risoluzione di problemi complessi. Sarà inoltre incoraggiato lo sviluppo di competenze interpersonali e collaborative necessarie per lavorare in team multidisciplinari su progetti tecnologici avanzati.





“

Gli obiettivi principali del Master Semipresenziale sono di fornire agli informatici una conoscenza approfondita e aggiornata delle tecniche e applicazioni più avanzate dell'Intelligenza Artificiale”



Obiettivo generale

- L'obiettivo generale del Master Semipresenziale in Intelligenza Artificiale nella Programmazione è quello di fornire agli informatici competenze avanzate e pratiche fondamentali per padroneggiare ambienti di sviluppo efficienti. Ciò includerà la configurazione e la gestione degli strumenti essenziali per l'implementazione dei progetti di IA, garantendo una base solida per le applicazioni pratiche. Inoltre, il programma cercherà di migliorare la produttività integrando le estensioni di IA in Visual Studio Code e tecniche di progettazione *no-code*, ottimizzando il processo di sviluppo software



Rafforzerai le tue capacità di QA Testing avanzato, inclusa la pianificazione, l'esecuzione e l'automazione dei test di qualità, con il supporto degli strumenti di Intelligenza Artificiale"





Obiettivi specifici

Modulo 1. Fondamenti di Intelligenza Artificiale

- ♦ Analizzare l'evoluzione storica dell'Intelligenza Artificiale, dagli inizi allo stato attuale, identificando le pietre miliari e gli sviluppi principali
- ♦ Comprendere il funzionamento delle reti neurali e la loro applicazione nei modelli di apprendimento dell'Intelligenza Artificiale
- ♦ Studiare i principi e le applicazioni degli algoritmi genetici, analizzando la loro utilità nella risoluzione di problemi complessi
- ♦ Analizzare l'importanza di thesauri, vocabolari e tassonomie nella strutturazione ed elaborazione dei dati per i sistemi di IA
- ♦ Esplorare il concetto di web semantico e la sua influenza sull'organizzazione e la comprensione delle informazioni negli ambienti digitali

Modulo 2. Tipi e cicli di vita del dato

- ♦ Comprendere i concetti fondamentali della statistica e la loro applicazione nell'analisi dei dati
- ♦ Identificare e classificare i diversi tipi di dati statistici, da quelli quantitativi a quelli qualitativi
- ♦ Analizzare il ciclo di vita dei dati, dalla generazione allo smaltimento, identificando le fasi principali
- ♦ Esplorare le fasi iniziali del ciclo di vita dei dati, evidenziando l'importanza della pianificazione e della struttura dei dati
- ♦ Esplorare i processi di raccolta dei dati, compresi la metodologia, gli strumenti e i canali di raccolta
- ♦ Esplorare il concetto di *Datawarehouse* (Magazzino Dati), con particolare attenzione ai suoi elementi costitutivi e alla sua progettazione
- ♦ Analizzare gli aspetti normativi relativi alla gestione dei dati, rispettando le normative sulla privacy e sulla sicurezza, nonché le best practice

Modulo 3. Il dato nell'Intelligenza Artificiale

- ♦ Padroneggiare i fondamenti della scienza dei dati, coprendo gli strumenti, i tipi e le fonti per l'analisi delle informazioni
- ♦ Esplorare il processo di trasformazione dei dati in informazioni utilizzando tecniche di data mining e di visualizzazione dei dati
- ♦ Studiare la struttura e le caratteristiche dei *dataset*, comprendendo la sua importanza nella preparazione e nell'utilizzo dei dati per la modellazione dell'Intelligenza Artificiale
- ♦ Analizzare modelli supervisionati e non supervisionati, compresi i metodi e la classificazione
- ♦ Utilizzare strumenti specifici e best practice nella gestione e nell'elaborazione dei dati, garantendo efficienza e qualità nell'implementazione dell'Intelligenza Artificiale

Modulo 4. Data Mining: Selezione, pre-elaborazione e trasformazione

- ♦ Padroneggiare le tecniche di inferenza statistica per comprendere e applicare i metodi statistici nel data mining
- ♦ Eseguire un'analisi esplorativa dettagliata dei set di dati per identificare modelli, anomalie e tendenze rilevanti
- ♦ Sviluppare competenze per la preparazione dei dati, compresa la pulizia, l'integrazione e la formattazione dei dati per l'utilizzo nel data mining
- ♦ Implementare strategie efficaci per gestire i valori mancanti nei set di dati, applicando metodi di imputazione o rimozione in base al contesto
- ♦ Identificare e ridurre il rumore nei dati, utilizzando tecniche di filtraggio e lisciamiento per migliorare la qualità del set di dati
- ♦ Affrontare la pre-elaborazione dei dati in ambienti *Big Data*

Modulo 5. Algoritmi e complessità nell'Intelligenza Artificiale

- ♦ Introdurre le strategie di progettazione degli algoritmi, fornendo una solida comprensione degli approcci fondamentali alla risoluzione dei problemi
- ♦ Analizzare l'efficienza e la complessità degli algoritmi, applicando tecniche di analisi per valutare le prestazioni in termini di tempo e spazio
- ♦ Studiare e applicare algoritmi di ordinamento, comprendendo le loro prestazioni e confrontando la loro efficienza in contesti diversi
- ♦ Esplorare gli algoritmi ad albero, comprendendo la loro struttura e le loro applicazioni
- ♦ Esaminare gli algoritmi con *Heaps*, analizzandone l'implementazione e l'utilità per una gestione efficiente dei dati
- ♦ Analizzare algoritmi basati su grafi, esplorando la loro applicazione nella rappresentazione e nella soluzione di problemi che coinvolgono relazioni complesse
- ♦ Studiare gli algoritmi *Greedy*, comprendendo la sua logica e le sue applicazioni nella risoluzione di problemi di ottimizzazione
- ♦ Ricercare e applicare la tecnica di *backtracking* per la risoluzione sistematica dei problemi, analizzando la loro efficacia in una varietà di contesti

Modulo 6. Sistemi intelligenti

- ♦ Esplorare la teoria degli agenti, comprendendo i concetti fondamentali del suo funzionamento e la sua applicazione nell'Intelligenza Artificiale e nell'ingegneria del Software
- ♦ Studiare la rappresentazione della conoscenza, compresa l'analisi delle ontologie e la loro applicazione nell'organizzazione delle informazioni strutturate
- ♦ Analizzare il concetto di web semantico e il suo impatto sull'organizzazione e sul reperimento delle informazioni negli ambienti digitali

- ♦ Valutare e confrontare diverse rappresentazioni della conoscenza, integrandole per migliorare l'efficienza e la precisione dei sistemi intelligenti
- ♦ Studiare i ragionatori semantici, i sistemi basati sulla conoscenza e i sistemi esperti, comprendendone le funzionalità e le applicazioni nel processo decisionale intelligente

Modulo 7. Apprendimento automatico e data mining

- ♦ Introdurre i processi di scoperta della conoscenza e i concetti fondamentali dell'apprendimento automatico
- ♦ Studiare gli alberi decisionali come modelli di apprendimento supervisionato, comprendendone la struttura e le applicazioni
- ♦ Valutare i classificatori utilizzando tecniche specifiche per misurarne le prestazioni e l'accuratezza nella classificazione dei dati
- ♦ Studiare le reti neurali, comprendendone il funzionamento e l'architettura per risolvere problemi complessi di apprendimento automatico
- ♦ Esplorare i metodi bayesiani e la loro applicazione nell'apprendimento automatico, comprese le reti e i classificatori bayesiani
- ♦ Analizzare modelli di regressione e di risposta continua per la previsione di valori numerici dai dati
- ♦ Studiare tecniche di *clustering* per identificare schemi e strutture in insiemi di dati non etichettati
- ♦ Esplorare il data mining e l'elaborazione del linguaggio naturale (NLP), comprendendo come le tecniche di apprendimento automatico vengono applicate per analizzare e comprendere il testo

Modulo 8. Le reti neurali, base del *Deep Learning*

- ♦ Padroneggiare i fondamenti e comprendere il ruolo fondamentale del *Deep Learning*
- ♦ Esplorare le operazioni fondamentali delle reti neurali e comprendere la loro applicazione nella costruzione di modelli
- ♦ Analizzare i diversi strati utilizzati nelle reti neurali e imparare a selezionarli in modo appropriato
- ♦ Comprendere l'efficace collegamento di strati e operazioni per progettare architetture di reti neurali complesse ed efficienti
- ♦ Utilizzare trainer e ottimizzatori per mettere a punto e migliorare le prestazioni delle reti neurali
- ♦ Esplorare la connessione tra neuroni biologici e artificiali per una comprensione più approfondita della progettazione dei modelli
- ♦ Impostare gli iperparametri per il *Fine Tuning* delle reti neurali, ottimizzando le loro prestazioni su compiti specifici

Modulo 9. Addestramento delle reti neurali profonde

- ♦ Risolvere i problemi legati ai gradienti nell'addestramento delle reti neurali profonde
- ♦ Esplorare e applicare diversi ottimizzatori per migliorare l'efficienza e la convergenza dei modelli
- ♦ Programmare il tasso di apprendimento per regolare dinamicamente il tasso di convergenza del modello
- ♦ Comprendere e affrontare l'overfitting attraverso strategie specifiche durante l'addestramento
- ♦ Applicare linee guida pratiche per garantire un addestramento efficiente ed efficace delle reti neurali profonde
- ♦ Implementare *Transfer Learning* come tecnica avanzata per migliorare le prestazioni del modello su compiti specifici

- ♦ Esplorare e applicare le tecniche di *Data Augmentation* per arricchire i dataset e migliorare la generalizzazione del modello
- ♦ Sviluppare applicazioni pratiche utilizzando il *Transfer Learning* per risolvere i problemi del mondo reale
- ♦ Comprendere e applicare le tecniche di regolarizzazione per migliorare la generalizzazione ed evitare l'overfitting nelle reti neurali profonde

Modulo 10. Personalizzazione del modello e addestramento con *TensorFlow*

- ♦ Padroneggiare le basi di *TensorFlow* e la sua integrazione con NumPy per una gestione efficiente dei dati e dei calcoli
- ♦ Personalizzare i modelli e gli algoritmi di formazione utilizzando le funzionalità avanzate di *TensorFlow*
- ♦ Esplorare l'API *tf.data* per gestire e manipolare efficacemente gli insiemi di dati
- ♦ Implementare il formato *TFRecord* per la memorizzazione e l'accesso a grandi insiemi di dati in *TensorFlow*
- ♦ Utilizzare i livelli di pre-elaborazione di Keras per facilitare la costruzione di modelli personalizzati
- ♦ Esplorare il progetto *TensorFlow Datasets* per accedere a insiemi di dati predefiniti e migliorare l'efficienza dello sviluppo
- ♦ Sviluppare un'applicazione di *Deep Learning* con *TensorFlow*, integrando le conoscenze acquisite nel modulo
- ♦ Applicare in modo pratico tutti i concetti appresi nella costruzione e nell'addestramento di modelli personalizzati usando *TensorFlow* in situazioni reali

Modulo 11. Deep Computer Vision con Reti Neurali Convoluzionali

- ◆ Comprendere l'architettura della corteccia visiva e la sua importanza nella *Deep Computer Vision*
- ◆ Esplorare e applicare i livelli convoluzionali per estrarre caratteristiche chiave dalle immagini
- ◆ Implementare i livelli di clustering e il loro utilizzo nei modelli di *Deep Computer Vision* con Keras
- ◆ Analizzare varie architetture di reti neurali convoluzionali (CNN) e la loro applicabilità in diversi contesti
- ◆ Sviluppare e implementare una CNN ResNet utilizzando la libreria Keras per migliorare l'efficienza e le prestazioni del modello
- ◆ Utilizzare modelli Keras pre-addestrati per sfruttare l'apprendimento per trasferimento per compiti specifici
- ◆ Applicare tecniche di classificazione e localizzazione in ambienti di *Deep Computer Vision*
- ◆ Esplorare le strategie di rilevamento e tracciamento degli oggetti utilizzando le Reti Neurali Convoluzionali
- ◆ Implementare tecniche di segmentazione semantica per comprendere e classificare in modo dettagliato gli oggetti nelle immagini

Modulo 12. Elaborazione del Linguaggio Naturale (NLP) con Reti Neurali Ricorrenti (RNN) e Assistenza

- ◆ Sviluppare competenze nella generazione di testi utilizzando Reti Neurali Ricorrenti (RNN)
- ◆ Applicare le RNN nella classificazione delle opinioni per l'analisi del sentiment nei testi
- ◆ Comprendere e applicare i meccanismi di attenzione nei modelli di elaborazione del linguaggio naturale
- ◆ Analizzare e utilizzare i modelli *Transformers* in attività specifiche di NLP

- ◆ Esplorare l'applicazione dei modelli *Transformers* nel contesto dell'elaborazione delle immagini e della visione artificiale
- ◆ Acquisire familiarità con la libreria *Transformers* di *Hugging Face* per l'implementazione efficiente di modelli avanzati.
- ◆ Confrontare diverse librerie di *Transformers* per valutare la loro idoneità a specifiche attività
- ◆ Sviluppare un'applicazione pratica di NLP che integri RNN e meccanismi di attenzione per risolvere problemi del mondo reale

Modulo 13. Autoencoder, GAN, e modelli di diffusione

- ◆ Sviluppare rappresentazioni efficienti dei dati utilizzando *Autoencoder*, *GAN* e Modelli di Diffusione
- ◆ Eseguire la PCA utilizzando un codificatore automatico lineare incompleto per ottimizzare la rappresentazione dei dati
- ◆ Implementare e comprendere il funzionamento degli autoencoder impilati
- ◆ Esplorare e applicare gli autoencoder convoluzionali per un'efficiente rappresentazione visiva dei dati
- ◆ Analizzare e applicare l'efficacia degli autoencoder sparsi nella rappresentazione dei dati
- ◆ Generare immagini di moda dal set di dati MNIST utilizzando *Autoencoder*
- ◆ Comprendere il concetto di Reti Generative Avversarie (*GAN*) e Modelli di Diffusione
- ◆ Implementare e confrontare le prestazioni dei Modelli di Diffusione e *GAN* nella generazione di dati

Modulo 14. Computazione bio-ispirata

- ♦ Introdurre i concetti fondamentali della computazione bio-ispirata
- ♦ Esplorare gli algoritmi di adattamento sociale come approccio chiave nella computazione bio-ispirata
- ♦ Analizzare le strategie di esplorazione e sfruttamento dello spazio negli algoritmi genetici
- ♦ Esaminare modelli di calcolo evolutivo nel contesto dell'ottimizzazione
- ♦ Continuare l'analisi dettagliata dei modelli di calcolo evolutivo
- ♦ Applicare la programmazione evolutiva a problemi specifici di apprendimento
- ♦ Affrontare la complessità dei problemi multi-obiettivo nell'ambito della computazione bio-ispirata
- ♦ Esplorare l'applicazione delle reti neurali nel campo della computazione bio-ispirata
- ♦ Approfondire l'implementazione e l'utilità delle reti neurali nell'ambito della computazione bio-ispirata

