



1506
**UNIVERSITÀ
DEGLI STUDI
DI URBINO
CARLO BO**

UNIVERSITÀ DEGLI STUDI DI URBINO CARLO BO

Dipartimento di Scienze Pure e Applicate
Corso di Laurea in Informatica Applicata

Tesi di Laurea

**INTELLIGENZA ARTIFICIALE E SALUTE:
ADDESTRAMENTO DI LLM PER IL
SUPPORTO DEL PAZIENTE IPERTESO**

Relatore:
Chiar.ma Prof.ssa Sara Montagna

Candidato:
Giulia Costa

Correlatore:
Chiar.mo Prof. Stefano Ferretti
Chiar.mo Ing. Matteo Magnini

Anno Accademico 2023-2024

Alla mia famiglia, per avermi supportato in questa importante scelta di
cambiamento di vita, e al mio compagno Leandro, per il suo costante
sostegno nei momenti più impegnativi.

Indice

Introduzione	1
1 Introduzione al mondo dell'Intelligenza Artificiale	4
1.1 Che cos'è l'Intelligenza Artificiale	4
1.2 Fondamenti e origini dell'Intelligenza Artificiale	5
1.3 Crescita e impatto dell'Intelligenza Artificiale	5
1.4 Gli ambiti di ricerca dell'Intelligenza Artificiale	6
1.5 Il Machine Learning	7
1.5.1 Il Deep Learning	7
1.5.2 Le tecniche di Machine Learning	7
1.5.3 Gli algoritmi di Machine Learning	8
1.5.4 Le reti neurali	9
2 Il Natural Language Processing (NLP)	11
2.1 Introduzione al Natural Language Processing	12
2.1.1 Il Natural Language Understanding (NLU)	13
2.1.2 Il Natural Language Generation (NLG)	13
2.2 Il ruolo del Machine Learning e del Deep Learning nell'NLP . .	14
2.2.1 Apprendimento supervisionato nell'NLP	14
2.2.2 Machine Learning nell'apprendimento supervisionato . .	14
2.2.3 Deep Learning nell'apprendimento supervisionato	16
2.3 Applicazioni del NLP nell'AI	17
3 I Large Language Models	18
3.1 Introduzione agli LLM	19
3.2 Modelli Proprietari VS modelli Open Source	20
3.2.1 Modelli proprietari	20
3.2.2 Modelli Open Source	25
3.2.3 Confronto tra modelli Proprietari e Open Source:	26
3.3 Implicazioni etiche e sociali	27
4 L'impiego degli LLM nelle applicazioni mediche	28
4.1 Innovazione tecnologica degli LLM	28
4.2 Benchmarking e confronto tra modelli	29

4.3	Da PaLM a Med-PaLM: innovazioni e sfide nell'ambito medico.	30
4.4	Applicazioni degli LLM nel settore sanitario	31
4.4.1	Supporto alla pratica clinica	32
4.4.2	Scrittura scientifica	32
4.4.3	Educazione medica	32
5	Ottimizzazione di LLM per l'ipertensione: Fine-Tuning con risorse Open Source	33
5.1	I Chatbot come strumento di supporto per il paziente iperteso	34
5.2	Creazione del Dataset	35
5.2.1	Implementazione dello Script	35
5.3	Caricamento e preparazione del Dataset	36
5.3.1	Data Augmentation	36
5.3.2	Suddivisione del Dataset	37
5.3.3	Analisi del Dataset	37
5.4	Caricamento del Modello	38
5.4.1	Il Tokenizer	38
5.4.2	Quantizzazione del modello	39
5.4.3	Caricamento del Modello Quantizzato	40
5.5	Ottimizzazione della memoria	41
5.5.1	Tecniche di ottimizzazione utilizzate	41
5.6	Fine-Tuning con Ludwig	42
5.6.1	Tecniche usate durante il Fine-Tuning	42
5.7	Valutazione del modello	44
5.8	Analisi dei risultati per modello	47
5.8.1	GPT-2	47
5.8.2	Mistral-7B	54
5.8.3	BioMistral-7B	61
5.8.4	Llama-8B	67
5.9	Discussione dei risultati	68
6	Conclusione	73
	Bibliografia	75
	Ringraziamenti	78

Introduzione

L'intelligenza artificiale, con i suoi numerosi campi di ricerca, è oggi una delle tecnologie più innovative, con applicazioni che spaziano dalla medicina all'industria.

Il primo obiettivo di questa tesi è stato quello di comprendere meglio il vasto mondo dell'intelligenza artificiale, concentrandosi sul ruolo fondamentale che questa tecnologia può svolgere nel settore medico. In particolare, l'obiettivo è stato quello di fare *fine-tuning* su modelli di linguaggio di grandi dimensioni (Large Language Model - LLM), allo scopo di migliorarne le performance in un contesto specifico, ovvero per la gestione dell'ipertensione, rendendoli capaci di rispondere a domande mediche specifiche e integrabili in chatbot.