Modulo 15. Intelligenza Artificiale: Strategie e applicazioni

- ♦ Sviluppare strategie per l'implementazione dell'intelligenza artificiale nei servizi finanziari
- ♦ Analizzare le implicazioni dell'intelligenza artificiale nella fornitura di servizi sanitari
- ♦ Identificare e valutare i rischi associati all'uso dell'IA nel settore sanitario
- ♦ Valutare i rischi potenziali associati all'uso dell'IA nell'industria
- ♦ Applicare le tecniche di intelligenza artificiale nell'industria per migliorare la produttività
- ♦ Progettare soluzioni di intelligenza artificiale per ottimizzare i processi nella pubblica amministrazione
- ♦ Valutare l'implementazione delle tecnologie di IA nel settore dell'istruzione
- ♦ Applicare tecniche di intelligenza artificiale nel settore forestale e agricolo per migliorare la produttività
- ♦ Ottimizzare i processi delle risorse umane attraverso l'uso strategico dell'intelligenza artificiale

Modulo 16. Migliorare la produttività nello sviluppo del software con la IA

- ♦ Approfondisci l'implementazione di estensioni di IA indispensabili in Visual Studio Code per migliorare la produttività e agevolare lo sviluppo di *software*
- ♦ Acquisire una solida comprensione dei concetti di base dell'IA e della sua applicazione nello sviluppo di *software*, compresi gli algoritmi di apprendimento automatico, l'elaborazione del linguaggio naturale, le reti neurali, ecc.
- ♦ Padroneggiare l'impostazione di ambienti di sviluppo ottimizzati, garantendo che gli studenti possano creare ambienti favorevoli ai progetti di IA
- ♦ Applicare tecniche specifiche utilizzando ChatGPT per l'identificazione e la correzione automatica di possibili miglioramenti del codice, incoraggiando pratiche di programmazione più efficienti
- ♦ Promuovere la collaborazione tra diversi professionisti della programmazione (dai programmatori agli ingegneri dei dati ai designer dell'esperienza utente) per sviluppare soluzioni relative a *software* con un'IA efficace ed etica

Modulo 17. Architettura del software con IA

- ♦ Sviluppare le capacità di progettazione di piani di test robusti, che coprano diversi tipi di *testing* e garantire la qualità del *software*
- ♦ Riconoscere e analizzare i diversi tipi di struttura del *software*, come monolitico, microservizi o orientati ai servizi
- ♦ Acquisire una visione completa dei principi e delle tecniche di progettazione di sistemi informatici scalabili e in grado di gestire grandi volumi di dati
- ♦ Applicare conoscenze avanzate nell'implementazione di strutture di dati basate sull'intelligenza artificiale per ottimizzare le prestazioni e l'efficienza del *software*
- ♦ Sviluppare pratiche di sviluppo sicure, concentrandosi sulla prevenzione delle vulnerabilità per garantire la sicurezza del *software* a livello architettonico

Modulo 18. Progetti web con IA

- ♦ Sviluppare competenze complete per l'implementazione di progetti web, a partire dalla progettazione del *frontend* all'ottimizzazione del *backend*, con l'inclusione di elementi di IA
- ♦ Ottimizzare il processo di implementazione dei siti web, incorporando tecniche e strumenti per migliorare la velocità e l'efficienza
- ♦ Integrare l'IA nel cloud computing, consentendo agli studenti di creare progetti web altamente scalabili ed efficienti
- ♦ Acquisire la capacità di identificare problemi e opportunità specifiche nei progetti web in cui l'IA può essere applicata efficacemente, ad esempio nell'elaborazione dei testi, nella personalizzazione, nella raccomandazione di contenuti, ecc.
- ♦ Incoraggiare gli studenti a tenersi aggiornati sulle ultime tendenze e sugli sviluppi dell'IA per la sua corretta applicazione nei progetti web

Modulo 19. Applicazioni mobili con l'IA

- ♦ Applicare concetti avanzati di *clean architecture*, *datasources* e *repositories* per garantire una struttura robusta e modulare nelle applicazioni mobili con l'IA
- ♦ Sviluppare le competenze per progettare schermate interattive, icone e risorse grafiche utilizzando l'IA per migliorare l'esperienza dell'utente nelle applicazioni mobili
- ♦ Approfondire la configurazione dell'ambiente di lavoro per le applicazioni mobili e utilizzare *GitHub Copilot* per ottimizzare il processo di sviluppo
- ♦ Ottimizzare le applicazioni mobili di IA per ottenere prestazioni efficienti, tenendo conto della gestione delle risorse e dell'utilizzo dei dati
- ♦ Eseguire test di qualità sulle applicazioni mobili di IA, consentendo agli studenti di identificare i problemi e di eseguire il debug dei bug





Modulo 20. IA per QA Testing

- ◆ Padroneggiare principi e delle tecniche di progettazione di sistemi informatici scalabili e in grado di gestire grandi volumi di dati
- ◆ Applicare conoscenze avanzate nell'implementazione di strutture di dati basate sull'intelligenza artificiale per ottimizzare le prestazioni e l'efficienza del *software*
- ◆ Comprendere e applicare pratiche di sviluppo sicure, concentrandosi sulla prevenzione delle vulnerabilità dell'iniezione per garantire la sicurezza del *software* a livello architettonico
- ◆ Generare test automatizzati, soprattutto in ambienti web e mobile, integrando strumenti di AI per migliorare l'efficienza del processo
- ◆ Utilizzare strumenti avanzati di QA alimentati dall'IA per un'individuazione più efficiente di *bug* e un miglioramento continuo del *software*

“

Approfondisci la teoria più rilevante in questo campo, applicandola successivamente in un ambiente di lavoro reale”

04

Competenze

Le competenze acquisite in questo programma universitario copriranno una vasta gamma di abilità tecniche e pratiche, che prepareranno gli informatici ad affrontare sfide complesse nello sviluppo del software. In questo modo, svilupperanno la capacità di configurare e gestire ambienti di sviluppo efficienti, integrando estensioni di Intelligenza Artificiale in strumenti come Visual Studio Code, per aumentare la produttività. Utilizzare ChatGPT per ottimizzare e migliorare la qualità del codice, applicando pratiche di programmazione avanzate. Nell'ambito del QA *Testing*, acquisiranno competenze nella pianificazione, esecuzione e automazione dei test di qualità.

```
...ings.js  
...js  
JS command.js  
JS editor.js  
JS fileManager.js  
JS main.js  
... readme.rst  
JS sequences.js
```

```
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65  
66
```

```
self  
input.on  
//escap  
if (e.  
// th  
self.  
return  
}  
if (e.  
e.sto  
e.pro  
self.  
self.  
return  
}  
//up/d  
if (e.  
e.pro  
e.sto  
if (c  
//  
if
```

```
searchHistory;  
= this;  
("keydown", function(e) {  
  if (e.keyCode == 27) {  
    // this is a new line  
    .deactivate(true);  
    return; // this is a modified line  
  }  
})  
("keydown", function(e) {  
  if (e.keyCode == 13) {  
    e.stopPropagation();  
    e.preventDefault();  
    .search();  
    .deactivate();  
    return;  
  }  
})  
("keydown", function(e) {  
  if (e.keyCode == 38 || e.keyCode == 40) {  
    e.preventDefault();  
    e.stopPropagation();  
    .show();  
  }  
})
```

“

Padroneggerai lo sviluppo web completo e la creazione di applicazioni mobili potenziate dall'IA, garantendo prestazioni, scalabilità e sicurezza dei sistemi"



Competenze generali

- ♦ Applicare le estensioni di IA in Visual Studio Code e le tecniche di progettazione *no-code* per aumentare l'efficienza nello sviluppo del software
- ♦ Utilizzare ChatGPT per ottimizzare e migliorare la qualità del codice, applicando pratiche di programmazione avanzate
- ♦ Implementare progetti web, dalla creazione di *workspaces* all'implementazione, integrando la IA, sia nel *frontend* che nel *backend*
- ♦ Sviluppare applicazioni mobili basate sull'IA, dalla configurazione dell'ambiente alla creazione di funzioni avanzate e alla gestione delle risorse grafiche
- ♦ Applicare concetti di archiviazione avanzati e strutture di dati basate sull'intelligenza artificiale per migliorare l'efficienza e la scalabilità dei sistemi
- ♦ Includere pratiche di sviluppo sicure, evitando vulnerabilità come l'iniezione, per garantire l'integrità e la sicurezza del software sviluppato

“

Applicherai standard di progettazione e pratiche di sviluppo sicuro, garantendo l'integrità e la protezione del software sviluppato. Cosa aspetti ad iscriverti?"





Competenze specifiche

- ♦ Applicare tecniche e strategie di IA per migliorare l'efficienza del settore *retail*
- ♦ Implementare tecniche di denoising utilizzando codificatori automatici
- ♦ Creare efficacemente set di dati di addestramento per compiti di elaborazione del linguaggio naturale (NLP)
- ♦ Eseguire i livelli di clustering e il loro utilizzo nei modelli di *Deep Computer Vision* con Keras
- ♦ Utilizzare le funzioni e i grafici di *TensorFlow* per ottimizzare le prestazioni dei modelli personalizzati
- ♦ Ottimizzare lo sviluppo e l'applicazione di *chatbot* e assistenti virtuali, comprendendo il loro funzionamento e le loro potenziali applicazioni
- ♦ Padroneggiare il riutilizzo di strati pre-addestrati per ottimizzare e accelerare il processo di addestramento
- ♦ Costruire la prima rete neurale, applicando i concetti appresi nella pratica
- ♦ Attivare il perceptrone multistrato (MLP) utilizzando la libreria Keras
- ♦ Applicare tecniche di esplorazione e pre-elaborazione dei dati, identificando e preparando i dati per un uso efficace nei modelli di apprendimento automatico
- ♦ Studiare linguaggi e software per la creazione di ontologie, utilizzando strumenti specifici per lo sviluppo di modelli semantici
- ♦ Sviluppare tecniche di pulizia dei dati per garantire la qualità e l'accuratezza delle informazioni utilizzate nelle analisi successive
- ♦ Padroneggiare l'impostazione di ambienti di sviluppo ottimizzati, garantendo che gli studenti possano creare ambienti favorevoli ai progetti di IA
- ♦ Applicare tecniche specifiche utilizzando ChatGPT per l'identificazione e la correzione automatica di possibili miglioramenti del codice, incoraggiando pratiche di programmazione più efficienti
- ♦ Creare test automatizzati, soprattutto in ambienti web e mobile, integrando strumenti di IA per migliorare l'efficienza del processo
- ♦ Utilizzare strumenti avanzati di QA alimentati dall'IA per un'individuazione più efficiente di bugs e un miglioramento continuo del software
- ♦ Integrare l'IA nel cloud computing, consentendo secondo i di creare progetti web altamente scalabili ed efficienti
- ♦ Approfondire la configurazione dell'ambiente di lavoro per le applicazioni mobili e Utilizzare Github Copilot per ottimizzare il processo di sviluppo

05

Direzione del corso

Gli insegnanti di questo Master Semipresenziale in Intelligenza Artificiale nella Programmazione sono esperti altamente qualificati e riconosciuti nel campo dell'IA e dello sviluppo di software. Provenienti da aziende leader nel settore tecnologico, questi professionisti combinano una solida formazione accademica con una vasta esperienza pratica in progetti innovativi e di grande impatto. Inoltre, il suo approccio pedagogico si basa sull'applicazione di metodologie didattiche avanzate, compresa l'integrazione di strumenti e tecnologie emergenti, per fornire un'esperienza di apprendimento arricchente e aggiornata.



“

Gli insegnanti offriranno una conoscenza approfondita e pratica sull'apprendimento automatico, l'ottimizzazione del codice, lo sviluppo sicuro e l'implementazione dell'IA in applicazioni web e mobili"

Direzione



Dott. Peralta Martín-Palomino, Arturo

- CEO e CTO presso Prometheus Global Solutions
- CTO presso Korporate Technologies
- CTO presso AI Shephers GmbH
- Consulente e Assessore Aziendale Strategico presso Alliance Medical
- Direttore di Design e Sviluppo presso DocPath
- Dottorato in Ingegneria Informatica presso l'Università di Castiglia-La Mancha
- Dottorato in Economia Aziendale e Finanza conseguito presso l'Università Camilo José Cela
- Dottorato in Psicologia presso l'Università di Castiglia-La Mancha
- Master in Executive MBA presso l'Università Isabel I
- Master in Direzione Commerciale e Marketing presso l'Università Isabel I
- Master in Big Data presso la Formación Hadoop
- Master in Tecnologie Informatiche Avanzate conseguito presso l'Università di Castiglia-La Mancha
- Membro di: Gruppo di Ricerca SMILE



Dott. Castellanos Herreros, Ricardo

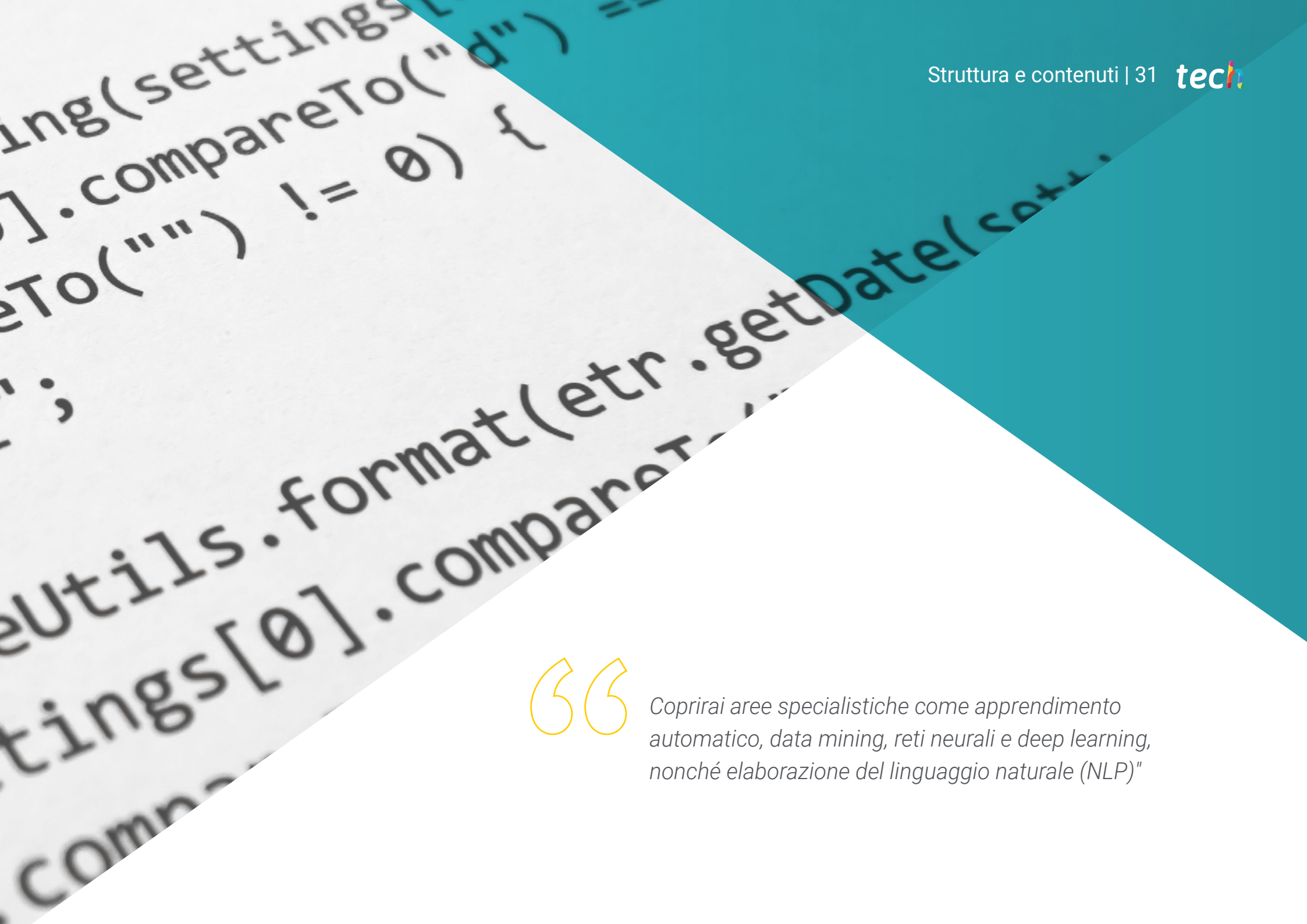
- *Chief Technology Officer* presso OWQLO
- Specialista in Ingegneria Informatica dei Sistemi e *Machine Learning Engineer*
- Consulente Tecnico *Freelance*
- Sviluppatore di applicazioni mobili per eDreams, Fnac, Air Europa, Bankia, Cetelem, Banco Santander, Santillana, Groupón e Grupo Planeta
- Sviluppatore web presso Openbank e Banco Santander
- Ingegnere Tecnico in Informatica dei Sistemi presso l'Università di Castiglia-La Mancha

06

Struttura e contenuti

Il titolo accademico offrirà un piano di studi completo e aggiornato, che affronterà sia le basi teoriche che le applicazioni pratiche dell'Intelligenza Artificiale. Tra i contenuti del programma, sono stati inclusi moduli essenziali come "Fondamenti di Intelligenza Artificiale" e "Tipi e ciclo di vita dei dati", che getteranno le basi per comprendere la gestione e l'elaborazione di grandi volumi di informazioni. Altri moduli in evidenza sono "Data mining: Selezione, pre-elaborazione e trasformazione", "Algoritmi e complessità nell'Intelligenza Artificiale" e "Sistemi intelligenti", che approfondiranno tecniche avanzate e algoritmi cruciali per lo sviluppo dell'IA.





“

Coprirai aree specialistiche come apprendimento automatico, data mining, reti neurali e deep learning, nonché elaborazione del linguaggio naturale (NLP)”

Modulo 1. Fondamenti di Intelligenza Artificiale

- 1.1. Storia dell'intelligenza artificiale
 - 1.1.1. Quando si è cominciato a parlare di intelligenza artificiale?
 - 1.1.2. Riferimenti nel cinema
 - 1.1.3. Importanza dell'intelligenza artificiale
 - 1.1.4. Tecnologie che favoriscono e supportano l'intelligenza artificiale
- 1.2. Intelligenza artificiale nei giochi
 - 1.2.1. Teoria dei giochi
 - 1.2.2. *Minimax* e potatura Alfa-Beta
 - 1.2.3. Simulazione: Monte Carlo
- 1.3. Reti neurali
 - 1.3.1. Basi biologiche
 - 1.3.2. Modello computazionale
 - 1.3.3. Reti neurali supervisionate e non
 - 1.3.4. Percettrone semplice
 - 1.3.5. Percettrone multistrato
- 1.4. Algoritmi genetici
 - 1.4.1. Storia
 - 1.4.2. Base biologica
 - 1.4.3. Codifica dei problemi
 - 1.4.4. Generazione della popolazione iniziale
 - 1.4.5. Algoritmo principale e operatori genetici
 - 1.4.6. Valutazione degli individui: Fitness
- 1.5. Thesauri, vocabolari, tassonomie
 - 1.5.1. Vocabolari
 - 1.5.2. Tassonomie
 - 1.5.3. Thesauri
 - 1.5.4. Ontologie
 - 1.5.5. Rappresentazione della conoscenza: web semantico
- 1.6. Web semantico
 - 1.6.1. Specifiche: RDF, RDFS e OWL
 - 1.6.2. Inferenza/ragionamento
 - 1.6.3. *Linked Data*

- 1.7. Sistemi esperti e DSS
 - 1.7.1. Sistemi esperti
 - 1.7.2. Sistemi di supporto decisionale
- 1.8. *Chatbot* e Assistenti Virtuali
 - 1.8.1. Tipi di assistenti: assistente vocale e scritto
 - 1.8.2. Parti fondamentali per lo sviluppo di un assistente: *Intent*, entità e flusso di dialogo
 - 1.8.3. Integrazioni: web, *Slack*, Whatsapp, Facebook
 - 1.8.4. Strumenti per lo sviluppo di un assistente: *Dialog Flow*, *Watson Assistant*
- 1.9. Strategia di implementazione dell'IA
- 1.10. Futuro dell'intelligenza artificiale
 - 1.10.1. Comprendere come identificare emozioni tramite algoritmi
 - 1.10.2. Creazione di una personalità: linguaggio, espressioni e contenuto
 - 1.10.3. Tendenze dell'intelligenza artificiale
 - 1.10.4. Riflessioni

Modulo 2. Tipi e cicli di vita del dato

- 2.1. La Statistica
 - 2.1.1. Statistica: statistiche descrittive, inferenze statistiche
 - 2.1.2. Popolazione, campione, individuo
 - 2.1.3. Variabili: definizione, scale di misurazione
- 2.2. Tipi di dati statistici
 - 2.2.1. Secondo la tipologia
 - 2.2.1.1. Quantitativi: dati continui e discreti
 - 2.2.1.2. Qualitativi: dati binominali, nominali e ordinali
 - 2.2.2. Secondo la forma
 - 2.2.2.1. Numerici
 - 2.2.2.2. Testuali
 - 2.2.2.3. Logici
 - 2.2.3. Secondo la fonte
 - 2.2.3.1. Primari
 - 2.2.3.2. Secondari

- 2.3. Ciclo di vita dei dati
 - 2.3.1. Fasi del ciclo
 - 2.3.2. Tappe del ciclo
 - 2.3.3. Principi FAIR
- 2.4. Fasi iniziali del ciclo
 - 2.4.1. Definizione delle mete
 - 2.4.2. Determinazione delle risorse necessarie
 - 2.4.3. Diagramma di Gantt
 - 2.4.4. Struttura dei dati
- 2.5. Raccolta di dati
 - 2.5.1. Metodologia di raccolta
 - 2.5.2. Strumenti di raccolta
 - 2.5.3. Canali di raccolta
- 2.6. Pulizia del dato
 - 2.6.1. Fasi di pulizia dei dati
 - 2.6.2. Qualità del dato
 - 2.6.3. Elaborazione dei dati (con R)
- 2.7. Analisi dei dati, interpretazione e valutazione dei risultati
 - 2.7.1. Misure statistiche
 - 2.7.2. Indici di relazione
 - 2.7.3. Data Mining
- 2.8. Archiviazione dei dati (*Datawarehouse*)
 - 2.8.1. Elementi che lo integrano
 - 2.8.2. Progettazione
 - 2.8.3. Aspetti da considerare
- 2.9. Disponibilità del dato
 - 2.9.1. Accesso
 - 2.9.2. Utilità
 - 2.9.3. Sicurezza
- 2.10. Aspetti normativi
 - 2.10.1. Legge di protezione dei dati
 - 2.10.2. Best practice
 - 2.10.3. Altri aspetti normativi

Modulo 3. Il dato nell'Intelligenza Artificiale

- 3.1. Data Science
 - 3.1.1. Data Science
 - 3.1.2. Strumenti avanzati per i data scientist
- 3.2. Dati, informazioni e conoscenza
 - 3.2.1. Dati, informazioni e conoscenza
 - 3.2.2. Tipi di dati
 - 3.2.3. Fonti di dati
- 3.3. Dai dati all'informazione
 - 3.3.1. Analisi dei dati
 - 3.3.2. Tipi di analisi
 - 3.3.3. Estrazione di informazioni da un *Dataset*
- 3.4. Estrazione di informazioni tramite visualizzazione
 - 3.4.1. La visualizzazione come strumento di analisi
 - 3.4.2. Metodi di visualizzazione
 - 3.4.3. Visualizzazione di un insieme di dati
- 3.5. Qualità dei dati
 - 3.5.1. Dati di qualità
 - 3.5.2. Pulizia di dati
 - 3.5.3. Pre-elaborazione base dei dati
- 3.6. *Dataset*
 - 3.6.1. Arricchimento del *Dataset*
 - 3.6.2. La maledizione della dimensionalità
 - 3.6.3. Modifica di un insieme di dati
- 3.7. Squilibrio
 - 3.7.1. Squilibrio di classe
 - 3.7.2. Tecniche di mitigazione dello squilibrio
 - 3.7.3. Equilibrio di un *Dataset*
- 3.8. Modelli non supervisionati
 - 3.8.1. Modello non supervisionato
 - 3.8.2. Metodi
 - 3.8.3. Classificazione con modelli non supervisionati

- 3.9. Modelli supervisionati
 - 3.9.1. Modello supervisionato
 - 3.9.2. Metodi
 - 3.9.3. Classificazione con modelli supervisionati
- 3.10. Strumenti e best practice
 - 3.10.1. Best practice per i data scientist
 - 3.10.2. Il modello migliore
 - 3.10.3. Strumenti utili

Modulo 4. Data Mining: Selezione, pre-elaborazione e trasformazione

- 4.1. Inferenza statistica
 - 4.1.1. Statistica descrittiva vs Inferenza statistica
 - 4.1.2. Procedure parametriche
 - 4.1.3. Procedure non parametriche
- 4.2. Analisi esplorativa
 - 4.2.1. Analisi descrittiva
 - 4.2.2. Visualizzazione
 - 4.2.3. Preparazione dei dati
- 4.3. Preparazione dei dati
 - 4.3.1. Integrazione e pulizia di dati
 - 4.3.2. Standardizzazione dei dati
 - 4.3.3. Trasformazione degli attributi
- 4.4. I valori mancanti
 - 4.4.1. Trattamenti dei valori mancanti
 - 4.4.2. Metodi di imputazione a massima verosimiglianza
 - 4.4.3. Imputazione di valori mancanti mediante apprendimento automatico
- 4.5. Rumore nei dati
 - 4.5.1. Classi di rumore e attributi
 - 4.5.2. Filtraggio del rumore
 - 4.5.3. Effetto del rumore
- 4.6. La maledizione della dimensionalità
 - 4.6.1. *Oversampling*
 - 4.6.2. *Undersampling*
 - 4.6.3. Riduzione dei dati multidimensionali

- 4.7. Da attributi continui a discreti
 - 4.7.1. Dati continui vs discreti
 - 4.7.2. Processo di discretizzazione
- 4.8. I dati
 - 4.8.1. Selezione dei dati
 - 4.8.2. Prospettiva e criteri di selezione
 - 4.8.3. Metodi di selezione
- 4.9. Selezione di istanze
 - 4.9.1. Metodi per la selezione di istanze
 - 4.9.2. Selezione di prototipi
 - 4.9.3. Metodi avanzati per la selezione di istanze
- 4.10. Pre-elaborazione dei dati negli ambienti Big Data

Modulo 5. Algoritmi e complessità nell'Intelligenza Artificiale

- 5.1. Introduzione ai modelli di progettazione di algoritmi
 - 5.1.1. Risorse
 - 5.1.2. Dividi e conquista
 - 5.1.3. Altre strategie
- 5.2. Efficienza e analisi degli algoritmi
 - 5.2.1. Misure di efficienza
 - 5.2.2. Misurare l'ingresso di input
 - 5.2.3. Misurare il tempo di esecuzione
 - 5.2.4. Caso peggiore, migliore e medio
 - 5.2.5. Notazione asintotica
 - 5.2.6. Criteri di analisi matematica per algoritmi non ricorsivi
 - 5.2.7. Analisi matematica per algoritmi ricorsivi
 - 5.2.8. Analisi empirica degli algoritmi
- 5.3. Algoritmi di ordinamento
 - 5.3.1. Concetto di ordinamento
 - 5.3.2. Ordinamento delle bolle
 - 5.3.3. Ordinamento per selezione
 - 5.3.4. Ordinamento per inserimento
 - 5.3.5. Ordinamento per fusione (*Merge_Sort*)
 - 5.3.6. Ordinamento rapido (*Quick_Sort*)

- 5.4. Algoritmi con alberi
 - 5.4.1. Concetto di albero
 - 5.4.2. Alberi binari
 - 5.4.3. Percorsi degli alberi
 - 5.4.4. Rappresentare le espressioni
 - 5.4.5. Alberi binari ordinati
 - 5.4.6. Alberi binari bilanciati
- 5.5. Algoritmi con *Heaps*
 - 5.5.1. Gli *Heaps*
 - 5.5.2. L'algoritmo *Heapsort*
 - 5.5.3. Code prioritarie
- 5.6. Algoritmi con grafi
 - 5.6.1. Rappresentazione
 - 5.6.2. Percorso in larghezza
 - 5.6.3. Percorso in profondità
 - 5.6.4. Ordinamento topologico
- 5.7. Algoritmi *Greedy*
 - 5.7.1. La strategia *Greedy*
 - 5.7.2. Elementi della strategia *Greedy*
 - 5.7.3. Cambio valuta
 - 5.7.4. Il problema del viaggiatore
 - 5.7.5. Problema dello zaino
- 5.8. Ricerca del percorso minimo
 - 5.8.1. Il problema del percorso minimo
 - 5.8.2. Archi e cicli negativi
 - 5.8.3. Algoritmo di Dijkstra
- 5.9. Algoritmi *Greedy* sui grafi
 - 5.9.1. L'albero a sovrapposizione minima
 - 5.9.2. Algoritmo di Prim
 - 5.9.3. Algoritmo di Kruskal
 - 5.9.4. Analisi della complessità
- 5.10. *Backtracking*
 - 5.10.1. Il *Backtracking*
 - 5.10.2. Tecniche alternative

Modulo 6. Sistemi intelligenti

- 6.1. Teoria degli agenti
 - 6.1.1. Storia del concetto
 - 6.1.2. Definizione di agente
 - 6.1.3. Agenti nell'Intelligenza Artificiale
 - 6.1.4. Agenti nell'Ingegneria dei Software
- 6.2. Architetture di agenti
 - 6.2.1. Il processo di ragionamento dell'agente
 - 6.2.2. Agenti reattivi
 - 6.2.3. Agenti deduttivi
 - 6.2.4. Agenti ibridi
 - 6.2.5. Confronto
- 6.3. Informazione e conoscenza
 - 6.3.1. Distinzione tra dati, informazioni e conoscenza
 - 6.3.2. Valutazione della qualità dei dati
 - 6.3.3. Metodi di raccolta dei dati
 - 6.3.4. Metodi di acquisizione dei dati
 - 6.3.5. Metodi di acquisizione della conoscenza
- 6.4. Rappresentazione della conoscenza
 - 6.4.1. L'importanza della rappresentazione della conoscenza
 - 6.4.2. Definire la rappresentazione della conoscenza attraverso i suoi ruoli
 - 6.4.3. Caratteristiche di una rappresentazione della conoscenza
- 6.5. Ontologie
 - 6.5.1. Introduzione ai metadati
 - 6.5.2. Concetto filosofico di ontologia
 - 6.5.3. Concetto informatico di ontologia
 - 6.5.4. Ontologie di dominio e di livello superiore
 - 6.5.5. Come costruire un'ontologia?