La sfida principale è stata bilanciare i vantaggi delle risorse open, che includono sia modelli open source che le tecnologie computazionali necessarie per l'addestramento e l'esecuzione, con le esigenze legate alle elevate risorse hardware necessarie per operare su larga scala.

Sebbene l'adozione di queste risorse consenta una maggiore tutela della privacy, grazie alla possibilità di eseguire i modelli localmente, e comporti una riduzione dei costi, rimane comunque il limite legato alla necessità di infrastrutture computazionali sufficientemente potenti per gestire modelli di grandi dimensioni.

Il percorso intrapreso si è articolato in una serie di fasi, sia teoriche che pratiche.

L'approfondimento teorico ha inizialmente riguardato il concetto di intelligenza artificiale e le sue principali applicazioni. Si è iniziato esplorando i fondamenti dell'intelligenza artificiale e la sua evoluzione storica, per poi concentrarsi sul ruolo centrale del Machine Learning e delle reti neurali. Sono state approfondite le tecniche di apprendimento automatico, che permettono ai modelli di apprendere dai dati e migliorare progressivamente le loro prestazioni, riducendo l'errore nelle predizioni.

A seguire, ci si è concentrati sul Natural Language Processing (NLP), una delle aree più affascinanti e complesse dell'intelligenza artificiale, dedicata all'interazione tra computer e linguaggio umano. In questo ambito, è stato esaminato come i modelli avanzati, con particolare attenzione ai LLM, siano in grado di comprendere e generare linguaggio naturale, affrontando compiti

complessi come la generazione e l'elaborazione del linguaggio umano. Grazie a tecniche di apprendimento semi-supervisionato e non supervisionato, questi modelli migliorano costantemente la qualità delle risposte. In questo contesto, è stata inoltre analizzata la differenza tra modelli open source e modelli proprietari, valutandone i rispettivi vantaggi e svantaggi in termini di costi, trasparenza, personalizzazione e supporto tecnico.

Particolare attenzione è stata rivolta all'impiego degli LLM nel settore medico, dove l'accuratezza e l'affidabilità delle risposte sono essenziali. Studi scientifici hanno dimostrato che questi modelli possono fornire un valido supporto nella gestione delle patologie, migliorando sia la gestione delle informazioni che l'assistenza clinica.

Questa analisi preliminare ha fornito una solida base teorica, fondamentale per guidare la successiva fase sperimentale del progetto. In questa fase pratica, il focus principale è stato sull'utilizzo di risorse open source per il fine-tuning di LLM open, con l'obiettivo di adattarli ad un compito specifico: la gestione dell'ipertensione.

Il fine-tuning consente di adattare modelli generici a un dominio specifico utilizzando dataset mirati. In questo caso, è stata applicata questa tecnica a modelli come GPT-2, Mistral7B e BioMistral7B, utilizzando risorse open source come Google Colab, Hugging Face, Ludwig e PyTorch.

Il progetto ha seguito un approccio pratico, che ha incluso la creazione e preparazione del dataset, l'ottimizzazione dei modelli e la valutazione delle performance. Metriche come *loss*, *perplexity*, *token accuracy* e *BERTScore* sono state impiegate per misurare l'efficacia dei modelli nella generazione di risposte accurate e pertinenti.

I risultati hanno messo in evidenza i vantaggi delle risorse open source, ma anche le loro limitazioni. Sebbene vi siano stati miglioramenti nelle prestazioni, in particolare nel BERTScore per i modelli Mistral e BioMistral, i risultati non sono stati soddisfacenti per quanto riguarda *loss*, *perplexity* e *token accuracy*.

È cruciale considerare che i modelli utilizzati richiedono risorse computazionali molto diverse. Ad esempio, *GroNLP/gpt2-small-italian* richiede circa 2-4 GB di RAM per essere caricato in memoria, un requisito facilmente gestibile anche su piattaforme come Google Colab nella sua versione gratuita. Tuttavia, modelli più complessi come Mistral7B, che necessitano di 30-40 GB di RAM, pongono sfide significative, soprattutto in termini di memoria e potenza di calcolo necessarie prima di eseguire il fine-tuning.

Il confronto con le risorse disponibili su Google Colab mostra che, sebbene la versione gratuita offra fino a 12 GB di RAM e l'accesso a GPU T4 con 16 GB di VRAM, queste risorse possono risultare insufficienti per modelli di grandi dimensioni come Mistral7B. Questo accentua la complessità del problema, rendendo necessario ottimizzare l'uso delle risorse tramite tecniche come la quantizzazione e l'offloading, oppure l'adozione di hardware più potente, al

fine di gestire adeguatamente l'enorme quantità di dati e memoria richiesta per garantire risultati ottimali.