- 6.6. Linguaggi ontologici e software per la creazione di ontologie
 - 6.6.1. Triple RDF, *Turtle* e N
 - 6.6.2. Schema *RDF*
 - 6.6.3. OWL
 - 6.6.4. SPARQL
 - 6.6.5. Introduzione ai diversi strumenti per la creazione di ontologie
 - 6.6.6. Installazione e utilizzo di *Protégé*
- 6.7. Sito web semantico
 - 6.7.1. Lo stato attuale e il futuro del web semantico
 - 6.7.2. Applicazioni del web semantico
- 6.8. Altri modelli di rappresentazione della conoscenza
 - 6.8.1. Vocabolari
 - 6.8.2. Panoramica
 - 6.8.3. Tassonomie
 - 6.8.4. Thesauri
 - 6.8.5. Folksonomie
 - 6.8.6. Confronto
 - 6.8.7. Mappe mentali
- 6.9. Valutazione e integrazione delle rappresentazioni della conoscenza
 - 6.9.1. Logica dell'ordine zero
 - 6.9.2. Logica di prim'ordine
 - 6.9.3. Logica descrittiva
 - 6.9.4. Relazione tra i diversi tipi di logica
 - 6.9.5. *Prolog*: programmazione basata sulla logica del primo ordine
- 6.10. Rationatori semantici, sistemi basati sulla conoscenza e sistemi esperti
 - 6.10.1. Concetto di ragionatore
 - 6.10.2. Applicazioni di un ragionatore
 - 6.10.3. Sistemi basati sulla conoscenza
 - 6.10.4. MYCIN, storia dei sistemi esperti
 - 6.10.5. Elementi e architettura dei sistemi esperti
 - 6.10.6. Creazione di sistemi esperti

Modulo 7. Apprendimento automatico e data mining

- 7.1. Introduzione ai processi di scoperta della conoscenza e ai concetti di base dell'apprendimento automatico
 - 7.1.1. Concetti chiave dei processi di scoperta della conoscenza
 - 7.1.2. Prospettiva storica sui processi di scoperta della conoscenza
 - 7.1.3. Fasi dei processi di scoperta della conoscenza
 - 7.1.4. Tecniche utilizzate nei processi di scoperta della conoscenza
 - 7.1.5. Caratteristiche dei buoni modelli di apprendimento automatico
 - 7.1.6. Tipi di informazioni sull'apprendimento automatico
 - 7.1.7. Concetti di base dell'apprendimento
 - 7.1.8. Concetti di base dell'apprendimento non supervisionato
- 7.2. Analisi e pre-elaborazione dei dati
 - 7.2.1. Elaborazione dei dati
 - 7.2.2. Trattamento dei dati nel flusso di analisi dei dati
 - 7.2.3. Tipi di dati
 - 7.2.4. Trasformazione dei dati
 - 7.2.5. Visualizzazione ed esplorazione di variabili continue
 - 7.2.6. Visualizzazione ed esplorazione di variabili categoriche
 - 7.2.7. Misure di correlazione
 - 7.2.8. Rappresentazioni grafiche più comuni
 - 7.2.9. Introduzione all'analisi multivariata e alla riduzione delle dimensioni
- 7.3. Alberi decisionali
 - 7.3.1. Algoritmo ID
 - 7.3.2. Algoritmo C
 - 7.3.3. Sovrallenamento e potatura
 - 7.3.4. Analisi dei risultati
- 7.4. Valutazione dei classificatori
 - 7.4.1. Matrici di confusione
 - 7.4.2. Matrici di valutazione numerica
 - 7.4.3. Statistica Kappa
 - 7.4.4. La curva ROC

- 7.5. Regole di classificazione
 - 7.5.1. Misure di valutazione delle regole
 - 7.5.2. Introduzione alla rappresentazione grafica
 - 7.5.3. Algoritmo di sovrapposizione sequenziale
- 7.6. Reti neurali
 - 7.6.1. Concetti di base
 - 7.6.2. Reti neurali semplici
 - 7.6.3. Algoritmo di *Backpropagation*
 - 7.6.4. Introduzione alle reti neurali ricorrenti
- 7.7. Metodi bayesiani
 - 7.7.1. Concetti di base della probabilità
 - 7.7.2. Teorema di Bayes
 - 7.7.3. Naive Bayes
 - 7.7.4. Introduzione alle reti bayesiane
- 7.8. Modelli di regressione e di risposta continua
 - 7.8.1. Regressione lineare semplice
 - 7.8.2. Regressione lineare multipla
 - 7.8.3. Regressione logistica
 - 7.8.4. Alberi di regressione
 - 7.8.5. Introduzione alle macchine a vettori di supporto (SVM)
 - 7.8.6. Misure di bontà di adattamento
- 7.9. *Clustering*
 - 7.9.1. Concetti di base
 - 7.9.2. *Clustering* gerarchico
 - 7.9.3. Metodi probabilistici
 - 7.9.4. Algoritmo EM
 - 7.9.5. Metodo *B-Cubed*
 - 7.9.6. Metodi impliciti
- 7.10. Estrazione di testi ed elaborazione del linguaggio naturale (NLP)
 - 7.10.1. Concetti di base
 - 7.10.2. Creazione del corpus
 - 7.10.3. Analisi descrittiva
 - 7.10.4. Introduzione alla sentiment analysis

Modulo 8. Le reti neurali, base del *Deep Learning*

- 8.1. Deep Learning
 - 8.1.1. Tipi di Deep Learning
 - 8.1.2. Applicazioni del Deep Learning
 - 8.1.3. Vantaggi e svantaggi del Deep Learning
- 8.2. Operazioni
 - 8.2.1. Somma
 - 8.2.2. Prodotto
 - 8.2.3. Trasporto
- 8.3. Livelli
 - 8.3.1. Livello di input
 - 8.3.2. Livello nascosto
 - 8.3.3. Livello di output
- 8.4. Unione di livelli e operazioni
 - 8.4.1. Progettazione dell'architettura
 - 8.4.2. Connessione tra i livelli
 - 8.4.3. Propagazione in avanti
- 8.5. Costruzione della prima rete neurale
 - 8.5.1. Progettazione della rete
 - 8.5.2. Impostare i pesi
 - 8.5.3. Addestramento della rete
- 8.6. Trainer e ottimizzatore
 - 8.6.1. Selezione dell'ottimizzatore
 - 8.6.2. Ristabilire una funzione di perdita
 - 8.6.3. Ristabilire una metrica
- 8.7. Applicazione dei Principi delle Reti Neurali
 - 8.7.1. Funzioni di attivazione
 - 8.7.2. Propagazione all'indietro
 - 8.7.3. Regolazioni dei parametri
- 8.8. Dai neuroni biologici a quelli artificiali
 - 8.8.1. Funzionamento di un neurone biologico
 - 8.8.2. Trasferimento della conoscenza ai neuroni artificiali
 - 8.8.3. Stabilire relazioni tra di essi

- 8.9. Implementazione di MLP (Perceptron multistrato) con Keras
 - 8.9.1. Definizione della struttura di reti
 - 8.9.2. Creazione del modello
 - 8.9.3. Addestramento del modello
- 8.10. Iperparametri di *Fine tuning* di Reti Neurali
 - 8.10.1. Selezione della funzione di attivazione
 - 8.10.2. Stabilire il *Learning rate*
 - 8.10.3. Regolazioni dei pesi

Modulo 9. Addestramento delle reti neurali profonde

- 9.1. Problemi di Gradiente
 - 9.1.1. Tecniche di ottimizzazione gradiente
 - 9.1.2. Gradienti Stocastici
 - 9.1.3. Tecniche di inizializzazione del peso
- 9.2. Riutilizzo di strati pre-addestrati
 - 9.2.1. Addestramento del trasferimento della conoscenza
 - 9.2.2. Estrazione delle caratteristiche
 - 9.2.3. Deep Learning
- 9.3. Ottimizzatori
 - 9.3.1. Ottimizzatori a discesa stocastica del gradiente
 - 9.3.2. Ottimizzatori Adam e *RMSprop*
 - 9.3.3. Ottimizzatori di momento
- 9.4. Programmazione del tasso di apprendimento
 - 9.4.1. Controllo del tasso di apprendimento automatico
 - 9.4.2. Cicli di apprendimento
 - 9.4.3. Termini di lisciatura
- 9.5. Overfitting
 - 9.5.1. Convalida incrociata
 - 9.5.2. Regularizzazione
 - 9.5.3. Metriche di valutazione
- 9.6. Linee guida pratiche
 - 9.6.1. Progettazione dei modelli
 - 9.6.2. Selezione delle metriche e dei parametri di valutazione
 - 9.6.3. Verifica delle ipotesi

- 9.7. *Transfer Learning*
 - 9.7.1. Addestramento del trasferimento della conoscenza
 - 9.7.2. Estrazione delle caratteristiche
 - 9.7.3. Deep Learning
- 9.8. *Data Augmentation*
 - 9.8.1. Trasformazioni dell'immagine
 - 9.8.2. Generazione di dati sintetici
 - 9.8.3. Trasformazione del testo
- 9.9. Applicazione Pratica del *Transfer Learning*
 - 9.9.1. Addestramento del trasferimento della conoscenza
 - 9.9.2. Estrazione delle caratteristiche
 - 9.9.3. Deep Learning
- 9.10. Regularizzazione
 - 9.10.1. L e L
 - 9.10.2. Regularizzazione a entropia massima
 - 9.10.3. *Dropout*

Modulo 10. Personalizzazione di Modelli e addestramento con *TensorFlow*

- 10.1. *TensorFlow*
 - 10.1.1. Utilizzo della libreria *TensorFlow*
 - 10.1.2. Addestramento dei modelli con *TensorFlow*
 - 10.1.3. Operazioni grafiche su *TensorFlow*
- 10.2. *TensorFlow* e NumPy
 - 10.2.1. Ambiente computazionale NumPy per *TensorFlow*
 - 10.2.2. Utilizzo degli array NumPy con *TensorFlow*
 - 10.2.3. Operazioni NumPy per i grafici di *TensorFlow*
- 10.3. Personalizzazione di modelli e algoritmi di addestramento
 - 10.3.1. Costruire modelli personalizzati con *TensorFlow*
 - 10.3.2. Gestione dei parametri di addestramento
 - 10.3.3. Utilizzo di tecniche di ottimizzazione per l'addestramento

- 10.4. Funzioni e grafica di *TensorFlow*
 - 10.4.1. Funzioni con *TensorFlow*
 - 10.4.2. Utilizzo di grafici per l'addestramento dei modelli
 - 10.4.3. Ottimizzazione dei grafici con le operazioni di *TensorFlow*
- 10.5. Caricamento e pre-elaborazione dei dati con *TensorFlow*
 - 10.5.1. Caricamento di insiemi di dati con *TensorFlow*
 - 10.5.2. Pre-elaborazione dei dati con *TensorFlow*
 - 10.5.3. Utilizzo di strumenti di *TensorFlow* per la manipolazione dei dati
- 10.6. La API *tfdata*
 - 10.6.1. Utilizzo dell'API *tfdata* per il trattamento dei dati
 - 10.6.2. Costruzione di flussi di dati con *tfdata*
 - 10.6.3. Uso dell'API *tfdata* per l'addestramento dei modelli
- 10.7. Il formato *TFRecord*
 - 10.7.1. Utilizzo dell'API *TFRecord* per la serialità dei dati
 - 10.7.2. Caricamento di file *TFRecord* con *TensorFlow*
 - 10.7.3. Utilizzo di file *TFRecord* per l'addestramento dei modelli
- 10.8. Livelli di pre-elaborazione di Keras
 - 10.8.1. Utilizzo dell'API di pre-elaborazione Keras
 - 10.8.2. Costruzione di *pipeline* di pre-elaborazione con Keras
 - 10.8.3. Uso dell'API nella pre-elaborazione di Keras per l'addestramento dei modelli
- 10.9. Il progetto *TensorFlow Datasets*
 - 10.9.1. Utilizzo di *TensorFlow Datasets* per la serialità dei dati
 - 10.9.2. Pre-elaborazione dei dati con *TensorFlow Dataset*
 - 10.9.3. Uso di *TensorFlow Datasets* per l'addestramento dei modelli
- 10.10. Costruire un'applicazione di Deep Learning con *TensorFlow*
 - 10.10.1. Applicazione Pratica
 - 10.10.2. Costruire un'applicazione di Deep Learning con *TensorFlow*
 - 10.10.3. Addestramento dei modelli con *TensorFlow*
 - 10.10.4. Utilizzo dell'applicazione per la previsione dei risultati

Modulo 11. Deep Computer Vision con Reti Neurali Convoluzionali

- 11.1. L'architettura *Visual Cortex*
 - 11.1.1. Funzioni della corteccia visiva
 - 11.1.2. Teoria della visione computazionale
 - 11.1.3. Modelli di elaborazione delle immagini
- 11.2. Layer convoluzionali
 - 11.2.1. Riutilizzo dei pesi nella convoluzione
 - 11.2.2. Convoluzione D
 - 11.2.3. Funzioni di attivazione
- 11.3. Livelli di raggruppamento e distribuzione dei livelli di raggruppamento con Keras
 - 11.3.1. *Pooling* e *Striding*
 - 11.3.2. *Flattening*
 - 11.3.3. Tipi di *Pooling*
- 11.4. Architetture CNN
 - 11.4.1. Architettura VGG
 - 11.4.2. Architettura *AlexNet*
 - 11.4.3. Architettura *ResNet*
- 11.5. Implementazione di una CNN *ResNet*- usando Keras
 - 11.5.1. Inizializzazione dei pesi
 - 11.5.2. Definizione del livello di input
 - 11.5.3. Definizione di output
- 11.6. Uso di modelli pre-addestramento di Keras
 - 11.6.1. Caratteristiche dei modelli pre-addestramento
 - 11.6.2. Usi dei modelli pre-addestramento
 - 11.6.3. Vantaggi dei modelli pre-addestramento
- 11.7. Modelli pre-addestramento per l'apprendimento tramite trasferimento
 - 11.7.1. L'apprendimento attraverso il trasferimento
 - 11.7.2. Processo di apprendimento per trasferimento
 - 11.7.3. Vantaggi dell'apprendimento per trasferimento
- 11.8. Classificazione e localizzazione in *Deep Computer Vision*
 - 11.8.1. Classificazione di immagini
 - 11.8.2. Localizzazione di oggetti nelle immagini
 - 11.8.3. Rilevamento di oggetti