In conclusione, questo studio ha dimostrato che, sebbene le risorse open siano una valida alternativa per sviluppare soluzioni accessibili e scalabili, esse presentano delle limitazioni quando si tratta di affrontare compiti complessi, come quelli nel settore medico. Di conseguenza, risorse più avanzate, non open source, potrebbero essere necessarie per ottenere prestazioni ottimali e rispondere alle sfide poste dall'accuratezza e dalla sicurezza delle risposte in contesti clinici.

Capitolo 1

Introduzione al mondo dell'Intelligenza Artificiale

In questo capitolo parleremo dell'Intelligenza Artificiale (AI), cercando di capire che cos'è, quali siano le sue origini, come si sta evolvendo e quali siano gli ambiti di ricerca principali.

L'intelligenza artificiale è una delle tecnologie più innovative e trasformative del nostro tempo, con applicazioni che spaziano dalla ricerca scientifica alla vita quotidiana.

Partiremo da una definizione generale, passando per i fondamenti teorici e storici, fino ad arrivare alla sua crescita esponenziale. Approfondiremo gli ambiti di ricerca più rilevanti dell'Intelligenza Artificiale, soffermandoci sul Machine Learning e sulle tecniche principali utilizzate, spiegando come ciascuna di esse contribuisca all'evoluzione dei modelli AI. Infine, ci concentreremo sulle reti neurali, analizzando le diverse tipologie e il loro ruolo, fondamentale nel progresso dell'Intelligenza Artificiale. Questo capitolo ci fornirà una panoramica completa, preparandoci ad approfondire ulteriormente le tecniche e le applicazioni specifiche nell'ambito di ricerca del NLP.

1.1 Che cos'è l'Intelligenza Artificiale

L'intelligenza artificiale è un ramo dell'informatica che si occupa di creare sistemi in grado di simulare l'intelligenza umana. I compiti che possono essere svolti da questi sistemi sono molteplici e, in combinazione con altre tecnologie, l'AI può eseguire compiti quali: ragionare, apprendere, risolvere problemi, pianificare e comunicare, cercando di emulare quindi le funzioni cognitive umane.

Come affermava il pioniere dell'informatica Alan Turing:

"Se una macchina si comporta in tutti i modi in cui si comporta un essere intelligente, allora si deve considerare intelligente". [22]

1.2 Fondamenti e origini dell'Intelligenza Artificiale

Le origini dell'Intelligenza Artificiale risiedono nella logica e nella matematica, le quali hanno fornito un quadro formale per ragionare sui concetti e sulle relazioni tra di essi, essenziali per la creazione di modelli e algoritmi computazionali.

Un punto di svolta fondamentale fu rappresentato dalle macchine di Turing, progettate per simulare qualsiasi algoritmo computazionale, fornendo le basi teoriche per il calcolo automatico e ponendo le fondamenta per la creazione di macchine capaci di eseguire compiti di elaborazione delle informazioni.

John von Neumann contribuì significativamente a questo sviluppo, lavorando sulla teoria della programmazione e introducendo il concetto di memoria ad accesso casuale (RAM), essenziale per i primi computer.

Un altro passo significativo fu il progetto ENIAC, uno dei primi computer elettronici su larga scala, che dimostrò il potenziale dei computer per eseguire calcoli molto più rapidi ed efficienti, rispetto ai metodi precedenti, contribuendo allo sviluppo delle tecnologie che condussero poi alla nascita dell'AI.

Nonostante le sue radici antiche, la nascita formale dell'AI come campo scientifico si può far risalire al 1956, con il Dartmouth Conference, organizzato da John McCarthy, Marvin Minsky, Nathaniel Rochester e Claude Shannon. Questo incontro segnò l'inizio ufficiale della ricerca sull'intelligenza artificiale, con McCarthy che definì l'AI come "la scienza e l'ingegneria di creare macchine intelligenti" [13], inaugurando una nuova era di innovazione e scoperte scientifiche.

1.3 Crescita e impatto dell'Intelligenza Artificiale

Dalla sua nascita, l'intelligenza artificiale ha vissuto una crescita esponenziale, alimentata dall'aumento della potenza computazionale e dalla disponibilità di enormi quantità di dati. Questo sviluppo ha portato a progressi significativi non solo nella teoria e negli algoritmi, ma anche nelle applicazioni che oggi influenzano molti aspetti della vita quotidiana.

L'intelligenza artificiale non è solo un campo accademico, ma una forza trainante dell'innovazione tecnologica e del cambiamento sociale.

Andrew Ng, un leader nel campo dell'AI, ha affermato:

"L'intelligenza artificiale è la nuova elettricità" [17], sottolineando in tal modo l'impatto rivoluzionario dell'AI, paragonandolo a quello che l'elettricità ha avuto sull'industria e sulla società.

1.4 Gli ambiti di ricerca dell'Intelligenza Artificiale

La ricerca nell'Intelligenza Artificiale è vasta ed in continua evoluzione, abbracciando numerose aree che si ampliano costantemente grazie ai rapidi progressi del settore. Tra le aree di ricerca più attive e rilevanti possiamo includere:

- *Machine Learning (Apprendimento Automatico)*: questa è una delle aree più dinamiche dell'AI, il cui obiettivo è sviluppare algoritmi e modelli in grado di migliorare le proprie prestazioni apprendendo dai dati. Le tecniche più importanti includono l'apprendimento supervisionato, non supervisionato e per rinforzo, ciascuno con applicazioni in vari campi: dalla previsione dei mercati finanziari, alla raccomandazione di contenuti su piattaforme come Netflix, fino alla classificazione di immagini e dati per compiti come il riconoscimento facciale o la diagnostica medica.
- *Elaborazione del Linguaggio Naturale (NLP)*: questo ambito si occupa della capacità delle macchine di comprendere, analizzare e generare linguaggio umano. Applicazioni tipiche includono la traduzione automatica, i chatbot, il riassunto automatico e l'analisi del sentiment.
- *Computer Vision (Visione Artificiale)*: questa area permette ai computer di analizzare e comprendere le immagini ed i video, replicando le capacità visive umane. È alla base di tecnologie come il riconoscimento facciale, l'identificazione di oggetti e l'elaborazione delle immagini mediche. Viene utilizzata in settori come la sicurezza, la diagnostica radiologica e i veicoli a guida autonoma, dove la capacità di interpretare visivamente l'ambiente è fondamentale per la navigazione e la sicurezza.
- *Robotica*: la robotica è un campo che si occupa della progettazione, costruzione e controllo di robot fisici, spesso combinando AI per permettere loro di eseguire compiti in modo autonomo. I robot vengono utilizzati in settori come la manifattura, dove l'automazione è cruciale per la produttività, così come nella chirurgia robotica e nell'agricoltura di precisione, dove la robotica sta rivoluzionando le modalità di lavoro grazie alla capacità di automatizzare operazioni come semina, raccolta e monitoraggio delle colture, migliorando l'efficienza e riducendo l'uso di risorse.
- *Sistemi Esperti*: questi sistemi utilizzano conoscenze specifiche di un dominio per risolvere problemi complessi, simulando la capacità decisionale di un esperto umano. I sistemi esperti sono utilizzati in ambiti come la diagnostica medica, per supportare i medici nella scelta delle terapie, e nella consulenza finanziaria, dove possono fornire raccomandazioni basate su analisi complesse dei mercati.

1.5 Il Machine Learning

Come abbiamo visto poco fa, il Machine Learning si occupa dello sviluppo di algoritmi e modelli capaci di apprendere dai dati e di migliorare le loro prestazioni nel tempo, senza essere esplicitamente programmati per ogni specifico compito. Questi sistemi sono in grado di estrarre informazioni dai dati di input, al fine di individuare modelli e relazioni significative, per poi utilizzare tali conoscenze in modo da prendere decisioni o effettuare previsioni autonomamente. Tecniche come le reti neurali artificiali e l'apprendimento profondo (deep learning) hanno rivoluzionato il campo dell'AI.

1.5.1 Il Deep Learning

Il Deep Learning è una sotto-disciplina dell'apprendimento automatico che utilizza reti neurali multilivello, chiamate reti neurali profonde, per simulare il complesso potere decisionale del cervello umano e apprendere rappresentazioni complesse dei dati. L'utilizzo del deep learning è diventato particolarmente diffuso negli ultimi anni, grazie ai significativi progressi nella potenza di calcolo ed alla disponibilità di grandi quantità di dati di addestramento. Nel campo del NLP, le reti neurali sono impiegate per:

- *Interpretare il linguaggio naturale:* le reti neurali sono in grado di apprendere rappresentazioni complesse del linguaggio naturale, consentendo la comprensione di testi articolati e sofisticati.
- *Generare testo coerente:* le reti neurali possono essere utilizzate per produrre testi realistici e coerenti, come articoli di blog, e-mail e poesie.

1.5.2 Le tecniche di Machine Learning

Esistono diverse tecniche di apprendimento automatico, tra cui l'apprendimento supervisionato, non supervisionato, semi-supervisionato e per rinforzo.