- 11.9. Rilevamento di oggetti e tracciamento degli oggetti
 - 11.9.1. Metodi di rilevamento degli oggetti
 - 11.9.2. Algoritmi di tracciamento degli oggetti
 - 11.9.3. Tecniche di tracciamento e localizzazione
- 11.10. Segmentazione semantica
 - 11.10.1. Deep Learning con segmentazione semantica
 - 11.10.2. Rilevamento dei bordi
 - 11.10.3. Metodi di segmentazione basati su regole

Modulo 12. Elaborazione del Linguaggio Naturale (NLP) con Reti Neurali Ricorrenti (RNN) e Assistenza

- 12.1. Generazione di testo utilizzando RNN
 - 12.1.1. Addestramento di una RNN per la generazione di testo
 - 12.1.2. Generazione di linguaggio naturale con RNN
 - 12.1.3. Applicazioni di generazione di testo con RNN
- 12.2. Creazione del set di dati di addestramento
 - 12.2.1. Preparazione dei dati per l'addestramento di una RNN
 - 12.2.2. Conservazione del set di dati di addestramento
 - 12.2.3. Pulizia e trasformazione dei dati
 - 12.2.4. Analisi del Sentiment
- 12.3. Classificazione delle opinioni con RNN
 - 12.3.1. Rilevamento degli argomenti nei commenti
 - 12.3.2. Analisi dei sentimenti con algoritmi di deep learning
- 12.4. Rete encoder-decoder per eseguire la traduzione automatica neurale
 - 12.4.1. Addestramento di una RNN per eseguire la traduzione automatica
 - 12.4.2. Utilizzo di una rete *encoder-decoder* per la traduzione automatica
 - 12.4.3. Migliore precisione della traduzione automatica con RNN
- 12.5. Meccanismi di assistenza
 - 12.5.1. Attuazione di meccanismi di assistenza in RNN
 - 12.5.2. Utilizzo di meccanismi di assistenza per migliorare la precisione dei modelli
 - 12.5.3. Vantaggi dei meccanismi di assistenza nelle reti neurali

- 12.6. Modelli *Transformers*
 - 12.6.1. Utilizzo dei modelli *Transformers* per l'elaborazione del linguaggio naturale
 - 12.6.2. Applicazione dei modelli *Transformers* per la visione
 - 12.6.3. Vantaggi dei modelli *Transformers*
- 12.7. *Transformers* per la visione
 - 12.7.1. Uso dei modelli *Transformers* per la visione
 - 12.7.2. Elaborazione dei dati di immagine
 - 12.7.3. Addestramento dei modelli *Transformers* per la visione
- 12.8. Libreria di *Transformers* di *Hugging Face*
 - 12.8.1. Uso della libreria di *Transformers* di *Hugging Face*
 - 12.8.2. Applicazione della libreria *Transformers* di *Hugging Face*
 - 12.8.3. Vantaggi della libreria di *Transformers* di *Hugging Face*
- 12.9. Altre Librerie di *Transformers*: Confronto
 - 12.9.1. Confronto tra le diverse librerie di *Transformers*
 - 12.9.2. Uso di altre librerie di *Transformers*
 - 12.9.3. Vantaggi delle altre librerie di *Transformers*
- 12.10. Sviluppo di un'applicazione NLP con RNN e Assistenza: Applicazione Pratica
 - 12.10.1. Sviluppare di un'applicazione di elaborazione di linguaggio naturale con RNN e assistenza
 - 12.10.2. Utilizzo di RNN, meccanismi di assistenza e modelli *Transformers* nell'applicazione
 - 12.10.3. Valutazione dell'attuazione pratica

Modulo 13. *Autoencoders*, GAN, e modelli di diffusione

- 13.1. Rappresentazione dei dati efficienti
 - 13.1.1. Riduzione della dimensionalità
 - 13.1.2. Deep Learning
 - 13.1.3. Rappresentazioni compatte
- 13.2. Realizzazione di PCA con un encoder automatico lineare incompleto
 - 13.2.1. Processo di addestramento
 - 13.2.2. Implementazione in Python
 - 13.2.3. Uso dei dati di prova
- 13.3. Codificatori automatici raggruppati
 - 13.3.1. Reti neurali profonde
 - 13.3.2. Costruzione di architetture di codifica
 - 13.3.3. Uso della regolarizzazione

- 13.4. Autocodificatori convoluzionali
 - 13.4.1. Progettazione di modelli convoluzionali
 - 13.4.2. Addestramento di modelli convoluzionali
 - 13.4.3. Valutazione dei risultati
- 13.5. Eliminazione del rumore dei codificatori automatici
 - 13.5.1. Applicare filtro
 - 13.5.2. Progettazione di modelli di codificazione
 - 13.5.3. Uso di tecniche di regolarizzazione
- 13.6. Codificatori automatici dispersi
 - 13.6.1. Aumentare l'efficienza della codifica
 - 13.6.2. Ridurre al minimo il numero di parametri
 - 13.6.3. Uso di tecniche di regolarizzazione
- 13.7. Codificatori automatici variazionali
 - 13.7.1. Utilizzo dell'ottimizzazione variazionale
 - 13.7.2. Deep learning non supervisionato
 - 13.7.3. Rappresentazioni latenti profonde
- 13.8. Creazione di immagini MNIST di moda
 - 13.8.1. Riconoscimento di pattern
 - 13.8.2. Creazione di immagini
 - 13.8.3. Addestramento delle reti neurali profonde
- 13.9. Reti generative avversarie e modelli di diffusione
 - 13.9.1. Generazione di contenuti da immagini
 - 13.9.2. Modello di distribuzione dei dati
 - 13.9.3. Uso di reti avversarie
- 13.10. L'implementazione dei modelli
 - 13.10.1. Applicazione pratica
 - 13.10.2. L'implementazione dei modelli
 - 13.10.3. Utilizzo dei dati di prova
 - 13.10.4. Valutazione dei risultati

Modulo 14. Computazione bio-ispirata

- 14.1. Introduzione alla computazione bio-ispirata
 - 14.1.1. Introduzione alla computazione bio-ispirata
- 14.2. Algoritmi di adattamento sociale
 - 14.2.1. Computazione bio-ispirata basata su colonie di formiche
 - 14.2.2. Varianti degli algoritmi di colonia di formiche
 - 14.2.3. Elaborazione particellare basata su cloud
- 14.3. Algoritmi genetici
 - 14.3.1. Struttura generale
 - 14.3.2. Implementazioni dei principali operatori
- 14.4. Strategie spaziali di esplorazione-sfruttamento per algoritmi genetici
 - 14.4.1. Algoritmo CHC
 - 14.4.2. Problemi multimodali
- 14.5. Modelli di calcolo evolutivo (I)
 - 14.5.1. Strategie evolutive
 - 14.5.2. Programmazione evolutiva
 - 14.5.3. Algoritmi basati sull'evoluzione differenziale
- 14.6. Modelli di calcolo evolutivo (II)
 - 14.6.1. Modelli evolutivi basati sulla stima delle distribuzioni (EDA)
 - 14.6.2. Programmazione genetica
- 14.7. Programmazione evolutiva applicata ai problemi di apprendimento
 - 14.7.1. Apprendimento basato sulle regole
 - 14.7.2. Metodi evolutivi nei problemi di selezione delle istanze
- 14.8. Problemi multi-obiettivo
 - 14.8.1. Concetto di dominanza
 - 14.8.2. Applicazione degli algoritmi evolutivi ai problemi multi-obiettivo
- 14.9. Reti neurali (I)
 - 14.9.1. Introduzione alle reti neurali
 - 14.9.2. Esempio pratico con le reti neurali
- 14.10. Reti neurali (II)
 - 14.10.1. Casi di utilizzo delle reti neurali nella ricerca medica
 - 14.10.2. Casi di utilizzo delle reti neurali in economia
 - 14.10.3. Casi di utilizzo delle reti neurali nella visione artificiale

Modulo 15. Intelligenza Artificiale: Strategie e applicazioni

- 15.1. Servizi finanziari
 - 15.1.1. Le implicazioni dell'Intelligenza Artificiale (IA) nei servizi finanziari: Opportunità e sfide
 - 15.1.2. Casi d'uso
 - 15.1.3. Potenziali rischi legati all'uso dell'IA
 - 15.1.4. Potenziali sviluppi/utilizzi futuri dell'IA
- 15.2. Implicazioni dell'Intelligenza Artificiale nel servizio sanitario
 - 15.2.1. Implicazioni dell'IA nel settore sanitario: Opportunità e sfide
 - 15.2.2. Casi d'uso
- 15.3. Rischi legati all'uso dell'IA nel servizio sanitario
 - 15.3.1. Potenziali rischi legati all'uso dell'IA
 - 15.3.2. Potenziali sviluppi/utilizzi futuri dell'IA
- 15.4. *Retail*
 - 15.4.1. Implicazioni dell'IA nel *Retail*: Opportunità e sfide
 - 15.4.2. Casi d'uso
 - 15.4.3. Potenziali rischi legati all'uso dell'IA
 - 15.4.4. Potenziali sviluppi/utilizzi futuri dell'IA
- 15.5. Industria
 - 15.5.1. Implicazioni dell'IA nell'Industria: Opportunità e sfide
 - 15.5.2. Casi d'uso
- 15.6. Potenziali rischi legati all'uso dell'IA nell'Industria
 - 15.6.1. Casi d'uso
 - 15.6.2. Potenziali rischi legati all'uso dell'IA
 - 15.6.3. Potenziali sviluppi/utilizzi futuri dell'IA
- 15.7. Pubblica Amministrazione
 - 15.7.1. Implicazioni dell'IA nella Pubblica Amministrazione: Opportunità e sfide
 - 15.7.2. Casi d'uso
 - 15.7.3. Potenziali rischi legati all'uso dell'IA
 - 15.7.4. Potenziali sviluppi/utilizzi futuri dell'IA
- 15.8. Educazione
 - 15.8.1. Implicazioni dell'IA nell'Educazione: Opportunità e sfide
 - 15.8.2. Casi d'uso
 - 15.8.3. Potenziali rischi legati all'uso dell'IA
 - 15.8.4. Potenziali sviluppi/utilizzi futuri dell'IA

- 15.9. Silvicoltura e agricoltura
 - 15.9.1. Implicazioni dell'IA nella silvicoltura e nell'agricoltura: Opportunità e sfide
 - 15.9.2. Casi d'uso
 - 15.9.3. Potenziali rischi legati all'uso dell'IA
 - 15.9.4. Potenziali sviluppi/utilizzi futuri dell'IA
- 15.10. Risorse Umane
 - 15.10.1. Implicazioni dell'IA nelle Risorse Umane: Opportunità e sfide
 - 15.10.2. Casi d'uso
 - 15.10.3. Potenziali rischi legati all'uso dell'IA
 - 15.10.4. Potenziali sviluppi/utilizzi futuri dell'IA

Modulo 16. Migliorare la produttività nello sviluppo del software con la IA

- 16.1. Preparazione dell'ambiente di sviluppo adeguati
 - 16.1.1. Selezione degli strumenti essenziali per lo sviluppo dell'IA
 - 16.1.2. Configurazione degli strumenti scelti
 - 16.1.3. Implementazione di pipeline CI/CD adatte ai progetti di IA
 - 16.1.4. Gestione efficiente delle dipendenze e delle versioni negli ambienti di sviluppo
- 16.2. Estensioni indispensabili per l'IA in Visual Studio Code
 - 16.2.1. Esplorazione e selezione delle estensioni AI per Visual Studio Code
 - 16.2.2. Integrazione degli strumenti di analisi statica e dinamica nell'IDE
 - 16.2.3. Automatizzare le attività ripetitive con estensioni specifiche
 - 16.2.4. Personalizzazione dell'ambiente di sviluppo per migliorare l'efficienza
- 16.3. Progettazione No-code di Interfacce Utente con Flutterflow
 - 16.3.1. Principi di progettazione *No-code* e la loro applicazione nelle interfacce utente
 - 16.3.2. Incorporazione di elementi di IA nella progettazione di interfacce visive
 - 16.3.3. Strumenti e piattaforme per la creazione *No-code* di interfacce intelligenti
 - 16.3.4. Valutazione e miglioramento continuo delle interfacce *No-code* con IA
- 16.4. Ottimizzazione del codice con ChatGPT
 - 16.4.1. Identificazione di codice duplicato
 - 16.4.2. Refactoring
 - 16.4.3. Creazione di codice leggibile
 - 16.4.4. Capire cosa fa il codice
 - 16.4.5. Migliorare la denominazione di variabili e funzioni
 - 16.4.6. Creare documentazione automatica

- 16.5. Gestire i repository con l'IA usando ChatGPT
 - 16.5.1. Automazione dei processi di controllo di versione con tecniche di IA
 - 16.5.2. Rilevamento dei conflitti e risoluzione automatica in ambienti collaborativi
 - 16.5.3. Analisi predittiva delle modifiche e delle tendenze nei repository di codice
 - 16.5.4. Miglioramento del controllo di versione con l'IA
- 16.6. Integrazione dell'IA nella gestione dei database con AskYourDatabase
 - 16.6.1. Ottimizzazione delle query e delle prestazioni con tecniche di IA
 - 16.6.2. Analisi predittiva dei modelli di accesso ai database
 - 16.6.3. Implementazione di sistemi di raccomandazione per ottimizzare la struttura dei database
 - 16.6.4. Monitoraggio e rilevamento proattivo di potenziali problemi del database
- 16.7. Individuazione dei guasti e creazione di test unitari con IA utilizzando ChatGPT
 - 16.7.1. Generazione automatica di casi di test con tecniche di IA
 - 16.7.2. Individuazione precoce di vulnerabilità e bug mediante l'analisi statica con l'IA
 - 16.7.3. Miglioramento della copertura dei test attraverso l'identificazione di aree critiche mediante l'IA
- 16.8. *Pair Programming* con GitHub Copilot
 - 16.8.1. Integrazione e utilizzo efficace di GitHub Copilot in sessioni di *Pair Programming*
 - 16.8.2. Integrazione e Miglioramenti alla comunicazione e alla collaborazione degli sviluppatori con GitHub Copilot
 - 16.8.3. Integrazione di strategie per sfruttare al meglio i suggerimenti di codice generati da GitHub Copilot
 - 16.8.4. Integrazione di casi di studio e best practice in *Pair Programming* assistita da IA
- 16.9. Traduzione automatica tra linguaggi di programmazione utilizzando ChatGPT
 - 16.9.1. Strumenti e servizi di traduzione automatica specifici per i linguaggi di programmazione
 - 16.9.2. Adattamento degli algoritmi di traduzione automatica ai contesti di sviluppo
 - 16.9.3. Miglioramento dell'interoperabilità tra lingue diverse attraverso la traduzione automatica
 - 16.9.4. Valutare e mitigare le potenziali sfide e limitazioni della traduzione automatica

- 16.10. Strumenti di IA consigliati per migliorare la produttività
 - 16.10.1. Analisi comparativa degli strumenti di IA per lo sviluppo di software
 - 16.10.2. Integrazione degli strumenti di IA nei flussi di lavoro
 - 16.10.3. Automazione di attività di routine con strumenti di IA
 - 16.10.4. Valutazione e selezione degli strumenti in base al contesto e ai requisiti del progetto

Modulo 17. Architettura del software con IA

- 17.1. Ottimizzazione e gestione delle prestazioni negli strumenti di IA con l'aiuto di ChatGPT
 - 17.1.1. Analisi delle prestazioni e profilazione negli strumenti di IA
 - 17.1.2. Strategie di ottimizzazione degli algoritmi e dei modelli di IA
 - 17.1.3. Implementazione di tecniche di IA *caching* e parallelizzazione per migliorare le prestazioni
 - 17.1.4. Strumenti e metodologie per il monitoraggio continuo delle prestazioni in tempo reale
- 17.2. Scalabilità nelle applicazioni di IA utilizzando ChatGPT
 - 17.2.1. Progettazione di architetture scalabili per applicazioni di IA
 - 17.2.2. Implementazione di tecniche di partizione e condivisione del carico
 - 17.2.3. Gestione dei flussi di lavoro e dei carichi di lavoro nei sistemi scalabili
 - 17.2.4. Strategie di espansione orizzontale e verticale in ambienti a domanda variabile
- 17.3. Manutenzione delle applicazioni con IA utilizzando ChatGPT
 - 17.3.1. Principi di progettazione per facilitare la manutenibilità nei progetti di IA
 - 17.3.2. Strategie di documentazione specifiche per i modelli e gli algoritmi di IA
 - 17.3.3. Implementazione di test unitari e di integrazione per facilitare il mantenimento
 - 17.3.4. Metodi per il refactoring e il miglioramento continuo nei sistemi con componenti di IA
- 17.4. Progettazione di sistemi su larga scala
 - 17.4.1. Principi architettonici per la progettazione di sistemi su larga scala
 - 17.4.2. Decomposizione di sistemi complessi in microservizi
 - 17.4.3. Implementazione di pattern di progettazione specifici per i sistemi distribuiti
 - 17.4.4. Strategie per la gestione della complessità in architetture su larga scala con componenti di IA

- 17.5. Archiviazione di dati su larga scala per strumenti di IA
 - 17.5.1. Selezione di tecnologie di archiviazione dei dati scalabili
 - 17.5.2. Progettazione di schemi di database per la gestione efficiente di grandi volumi di dati
 - 17.5.3. Strategie di partizione e di replica in ambienti di archiviazione di dati massivi
 - 17.5.4. Implementazione di sistemi di gestione dei dati per garantire l'integrità e la disponibilità nei progetti di IA
- 17.6. Struttura dei dati con IA utilizzando ChatGPT
 - 17.6.1. Adattamento di strutture dati classiche per l'uso in algoritmi di IA
 - 17.6.2. Progettazione e ottimizzazione di strutture di dati specifiche con ChatGPT
 - 17.6.3. Integrazione di strutture di dati efficienti in sistemi ad alta intensità di dati
 - 17.6.4. Strategie per la manipolazione e l'archiviazione dei dati in tempo reale nelle strutture di dati con IA
- 17.7. Algoritmi di programmazione per prodotti di IA
 - 17.7.1. Sviluppo e implementazione di algoritmi specifici per applicazioni di IA
 - 17.7.2. Strategie di selezione degli algoritmi in base al tipo di problema e ai requisiti del prodotto
 - 17.7.3. Adattamento di algoritmi classici per l'integrazione in sistemi di intelligenza artificiale
 - 17.7.4. Valutazione e confronto delle prestazioni tra diversi algoritmi in contesti di sviluppo con IA
- 17.8. Modelli di progettazione per lo sviluppo con IA
 - 17.8.1. Identificazione e applicazione di modelli di progettazione comuni nei progetti con componenti IA
 - 17.8.2. Sviluppo di modelli specifici per l'integrazione di modelli e algoritmi nei sistemi esistenti
 - 17.8.3. Strategie di implementazione dei modelli per migliorare la riutilizzabilità e la manutenibilità nei progetti IA
 - 17.8.4. Casi di studio e buone pratiche per l'applicazione di modelli di progettazione su architetture con IA
- 17.9. Implementazione di clean architecture usando ChatGPT
 - 17.9.1. Principi e concetti fondamentali di *Clean Architecture*
 - 17.9.2. Adattamento di *Clean Architecture* a progetti con componenti IA
 - 17.9.3. Distribuzione di livelli e dipendenze in sistemi con architettura pulita
 - 17.9.4. Vantaggi e sfide dell'implementazione di *Clean Architecture* nello sviluppo di software con IA

- 17.10. Sviluppo di software sicuro in applicazioni web con DeepCode
 - 17.10.1. Principi di sicurezza nello sviluppo di software con componenti IA
 - 17.10.2. Identificazione e mitigazione di potenziali vulnerabilità in modelli e algoritmi di IA
 - 17.10.3. Implementare pratiche di sviluppo sicuro nelle applicazioni Web con funzionalità di Intelligenza Artificiale
 - 17.10.4. Strategie per la protezione dei dati sensibili e la prevenzione degli attacchi nei progetti con IA

Modulo 18. Progetti web con IA

- 18.1. Preparazione dell'ambiente di lavoro per lo sviluppo web con IA
 - 18.1.1. Configurazione di ambienti di sviluppo web per progetti con intelligenza artificiale
 - 18.1.2. Selezione degli strumenti essenziali per lo sviluppo e preparazione dell'IA
 - 18.1.3. Integrazione di librerie e *framework* specifici per progetti web con intelligenza artificiale
 - 18.1.4. Implementare buone pratiche nella configurazione di ambienti di sviluppo collaborativi
- 18.2. Creazione di Workspace per i progetti di IA con GitHub Copilot
 - 18.2.1. Progettazione e organizzazione efficace di *workspace* per progetti web con componenti di intelligenza artificiale
 - 18.2.2. Utilizzo di strumenti di gestione dei progetti e controllo delle versioni nel *workspace*
 - 18.2.3. Strategie di collaborazione e comunicazione efficienti nel team di sviluppo
 - 18.2.4. Adattamento del *workspace* alle esigenze specifiche dei progetti web con IA
- 18.3. Modelli di progettazione in prodotti con Github Copilot
 - 18.3.1. Identificazione e applicazione di modelli di progettazione comuni nelle interfacce utente con elementi di intelligenza artificiale
 - 18.3.2. Sviluppo di modelli specifici per migliorare l'esperienza utente nei progetti Web con IA
 - 18.3.3. Integrazione dei modelli di progettazione nell'architettura generale dei progetti web con l'intelligenza artificiale
 - 18.3.4. Valutazione e selezione di modelli di progettazione appropriati in base al contesto del progetto

- 18.4. Sviluppo Frontend con GitHub Copilot
 - 18.4.1. Integrazione dei modelli IA nel livello di presentazione dei progetti Web
 - 18.4.2. Sviluppo di interfacce utente adattive con elementi di intelligenza artificiale
 - 18.4.3. Implementazione delle funzionalità di elaborazione del linguaggio naturale (PLN) in Frontend
 - 18.4.4. Strategie per l'ottimizzazione delle prestazioni nello sviluppo di Frontend con IA
- 18.5. Creazione di Database usando GitHub Copilot
 - 18.5.1. Selezione di tecnologie di database per progetti web con intelligenza artificiale
 - 18.5.2. Progettazione di schemi di database per l'archiviazione e la gestione dei dati relativi all'IA
 - 18.5.3. Implementazione di sistemi di archiviazione efficienti per grandi volumi di dati generati da modelli IA
 - 18.5.4. Strategie per la sicurezza e la protezione dei dati sensibili nei database di progetti web con IA
- 18.6. Sviluppo Backend con GitHub Copilot
 - 18.6.1. Integrazione di modelli e servizi IA nella logica di business del backend
 - 18.6.2. Sviluppo di API ed endpoint specifici per la comunicazione tra Frontend e componenti IA
 - 18.6.3. Implementazione della logica di elaborazione dati e del processo decisionale nel backend con l'intelligenza artificiale
 - 18.6.4. Strategie per la scalabilità e le prestazioni nello sviluppo Backend di progetti web con IA
- 18.7. Ottimizzare il processo di implementazione del tuo sito web
 - 18.7.1. Automazione dei processi di costruzione e implementazione di progetti web con ChatGPT
 - 18.7.2. Implementazione di pipeline CI/CD adattate alle applicazioni web con Github Copilot
 - 18.7.3. Strategie per una gestione efficiente delle versioni e degli aggiornamenti nelle distribuzioni continue
 - 18.7.4. Monitoraggio e analisi post-implementazione per il miglioramento continuo del processo
- 18.8. IA del Cloud Computing
 - 18.8.1. Integrazione dei servizi di intelligenza artificiale nelle piattaforme di cloud computing
 - 18.8.2. Sviluppo di soluzioni scalabili e distribuite utilizzando servizi cloud con funzionalità IA
 - 18.8.3. Strategie per una gestione efficiente delle risorse e dei costi in ambienti cloud con applicazioni web con IA
 - 18.8.4. Valutazione e confronto dei fornitori di servizi cloud per progetti web con Intelligenza Artificiale

- 18.9. Creazione di un progetto IA per ambienti LAMP con l'aiuto di ChatGPT
 - 18.9.1. Adattamento di progetti web basati su stack LAMP per includere componenti IA
 - 18.9.2. Integrazione delle librerie e *framework* specifici dell'IA in ambienti LAMP
 - 18.9.3. Sviluppo di funzionalità IA che completano l'architettura LAMP tradizionale
 - 18.9.4. Strategie per l'ottimizzazione e la manutenzione di progetti web con IA in ambienti LAMP
- 18.10. Creare un progetto con IA per ambienti MEVN usando ChatGPT
 - 18.10.1. Integrazione di tecnologie e strumenti dello stack MEVN con componenti IA
 - 18.10.2. Sviluppo di applicazioni web moderne e scalabili in ambienti MEVN con funzionalità IA
 - 18.10.3. Implementazione delle funzionalità di elaborazione dati e machine learning nei progetti MEVN
 - 18.10.4. Strategie per migliorare le prestazioni e la sicurezza delle applicazioni Web con AI in ambienti MEVN

Modulo 19. Applicazioni mobili con l'IA

- 19.1. Preparazione di ambiente di lavoro per lo sviluppo mobile con IA
 - 19.1.1. Configurazione di ambienti di sviluppo mobile per progetti con intelligenza artificiale
 - 19.1.2. Selezione e preparazione di strumenti specifici per lo sviluppo di applicazioni mobili con IA
 - 19.1.3. Integrazione delle librerie e *frameworks* dell'IA in ambienti di sviluppo mobile
 - 19.1.4. Configurazione di emulatori e dispositivi reali per testare applicazioni mobili con componenti di intelligenza artificiale
- 19.2. Creazione di un *Workspace* con GitHub Copilot
 - 19.2.1. Integrazione di GitHub Copilot in ambienti di sviluppo mobile
 - 19.2.2. Utilizzo efficace di GitHub Copilot per la generazione di codice nei progetti con IA
 - 19.2.3. Strategie per la collaborazione tra sviluppatori quando si utilizza GitHub Copilot nello *workspace*
 - 19.2.4. Buone pratiche e limitazioni nell'uso di GitHub Copilot nello sviluppo di applicazioni mobili con IA
- 19.3. Impostazioni di Firebase
 - 19.3.1. Configurazione iniziale di un progetto in Firebase per lo sviluppo mobile
 - 19.3.2. Integrazione di Firebase nelle app mobili con funzionalità di Intelligenza Artificiale
 - 19.3.3. Utilizzo dei servizi Firebase come database, autenticazione e notifiche nei progetti IA
 - 19.3.4. Strategie per la gestione di eventi e dati in tempo reale nelle app mobili con Firebase

- 19.4. Concetti di Clean Architecture, DataSources, Repositories
 - 19.4.1. Principi fondamentali di Clean Architecture nello sviluppo mobile con IA
 - 19.4.2. Distribuzione di livelli DataSources e repository con GitHub Copilot
 - 19.4.3. Progettazione e strutturazione di componenti in progetti mobili con Github Copilot
 - 19.4.4. Vantaggi e sfide dell'implementazione di *Clean Architecture* nello sviluppo di software con IA
- 19.5. Creazione di Schermate di Autenticazione con GitHub Copilot
 - 19.5.1. Progettazione e sviluppo di interfacce utente per le schermate di autenticazione nelle applicazioni mobili con IA
 - 19.5.2. Integrazione dei servizi di autenticazione con Firebase nella schermata di accesso
 - 19.5.3. Utilizzo di tecniche di sicurezza e protezione dei dati nella schermata di autenticazione
 - 19.5.4. Personalizzazione e adattamento dell'esperienza utente nella schermata di autenticazione
- 19.6. Creazione di Dashboard e Navigazione con GitHub Copilot
 - 19.6.1. Progettazione e sviluppo di *Dashboard* con elementi di intelligenza artificiale
 - 19.6.2. Implementazione di sistemi di navigazione efficienti nelle applicazioni mobili con IA
 - 19.6.3. Integrazione delle funzionalità di IA nel *Dashboard* per migliorare l'esperienza dell'utente
- 19.7. Creazione di Schermate con Elenco usando GitHub Copilot
 - 19.7.1. Sviluppo di interfacce utente per display con elenchi in applicazioni mobili con IA
 - 19.7.2. Integrazione di algoritmi di raccomandazione e filtraggio nella schermata di elenco
 - 19.7.3. Utilizzo di modelli di progettazione per una presentazione efficace dei dati nell'elenco
 - 19.7.4. Strategie per il caricamento efficiente dei dati in tempo reale sullo schermo con elenco
- 19.8. Creazione di Schermate di Dettagli con GitHub Copilot
 - 19.8.1. Progettazione e sviluppo di interfacce utente dettagliate per la presentazione di informazioni specifiche
 - 19.8.2. Integrazione delle funzionalità della IA per arricchire la schermata di dettaglio
 - 19.8.3. Implementare interazioni e animazioni nella schermata di dettaglio
 - 19.8.4. Strategie per l'ottimizzazione delle prestazioni di caricamento e visualizzazione dei dettagli nelle app mobili con IA