- **Apprendimento supervisionato.** Parliamo di una delle tecniche fondamentali nel campo del machine learning che svolge anche un ruolo cruciale nel Natural Language Processing. Questo metodo prevede l'addestramento di un modello su un dataset etichettato, dove ogni esempio di input è associato a un output desiderato. Durante il processo di apprendimento, il modello impara a mappare gli input agli output corretti rilevando pattern nei dati di addestramento. Esempi comuni sono la classificazione, che assegna etichette a categorie, e la regressione, che prevede invece valori numerici.

- **Apprendimento non supervisionato.** A differenza dell'apprendimento supervisionato, quello non supervisionato, lavora con dati non etichettati. L'obiettivo è identificare pattern o strutture nascoste nei dati senza un output predefinito, riconoscendo autonomamente somiglianze, differenze o relazioni tra i dati. Questo tipo di apprendimento viene spesso utilizzato per il clustering, l'analisi delle componenti principali (PCA), l'esplorazione della struttura dei dati e l'identificazione di anomalie.
- **Apprendimento semi-supervisionato.** L'apprendimento semi-supervisionato rappresenta un punto intermedio tra l'apprendimento supervisionato e quello non supervisionato. Viene utilizzato quando si dispone di un piccolo insieme di dati etichettati e di un grande insieme di dati non etichettati. Questo approccio sfrutta i pochi esempi etichettati per migliorare la capacità del modello di classificare o fare previsioni anche sui dati non etichettati.
- **Apprendimento per rinforzo.** Questa tecnica si basa sullo sviluppo di agenti che migliorano le loro performance basandosi su feedback, sotto forma di "ricompense" o "punizioni". Viene impiegata in applicazioni come i sistemi di dialogo interattivo, dove il modello impara a fare scelte ottimali nel corso delle conversazioni.

1.5.3 Gli algoritmi di Machine Learning

Alcuni degli algoritmi più noti nel campo dell'apprendimento automatico sono:

- **Support Vector Machines (SVM).** Questo algoritmo si basa sull'idea di trovare un iperpiano che separa i dati in due classi nel miglior modo possibile, massimizzando la distanza tra le categorie stesse.
- **Naive Bayes.** Si tratta di un classificatore probabilistico basato sul teorema di Bayes e presuppone che le diverse caratteristiche di un dato siano indipendenti tra loro. Questo approccio rende il calcolo delle probabilità più facile e veloce, rendendo il modello particolarmente efficace in scenari in cui ci sono molte caratteristiche da considerare.
- **Decision Trees.** Utilizza una struttura ad albero per prendere decisioni basate sui valori delle caratteristiche. In questa struttura, ogni nodo rappresenta una caratteristica, ogni ramo corrisponde ad una scelta, e ogni foglia indica un risultato finale.
- **K-Nearest Neighbors (KNN).** Si tratta di un algoritmo di classificazione o regressione che confronta un nuovo dato con i simili più "vicini" in un dataset già conosciuto. Calcola una "distanza", che indica il grado di similarità tra i dati, e assegna la classe a cui appartiene la maggior parte dei vicini più prossimi.

1.5.4 Le reti neurali

Le reti neurali giocano un ruolo cruciale nell'apprendimento automatico.

Si tratta di una classe di algoritmi ispirati alla struttura e al funzionamento del cervello umano, progettati per apprendere e risolvere problemi complessi attraverso l'imitazione dei processi cognitivi. Questi algoritmi sono composti da unità di calcolo interconnesse chiamati "neuroni artificiali", progettati per riconoscere pattern complessi nei dati.

I neuroni sono organizzati in tre tipi di strati principali: uno strato di input, che riceve i dati grezzi; uno o più strati nascosti, che elaborano e trasformano questi dati attraverso calcoli più complessi; e uno strato di output, che fornisce la predizione o la classificazione finale.

Durante l'addestramento, le reti neurali ottimizzano i pesi delle connessioni tra i neuroni, migliorando progressivamente le loro prestazioni e sviluppando la capacità di apprendere dalle esperienze passate e di generalizzare a nuove situazioni.

Questo rende le reti neurali estremamente potenti per una vasta gamma di applicazioni, tra cui il riconoscimento di immagini, la traduzione automatica, e la generazione di testi. Tuttavia, il loro successo dipende in gran parte dalla qualità e dalla quantità dei dati utilizzati per l'addestramento.