- 19.9. Creazione di Schermate di Settings con GitHub Copilot
 - 19.9.1. Sviluppo di interfacce utente per configurazione e dei mettere a in applicazioni mobili con IA
 - 19.9.2. Integrazione di impostazioni personalizzate relative ai componenti di intelligenza artificiale
 - 19.9.3. Implementazione delle opzioni di personalizzazione e delle preferenze nella schermata di configurazione
 - 19.9.4. Strategie per l'usabilità e la chiarezza nella presentazione delle opzioni sullo schermo di *setting*
- 19.10. Creare Icone, *Splash* e Risorse Grafiche per l'app con IA
 - 19.10.1. Design e creazione di icone accattivanti per rappresentare l'app mobile con IA
 - 19.10.2. Sviluppo di schermate di avvio (*splash*) con elementi visivi di impatto
 - 19.10.3. Selezione e adattamento delle risorse grafiche per migliorare l'estetica dell'app mobile
 - 19.10.4. Strategie per la coerenza e il branding visivo negli elementi grafici dell'app con IA

Modulo 20. IA per QA Testing

- 20.1. Ciclo di vita del *Testing*
 - 20.1.1. Descrizione e comprensione del ciclo di vita del *testing* nello sviluppo di software
 - 20.1.2. Fasi del ciclo di vita del *testing* e la loro importanza ai fini della garanzia della qualità
 - 20.1.3. Integrazione dell'intelligenza artificiale in diverse fasi del ciclo di vita del *testing*
 - 20.1.4. Strategie per il miglioramento continuo del ciclo di vita del *testing* mediante l'uso della IA
- 20.2. Test Case e Bug Detection con l'aiuto di ChatGPT
 - 20.2.1. Progettazione e scrittura efficaci di casi di prova nel contesto del QA *Testing*
 - 20.2.2. Identificazione di bug ed errori durante l'esecuzione di test case
 - 20.2.3. Applicazione di tecniche di rilevamento precoce dei bug mediante analisi statica
 - 20.2.4. Utilizzo di strumenti di intelligenza artificiale per l'identificazione automatica dei bug nei test case
- 20.3. Tipi di *Testing*
 - 20.3.1. Esplorazione di diversi tipi di *testing* nel settore del QA
 - 20.3.2. Test unitari, di integrazione, funzionali e di accettazione: caratteristiche e applicazioni
 - 20.3.3. Strategie per la corretta selezione e combinazione dei tipi di *testing* nei progetti con ChatGPT
 - 20.3.4. Adeguamento dei tipi di *testing* convenzionali a progetti con ChatGPT

- 20.4. Creare un piano di test usando ChatGPT
 - 20.4.1. Progettazione e strutturazione di un piano di test completo
 - 20.4.2. Identificare requisiti e scenari di test nei progetti con IA
 - 20.4.3. Strategie per la pianificazione di test manuali e automatizzati
 - 20.4.4. Valutazione e adeguamento continuo del piano di prova in funzione dello sviluppo del progetto
- 20.5. Rilevazione e segnalazione di *Bugs* con IA
 - 20.5.1. Implementare tecniche di rilevamento automatico dei bug utilizzando algoritmi di apprendimento automatico
 - 20.5.2. Utilizzo di ChatGPT per l'analisi dinamica del codice alla ricerca di possibili errori
 - 20.5.3. Strategie per la generazione automatica di report dettagliati sui bug rilevati usando ChatGPT
 - 20.5.4. Collaborazione efficace tra team di sviluppo e QA nella gestione dei bug identificati dall'IA
- 20.6. Creazione di test automatizzati con IA
 - 20.6.1. Sviluppo di script di test automatizzati per progetti usando ChatGPT
 - 20.6.2. Integrazione di strumenti di automazione dei test basati su IA
 - 20.6.3. Uso di ChatGPT per la generazione dinamica di test case automatizzati
 - 20.6.4. Strategie per l'esecuzione efficiente e la manutenzione dei test automatizzati nei progetti con IA
- 20.7. *API Testing*
 - 20.7.1. Concetti fondamentali di *API testing* e importanza nel QA
 - 20.7.2. Sviluppo di test per la verifica delle API in ambienti usando ChatGPT
 - 20.7.3. Strategie per la convalida dei dati e dei risultati nelle API testing con ChatGPT
 - 20.7.4. Utilizzo di strumenti specifici per il *testing* API in progetti con intelligenza artificiale
- 20.8. Strumenti IA per il *Web Testing*
 - 20.8.1. Esplorazione di strumenti di intelligenza artificiale per l'automazione dei test in ambienti web
 - 20.8.2. Integrazione di tecnologie di riconoscimento degli elementi e analisi visiva sul *Web testing*
 - 20.8.3. Strategie per il rilevamento automatico di modifiche e problemi di prestazioni nelle applicazioni Web usando ChatGPT
 - 20.8.4. Valutazione di strumenti specifici per migliorare l'efficienza nel web testing con IA
- 20.9. *Mobile Testing* Mediante IA
 - 20.9.1. Sviluppo di strategie di *testing* per applicazioni mobili con componenti di intelligenza artificiale
 - 20.9.2. Integrazione di strumenti di *testing* specifiche per piattaforme mobili basate sull'IA
 - 20.9.3. Utilizzo di ChatGPT per il rilevamento di problemi nelle prestazioni delle applicazioni mobili
 - 20.9.4. Strategie per la convalida di interfacce e funzioni specifiche delle applicazioni mobili tramite IA
- 20.10. Strumenti QA con IA
 - 20.10.1. Esplorazione di strumenti e piattaforme QA che incorporano funzionalità di Intelligenza Artificiale
 - 20.10.2. Valutazione degli strumenti per la gestione e l'esecuzione efficiente dei test di progetto con IA
 - 20.10.3. Uso di ChatGPT per la generazione e ottimizzazione di test case
 - 20.10.4. Strategie per la selezione e l'adozione efficace di strumenti QA con funzionalità IA

07 Tirocinio

Dopo aver superato il periodo teorico online, il programma prevede un periodo di formazione pratica in un'azienda di riferimento, specializzata nell'uso dell'Intelligenza Artificiale nella Programmazione. Lo studente disporrà del supporto di un tutor personale che lo accompagnerà durante tutto il percorso, sia nella preparazione che nello svolgimento del tirocinio.





“

Svolgerai il tirocinio presso aziende leader del settore tecnologico, specializzate nell'utilizzo dell'Intelligenza Artificiale per lo sviluppo di software"

Il periodo di formazione pratica di questo programma in Intelligenza Artificiale nella Programmazione è costituito da un tirocinio pratico in una prestigiosa azienda specializzata in materia, della durata di 3 settimane, dal lunedì al venerdì e con giornate di 8 ore consecutive di formazione pratica, affiancate da uno specialista aggiunto. Questo tirocinio permetterà all'informatico di sviluppare progetti reali di IA, insieme a un team di esperti di riferimento in questo campo, applicando le procedure e gli strumenti più innovativi, implementato l'ultima tecnologia.

In questa proposta di formazione, di carattere completamente pratico, le attività sono dirette allo sviluppo e al perfezionamento delle competenze necessarie per l'utilizzo dell'Intelligenza Artificiale nei lavori di Programmazione, in settori e condizioni che richiedono un alto livello di qualificazione, e che sono orientate alla formazione specifica per l'esercizio dell'attività. Si tratta, senza dubbio, di un'opportunità unica per imparare lavorando.

La parte pratica sarà svolta con la partecipazione attiva dello studente svolgendo le attività e le procedure di ogni area di competenza (imparare a imparare e imparare a fare), con l'accompagnamento e la guida di insegnanti e altri compagni di formazione che facilitano il lavoro di squadra e l'integrazione multidisciplinare come competenze trasversali per la pratica dell'Intelligenza Artificiale nella Programmazione (imparare a essere e imparare a relazionarsi).



Le procedure descritte di seguito saranno la base della parte pratica della formazione, e la loro realizzazione sarà soggetta alla disponibilità propria del centro e al suo volume di lavoro, essendo le attività proposte come segue:

Modulo	Attività Pratica
Sviluppo di Modelli di Apprendimento Automatico	Addestrare modelli di reti neurali
	Ottimizzare gli algoritmi di apprendimento automatico
	Implementare tecniche di regolarizzazione e ottimizzazione
	Valutare e confrontare la precisione di diversi modelli
Elaborazione del Linguaggio Naturale (PLN)	Sviluppare algoritmi per l'analisi del sentiment
	Creare sistemi di riconoscimento delle entità denominate (NER)
	Implementare modelli di traduzione automatica
	Progettare filtri spam utilizzando tecniche PLN
Computer Vision	Sviluppare algoritmi per il riconoscimento facciale
	Implementare sistemi di rilevamento degli oggetti nelle immagini
	Ottimizzare gli algoritmi di segmentazione delle immagini
	Valutare l'accuratezza dei modelli di visione computerizzata
Sistemi di Raccomandazione	Implementare motori di raccomandazione basati sul filtraggio collaborativo
	Sviluppare sistemi di raccomandazione ibridi
	Ottimizzare gli algoritmi di raccomandazione in tempo reale
	Valutare l'efficacia dei sistemi di raccomandazione attraverso metriche appropriate

Modulo	Attività Pratica
Applicazione Pratica dell'IA	Sviluppare <i>chatbot</i> utilizzando tecniche di elaborazione del linguaggio naturale
	Implementare sistemi di riconoscimento vocale per applicazioni mobili
	Progettare modelli di analisi predittiva per le aziende
	Creare sistemi automatici di generazione dei contenuti
Etica e Sicurezza in IA	Valutare il bias nei modelli di apprendimento automatico
	Implementare tecniche di privacy negli algoritmi IA
	Sviluppare politiche etiche nell'implementazione dei sistemi di IA
	Controllare i sistemi di IA per identificare vulnerabilità e rischi
Ricerca e Sviluppo	Ricerca nuove tecniche di deep learning
	Sviluppare prototipi di sistemi innovativi utilizzando l'IA
	Pubblicare articoli in conferenze e riviste specializzate
	Collaborare con team multidisciplinari su progetti di ricerca applicata

Assicurazione di responsabilità civile

La preoccupazione principale di questa istituzione è quella di garantire la sicurezza sia dei tirocinanti e degli altri agenti che collaborano ai processi di tirocinio in azienda. All'interno delle misure rivolte a questo fine ultimo, esiste la risposta a qualsiasi incidente che possa verificarsi durante il processo di insegnamento-apprendimento.

A tal fine, questa istituzione educativa si impegna a stipulare un'assicurazione di responsabilità civile per coprire qualsiasi eventualità che possa insorgere durante la permanenza presso il centro di tirocinio.

La polizza di responsabilità civile per i tirocinanti deve garantire una copertura completa e deve essere stipulata prima dell'inizio del periodo di tirocinio. Grazie a questa garanzia, il professionista si sentirà privo di ogni tipo di preoccupazione nel caso di eventuali situazioni impreviste che possano sorgere durante il tirocinio e potrà godere di una copertura assicurativa fino al termine dello stesso.



Condizioni generali del tirocinio

Le condizioni generali dell'accordo di tirocinio per il programma sono le seguenti:

1. TUTORAGGIO: durante il Master Semipresenziale agli studenti verranno assegnati due tutor che li seguiranno durante tutto il percorso, risolvendo eventuali dubbi e domande. Da un lato, lo studente disporrà di un tutor professionale appartenente al centro di inserimento lavorativo che lo guiderà e lo supporterà in ogni momento. Dall'altro lato, allo studente verrà assegnato anche un tutor accademico che avrà il compito di coordinare e aiutare lo studente durante l'intero processo, risolvendo i dubbi e fornendogli tutto ciò di cui potrebbe aver bisogno. In questo modo, il professionista sarà accompagnato in ogni momento e potrà risolvere tutti gli eventuali dubbi, sia di natura pratica che accademica.

2. DURATA: il programma del tirocinio avrà una durata di tre settimane consecutive di preparazione pratica, distribuite in giornate di 8 ore lavorative, per cinque giorni alla settimana. I giorni di frequenza e l'orario saranno di competenza del centro, che informerà debitamente e preventivamente il professionista, con un sufficiente anticipo per facilitarne l'organizzazione.

3. ASSENZE: in caso di mancata presentazione il giorno di inizio del Master Semipresenziale, lo studente perderà il diritto allo stesso senza possibilità di rimborso o di modifica di date. L'assenza per più di due giorni senza un giustificato motivo/certificato medico comporterà la rinuncia dello studente al tirocinio e, pertanto, la relativa automatica cessazione. In caso di ulteriori problemi durante lo svolgimento del tirocinio, essi dovranno essere debitamente e urgentemente segnalati al tutor accademico.

4. CERTIFICAZIONE: lo studente che supererà il Master Semipresenziale riceverà un certificato che attesterà il tirocinio svolto presso il centro in questione.

5. RAPPORTO DI LAVORO: il Master Semipresenziale non costituisce alcun tipo di rapporto lavorativo.

6. STUDI PRECEDENTI: alcuni centri potranno richiedere un certificato di studi precedenti per la partecipazione al Master Semipresenziale. In tal caso, sarà necessario esibirlo al dipartimento tirocini di TECH affinché venga confermata l'assegnazione del centro prescelto.

7. NON INCLUDE: il Master Semipresenziale non includerà nessun elemento non menzionato all'interno delle presenti condizioni. Pertanto, non sono inclusi alloggio, trasporto verso la città in cui si svolge il tirocinio, visti o qualsiasi altro servizio non menzionato.

Tuttavia, gli studenti potranno consultare il proprio tutor accademico per qualsiasi dubbio o raccomandazione in merito. Egli fornirà tutte le informazioni necessarie per semplificare le procedure.

08

Dove posso svolgere il Tirocinio?

Questo programma di Master Semipresenziale prevede nel suo itinerario un tirocinio pratico in una società di prestigio, dove lo studente metterà in pratica tutto ciò che ha imparato in materia di Intelligenza Artificiale nella Programmazione. In questo senso, e per avvicinare questa qualifica a un maggior numero di professionisti, TECH offre agli studenti l'opportunità di frequentarla in diversi centri in tutto il paese. In questo modo, l'istituzione rafforza il suo impegno per un'istruzione di qualità e a prezzi accessibili per tutti.





“

Affronterai sfide concrete, sviluppando soluzioni pratiche e collaborando in team, che miglioreranno le tue capacità tecniche e la tua capacità di risolvere problemi complessi”

tech 56 | Dove posso svolgere il Tirocinio?



Gli studenti potranno svolgere il tirocinio di questo Master Semipresenziale presso i seguenti centri:



Informatica

Captia Ingeniería

Paese	Città
Spagna	Madrid

Indirizzo: Av. de las Nieves, 37, Bloque A Planta 1
Oficina E, 28935, Móstoles, Madrid

Società di informatica dedicata a fornire soluzioni tecnologiche avanzate per le industrie

Tirocini correlati:

- Visual Analytics e Big Data
- Sviluppo di Software





“

Promuovi la tua carriera con un insegnamento olistico, che ti consente di avanzare sia a livello teorico che pratico”

09

Metodologia

Dieses Fortbildungsprogramm bietet eine andere Art des Lernens. Unsere Methodik wird durch eine zyklische Lernmethode entwickelt: **das Relearning**.

Dieses Lehrsystem wird z. B. an den renommiertesten medizinischen Fakultäten der Welt angewandt und wird von wichtigen Publikationen wie dem **New England Journal of Medicine** als eines der effektivsten angesehen.





Entdecken Sie Relearning, ein System, das das herkömmliche lineare Lernen hinter sich lässt und Sie durch zyklische Lehrsysteme führt: eine Art des Lernens, die sich als äußerst effektiv erwiesen hat, insbesondere in Fächern, die Auswendiglernen erfordern"

Fallstudie zur Kontextualisierung aller Inhalte

Unser Programm bietet eine revolutionäre Methode zur Entwicklung von Fähigkeiten und Kenntnissen. Unser Ziel ist es, Kompetenzen in einem sich wandelnden, wettbewerbsorientierten und sehr anspruchsvollen Umfeld zu stärken.

“

Mit TECH werden Sie eine Art des Lernens erleben, die an den Grundlagen der traditionellen Universitäten auf der ganzen Welt rüttelt"



Sie werden Zugang zu einem Lernsystem haben, das auf Wiederholung basiert, mit natürlichem und progressivem Unterricht während des gesamten Lehrplans.



Der Student wird durch gemeinschaftliche Aktivitäten und reale Fälle lernen, wie man komplexe Situationen in realen Geschäftsumgebungen löst.

Eine innovative und andersartige Lernmethode

Dieses TECH-Programm ist ein von Grund auf neu entwickeltes, intensives Lehrprogramm, das die anspruchsvollsten Herausforderungen und Entscheidungen in diesem Bereich sowohl auf nationaler als auch auf internationaler Ebene vorsieht. Dank dieser Methodik wird das persönliche und berufliche Wachstum gefördert und ein entscheidender Schritt in Richtung Erfolg gemacht. Die Fallmethode, die Technik, die diesem Inhalt zugrunde liegt, gewährleistet, dass die aktuellste wirtschaftliche, soziale und berufliche Realität berücksichtigt wird.

“ *Unser Programm bereitet Sie darauf vor, sich neuen Herausforderungen in einem unsicheren Umfeld zu stellen und in Ihrer Karriere erfolgreich zu sein* **”**

Die Fallmethode ist das am weitesten verbreitete Lernsystem an den besten Informatikschulen der Welt, seit es sie gibt. Die Fallmethode wurde 1912 entwickelt, damit Jurastudenten das Recht nicht nur auf der Grundlage theoretischer Inhalte erlernen. Sie bestand darin, ihnen reale komplexe Situationen zu präsentieren, damit sie fundierte Entscheidungen treffen und Werturteile darüber fällen konnten, wie diese zu lösen sind. Sie wurde 1924 als Standardlehrmethode in Harvard etabliert.

Was sollte eine Fachkraft in einer bestimmten Situation tun? Mit dieser Frage konfrontieren wir Sie in der Fallmethode, einer handlungsorientierten Lernmethode. Während des gesamten Kurses werden die Studenten mit mehreren realen Fällen konfrontiert. Sie müssen ihr gesamtes Wissen integrieren, recherchieren, argumentieren und ihre Ideen und Entscheidungen verteidigen.

Relearning Methodology

TECH kombiniert die Methodik der Fallstudien effektiv mit einem 100%igen Online-Lernsystem, das auf Wiederholung basiert und in jeder Lektion verschiedene didaktische Elemente kombiniert.

Wir ergänzen die Fallstudie mit der besten 100%igen Online-Lehrmethode: Relearning.

*Im Jahr 2019 erzielten wir die besten
Lernergebnisse aller spanischsprachigen
Online-Universitäten der Welt.*

Bei TECH lernen Sie mit einer hochmodernen Methodik, die darauf ausgerichtet ist, die Führungskräfte der Zukunft zu spezialisieren. Diese Methode, die an der Spitze der weltweiten Pädagogik steht, wird Relearning genannt.

Unsere Universität ist die einzige in der spanischsprachigen Welt, die für die Anwendung dieser erfolgreichen Methode zugelassen ist. Im Jahr 2019 ist es uns gelungen, die Gesamtzufriedenheit unserer Studenten (Qualität der Lehre, Qualität der Materialien, Kursstruktur, Ziele...) in Bezug auf die Indikatoren der besten spanischsprachigen Online-Universität zu verbessern.



In unserem Programm ist das Lernen kein linearer Prozess, sondern erfolgt in einer Spirale (lernen, verlernen, vergessen und neu lernen). Daher wird jedes dieser Elemente konzentrisch kombiniert. Mit dieser Methode wurden mehr als 650.000 Hochschulabsolventen mit beispiellosem Erfolg in so unterschiedlichen Bereichen wie Biochemie, Genetik, Chirurgie, internationales Recht, Managementfähigkeiten, Sportwissenschaft, Philosophie, Recht, Ingenieurwesen, Journalismus, Geschichte, Finanzmärkte und -instrumente fortgebildet. Dies alles in einem sehr anspruchsvollen Umfeld mit einer Studentenschaft mit hohem sozioökonomischem Profil und einem Durchschnittsalter von 43,5 Jahren.

Das Relearning ermöglicht es Ihnen, mit weniger Aufwand und mehr Leistung zu lernen, sich mehr auf Ihre Spezialisierung einzulassen, einen kritischen Geist zu entwickeln, Argumente zu verteidigen und Meinungen zu kontrastieren: eine direkte Gleichung zum Erfolg.

Nach den neuesten wissenschaftlichen Erkenntnissen der Neurowissenschaften wissen wir nicht nur, wie wir Informationen, Ideen, Bilder und Erinnerungen organisieren, sondern auch, dass der Ort und der Kontext, in dem wir etwas gelernt haben, von grundlegender Bedeutung dafür sind, dass wir uns daran erinnern und es im Hippocampus speichern können, um es in unserem Langzeitgedächtnis zu behalten.

Auf diese Weise sind die verschiedenen Elemente unseres Programms im Rahmen des so genannten Neurocognitive Context-Dependent E-Learning mit dem Kontext verbunden, in dem der Teilnehmer seine berufliche Praxis entwickelt.



Dieses Programm bietet die besten Lehrmaterialien, die sorgfältig für Fachleute aufbereitet sind:



Studienmaterial

Alle didaktischen Inhalte werden von den Fachleuten, die den Kurs unterrichten werden, speziell für den Kurs erstellt, so dass die didaktische Entwicklung wirklich spezifisch und konkret ist.

Diese Inhalte werden dann auf das audiovisuelle Format angewendet, um die Online-Arbeitsmethode von TECH zu schaffen. All dies mit den neuesten Techniken, die in jedem einzelnen der Materialien, die dem Studenten zur Verfügung gestellt werden, qualitativ hochwertige Elemente bieten.



Meisterklassen

Die Nützlichkeit der Expertenbeobachtung ist wissenschaftlich belegt.

Das sogenannte Learning from an Expert festigt das Wissen und das Gedächtnis und schafft Vertrauen für zukünftige schwierige Entscheidungen.



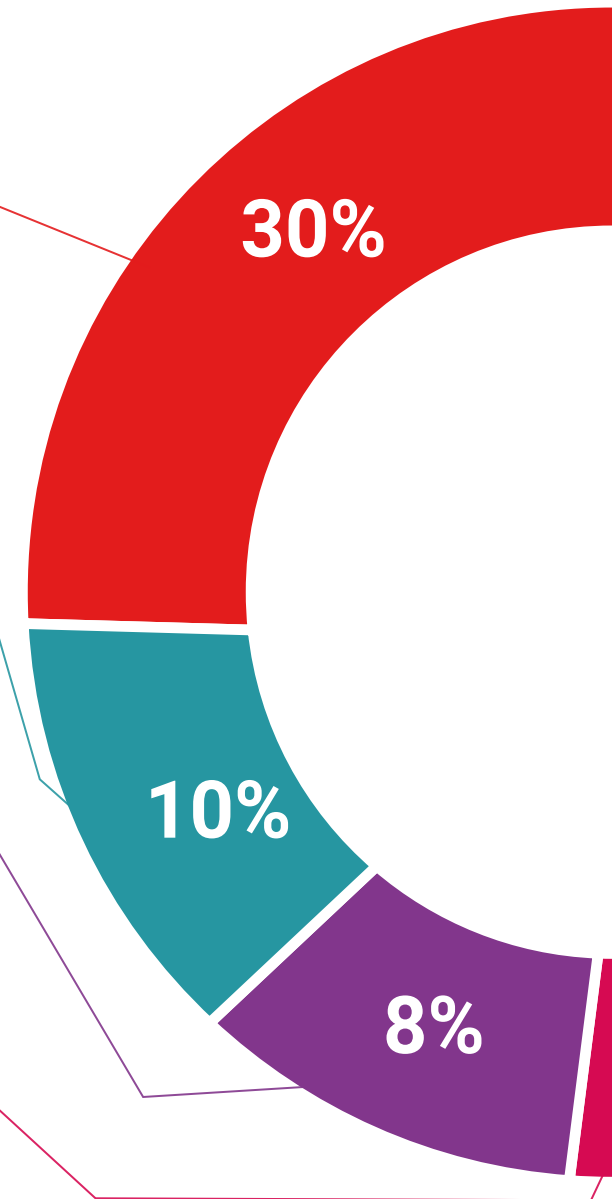
Übungen für Fertigkeiten und Kompetenzen

Sie werden Aktivitäten durchführen, um spezifische Kompetenzen und Fertigkeiten in jedem Fachbereich zu entwickeln. Übungen und Aktivitäten zum Erwerb und zur Entwicklung der Fähigkeiten und Fertigkeiten, die ein Spezialist im Rahmen der Globalisierung, in der wir leben, entwickeln muss.



Weitere Lektüren

Aktuelle Artikel, Konsensdokumente und internationale Leitfäden, u. a. In der virtuellen Bibliothek von TECH hat der Student Zugang zu allem, was er für seine Fortbildung benötigt.





Case Studies

Sie werden eine Auswahl der besten Fallstudien vervollständigen, die speziell für diese Qualifizierung ausgewählt wurden. Die Fälle werden von den besten Spezialisten der internationalen Szene präsentiert, analysiert und betreut.



Interaktive Zusammenfassungen

Das TECH-Team präsentiert die Inhalte auf attraktive und dynamische Weise in multimedialen Pillen, die Audios, Videos, Bilder, Diagramme und konzeptionelle Karten enthalten, um das Wissen zu vertiefen.

Dieses einzigartige Bildungssystem für die Präsentation multimedialer Inhalte wurde von Microsoft als "Europäische Erfolgsgeschichte" ausgezeichnet.



Testing & Retesting

Die Kenntnisse des Studenten werden während des gesamten Programms regelmäßig durch Bewertungs- und Selbsteinschätzungsaktivitäten und -übungen beurteilt und neu bewertet, so dass der Student überprüfen kann, wie er seine Ziele erreicht.



10

Titulación

Il Master Semipresenziale in Intelligenza Artificiale nella Programmazione garantisce, oltre alla preparazione più rigorosa e aggiornata, il conseguimento di una qualifica di Master Semipresenziale rilasciata da TECH Università Tecnologica.



“

Porta a termine questo programma e ricevi la tua qualifica universitaria senza spostamenti o fastidiose formalità”

Questo programma ti consentirà di ottenere il titolo di studio privato di **Master Semipresenziale in Chirurgia Veterinaria Mininvasiva degli Animali di Piccola Taglia** possiede il programma più completo e aggiornato del mercato.

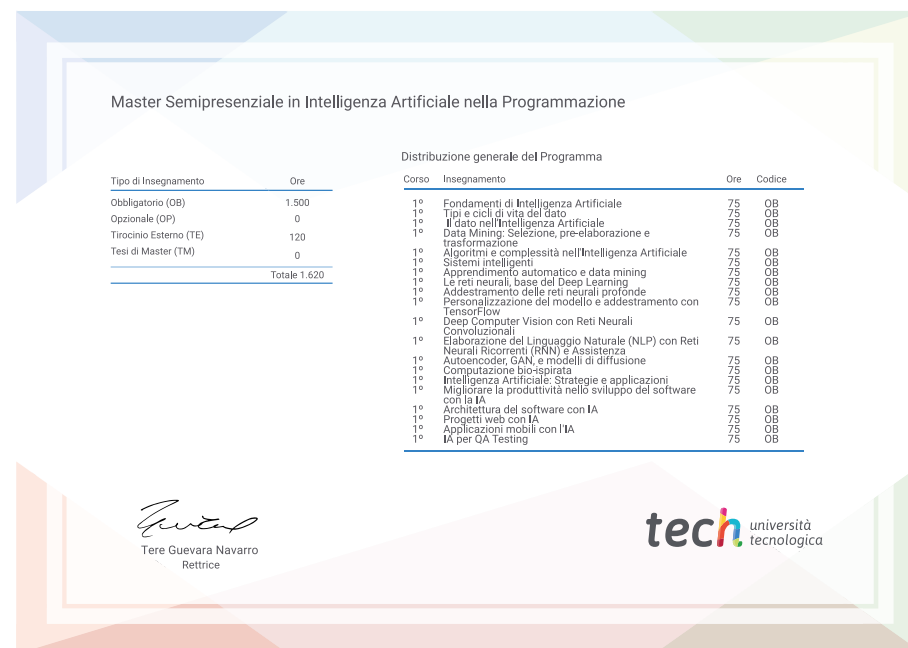
Dopo aver superato la valutazione, lo studente riceverà mediante lettera certificata* con ricevuta di ritorno, la sua corrispondente qualifica di **Master Semipresenziale** rilasciata da **TECH Università Tecnologica**.

Il titolo rilasciato da **TECH Università Tecnologica** esprime la qualifica ottenuta nel Corso Universitario, e riunisce tutti i requisiti comunemente richiesti da borse di lavoro, concorsi e commissioni di valutazione di carriere professionali.

Titolo: **Master Semipresenziale in Intelligenza Artificiale nella Programmazione**

Modalità: **Semipresenziale (Online + Tirocinio)**

Durata: **12 mesi**



*Apostille dell'Aia. Se lo studente dovesse richiedere che il suo diploma cartaceo sia provvisto di Apostille dell'Aia, TECH EDUCATION effettuerà le gestioni opportune per ottenerla pagando un costo aggiuntivo.



Master Semipresenziale Intelligenza Artificiale nella Programmazione

Modalità: Semipresenziale (Online + Tirocinio)

Durata: 12 mesi

Titolo: TECH Università Tecnologica

Crediti: 60 + 4 ECTS

Master Semipresenziale Intelligenza Artificiale nella Programmazione