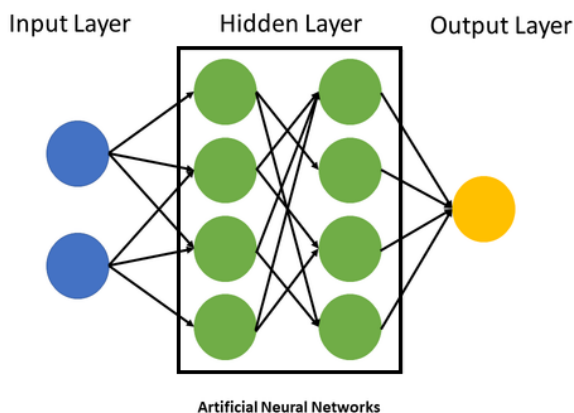


Figura 1.1: **Rappresentazione di una rete neurale**

Notiamo nell'immagine la suddivisione di una rete neurale in tre strati principali: lo strato di input (blu) che riceve i dati iniziali, lo strato nascosto (verde) che elabora le informazioni attraverso neuroni interconnessi, e lo strato di output (giallo) che produce la previsione finale.

Esistono vari tipi di reti neurali, ciascuna con caratteristiche specifiche che le rendono adatte a diverse applicazioni:

- **Reti Neurali Convoluzionali (CNN):** sono particolarmente efficaci nell'elaborazione di dati visivi, come immagini o video, diventate lo standard per molte applicazioni di computer vision. Utilizzano strati convoluzionali per analizzare piccole porzioni di dati alla volta, chiamate *recettive fields*. Questi strati applicano dei filtri per estrarre caratteristiche specifiche, come bordi, texture o colori, che aiutano a costruire una rappresentazione gerarchica dell'immagine.
- **Reti Neurali Ricorrenti (RNN):** si tratta di una classe di reti neurali particolarmente adatte all'elaborazione di dati sequenziali, come testi, serie temporali o dati audio. A differenza delle reti neurali tradizionali, che trattano ogni input in modo indipendente, le RNN possono mantenere una memoria interna che consente di utilizzare informazioni provenienti da input precedenti per influenzare le previsioni future.
- **Long Short-Term Memory (LSTM):** sono una variante delle reti neurali ricorrenti, progettate per superare le limitazioni delle RNN standard nella gestione delle dipendenze a lungo termine. Le RNN tradizionali tendono a dimenticare informazioni importanti quando lavorano con sequenze lunghe, il che rende difficile mantenere contesti rilevanti durante il processo di apprendimento. Le LSTM risolvono questi problemi grazie ad una struttura composta da "celle di memoria" e tre gate (di ingresso, uscita e dimenticanza) che regolano quali informazioni mantenere o eliminare. Questa architettura rende gli LSTM ideali per applicazioni come il riconoscimento del linguaggio e la traduzione automatica.
- **Transformers:** utilizzano meccanismi di *self-attention* ¹ per catturare relazioni a lungo termine nei dati, permettendo al modello di dare un "peso" diverso a ciascun elemento della sequenza, consentendogli di comprendere meglio il contesto generale. A differenza delle RNN o LSTM, i Transformers processano l'intera sequenza in parallelo, valutando il contesto complessivo simultaneamente. Questo approccio non solo accelera la velocità di addestramento, ma migliora anche la comprensione del significato generale e delle dipendenze a lungo termine tra le parole.

¹La self-attention è un meccanismo utilizzato per migliorare la capacità di elaborare sequenze di dati, come testi o immagini, in modo più efficace. L'idea principale è che ogni elemento di una sequenza (ad esempio, una parola in una frase) "presta attenzione" a tutti gli altri elementi della stessa sequenza per catturare le relazioni tra di essi.

Capitolo 2

Il Natural Language Processing (NLP)

In questo capitolo tratteremo uno dei sottocampi più affascinanti dell'intelligenza artificiale: il Natural Language Processing (NLP).

Eploreremo il ruolo del Machine Learning e del Deep Learning nell'NLP, evidenziando come queste tecniche avanzate consentano ai modelli di apprendere dai dati e migliorare continuamente le loro prestazioni.

Approfondiremo l'apprendimento supervisionato nel contesto del NLP e del Natural Language Generation (NLG). Inoltre, esamineremo il Natural Language Understanding (NLU), una componente fondamentale dell'NLP.

Discuteremo degli algoritmi più comuni utilizzati in questo ambito, senza dimenticare il ruolo cruciale delle reti neurali, citandone alcune delle più importanti e innovative.

Infine, discuteremo delle applicazioni pratiche del NLP nell'intelligenza artificiale, con particolare attenzione agli assistenti virtuali, ai chatbot, alla traduzione automatica e all'analisi del sentiment.

Concluderemo con una panoramica sui Large Language Models (LLM), esplorando le loro caratteristiche, le sfide etiche e sociali che comportano, ed il loro impatto futuro.

2.1 Introduzione al Natural Language Processing

Il NLP rappresenta un campo interdisciplinare, un'intersezione tra informatica, intelligenza artificiale e linguistica, il cui obiettivo è quello di permettere ai computer di interpretare, generare e comprendere il linguaggio umano in maniera utile e significativa. Questa disciplina si avvale di algoritmi e modelli computazionali per svolgere compiti che vanno dalla semplice conversione di parole in dati strutturati ad interazioni complesse che simulano la comprensione umana. Le tecniche impiegate nel NLP comprendono l'analisi sintattica, la semantica, il riconoscimento del discorso e l'apprendimento automatico, con applicazioni che spaziano in molti settori. Queste tecnologie non solo migliorano l'efficienza delle operazioni ma aprono anche nuove frontiere nella maniera in cui interagiamo con le macchine, influenzando significativamente l'avanzamento tecnologico.

Come afferma il pioniere dell'informatica John McCarthy,
"L'intelligenza artificiale è la scienza di come far fare alle macchine cose che, di solito, richiedono l'intelligenza umana." [12]

Il NLP rappresenta un passo fondamentale in questa direzione, consentendo alle macchine di comunicare con noi in modo naturale ed intuitivo.

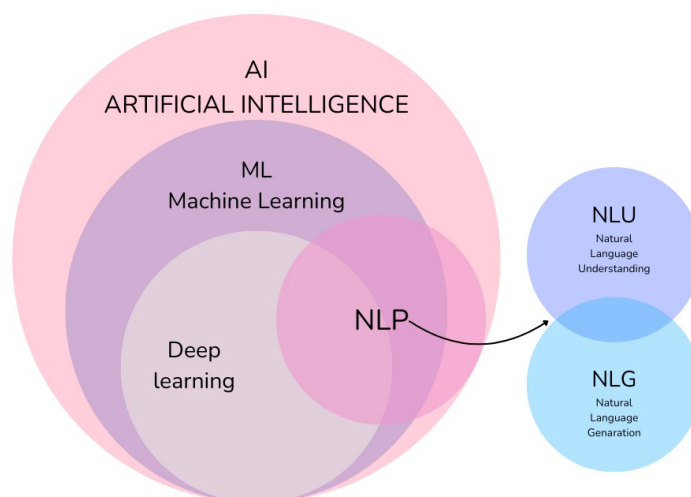


Figura 2.1: **Diagramma di Venn: relazioni tra i diversi campi dell'AI.** Il diagramma mostra AI come l'insieme più ampio che include ML e Deep Learning. Il NLP è rappresentato all'interno di ML e Deep Learning, mentre NLU e NLG sono sottoinsiemi specifici del NLP.

2.1.1 Il Natural Language Understanding (NLU)

Un altro aspetto fondamentale del NLP è la Comprensione del Linguaggio Naturale (NLU), un sottocampo che si concentra sulla capacità delle macchine di interpretare e comprendere il linguaggio umano in modo significativo e contestuale. Il NLU consente ai sistemi di analizzare il testo, identificando entità, nominate "relazioni semantiche" e "intenzioni dell'utente". Questo processo coinvolge diverse fasi, tra cui l'analisi sintattica, l'analisi semantica e la disambiguazione del significato. Attraverso tecniche avanzate di Machine Learning e Deep Learning, come i *word embeddings*¹ e le reti neurali, i sistemi di NLU possono decodificare il contesto e il contenuto delle informazioni, migliorando la loro capacità di rispondere accuratamente alle domande, estrarre informazioni ed interagire in modo più naturale con gli utenti.

2.1.2 Il Natural Language Generation (NLG)

Uno degli aspetti chiave del NLP è la Generazione del Linguaggio Naturale (NLG), un sottocampo che si concentra sulla produzione automatica di testo in linguaggio umano. Il NLG consente ai sistemi di creare contenuti scritti o parlati, che vanno dai semplici riassunti di dati strutturati a narrazioni complesse. Questo processo coinvolge diverse fasi, tra cui la determinazione del contenuto, la strutturazione del documento, l'aggregazione delle frasi, la lessicalizzazione e la *realizzazione superficiale*². Grazie a questo approccio, i computer possono generare risposte nei chatbot, creare sommari automatizzati e molto altro, migliorando significativamente la nostra interazione con le macchine.

¹I word embeddings sono rappresentazioni numeriche delle parole che catturano il loro significato semantico. Invece di trattare le parole come singole entità, i word embeddings le rappresentano come vettori in uno spazio multidimensionale. In questo spazio, parole con significati simili sono vicine tra loro.

²Realizzazione superficiale (o surface realization in inglese) è il processo finale nella generazione del linguaggio naturale (NLG) in cui una rappresentazione strutturata di contenuto viene trasformata in un testo fluente e grammaticalmente corretto, proprio come farebbe un essere umano.

2.2 Il ruolo del Machine Learning e del Deep Learning nell’NLP

Nel Natural Language Processing, il Machine Learning e il Deep Learning sono tecniche fondamentali per sviluppare modelli capaci di interpretare, generare e comprendere il linguaggio umano. Queste tecnologie utilizzano diverse tecniche di apprendimento per migliorare continuamente le prestazioni dei modelli. Tra queste, l’apprendimento supervisionato è particolarmente importante.

2.2.1 Apprendimento supervisionato nell’NLP

Nel contesto dell’elaborazione del linguaggio naturale, l’apprendimento supervisionato è fondamentale, poichè permette di addestrare modelli in grado di risolvere compiti complessi e specifici con un’alta precisione. Il principio base dell’apprendimento supervisionato è che il modello apprende dai dati etichettati, cioè esempi in cui ogni input (come un testo) è associato ad un output desiderato (come una categoria).

Nel contesto dell’NLP, i compiti più comuni che vengono affrontati con l’apprendimento supervisionato includono la classificazione del testo, il riconoscimento di entità nominate, la traduzione automatica, il riconoscimento vocale.

2.2.2 Machine Learning nell’apprendimento supervisionato

Il Machine Learning è il motore che alimenta i progressi più importanti nell’elaborazione del linguaggio naturale. L’obiettivo nell’NLP è quello di addestrare modelli capaci di generalizzare, cioè di applicare ciò che hanno imparato da un dataset ad altri dati simili, ma mai visti prima.

L’apprendimento automatico gioca un ruolo centrale nel migliorare le applicazioni NLP, tra cui:

- **Addestramento e generazione del linguaggio:** i modelli linguistici sono in grado di apprendere le relazioni tra parole e frasi nei dati di addestramento. Attraverso tecniche di Machine Learning, questi modelli non solo catturano il contesto e le sfumature semantiche delle parole, ma generano anche testo fluente e coerente, migliorando la qualità delle previsioni e delle risposte generate in applicazioni come la generazione automatica di testo (NLG).
- **Sviluppo di applicazioni NLP e riconoscimento del parlato:** il Machine Learning consente la creazione di applicazioni potenti, come i chatbot, gli assistenti virtuali (come Siri e Alexa), e i sistemi di traduzione automatica. Queste applicazioni utilizzano modelli addestrati per comprendere il linguaggio naturale e convertire il parlato in testo scritto, facilitando così l’interazione uomo-macchina.

Algoritmi di Machine Learning utilizzati nell’NLP

Nel capitolo precedente abbiamo trattato in maniera generica la definizione degli algoritmi utilizzati nel machine learning. Ora ci concentreremo sul loro utilizzo specifico nel contesto del NLP.

- **Support Vector Machines (SVM):**

sono algoritmi utilizzati principalmente per problemi di classificazione e regressione. che richiedono alta precisione, come la categorizzazione di documenti, la classificazione delle email di spam o il rilevamento di fake news, in cui è cruciale trovare una separazione accurata tra le classi.

- **Naïve Bayes:**

questo classificatore probabilistico calcola la probabilità che un dato appartenga a una certa classe, date alcune caratteristiche osservate ipotizzando che tutte le caratteristiche siano indipendenti tra loro. Viene per questo utilizzato per compiti come l’analisi del sentiment (determinare se un testo esprime un’opinione positiva o negativa) o la categorizzazione di articoli (assegnare una notizia a una categoria specifica). Grazie alla sua efficienza nel gestire grandi volumi di dati, è una scelta comune, sebbene possa avere una precisione leggermente inferiore rispetto ad algoritmi più complessi.

- **Decision Trees:**

sono particolarmente utili quando i dati hanno una struttura gerarchica o decisionale. Funzionano suddividendo i dati in gruppi più piccoli, basandosi su regole decisionali semplici, fino a quando non sono necessarie ulteriori decisioni, raggiungendo un risultato finale. Il modello identifica così, a quale classe appartiene il dato (per la classificazione) o prevede un valore specifico (per la regressione).

Questa tecnica è utile nel contesto della classificazione delle intenzioni nei chatbot, i quali possono identificare la richiesta o domanda dell’utente e guidare il chatbot verso la risposta corretta. Oppure per attività come il tagging delle parti del discorso (identificare verbi, sostantivi, ecc.) e la rilevazione di argomenti sui social media, segmentando i dati linguistici in base a regole specifiche.

- **K-Nearest Neighbors (KNN):**

utilizzato nel NLP per compiti di classificazione basati sulla similarità del testo. Esempi di applicazioni includono la classificazione di testi o frasi confrontandoli con altri esempi già classificati. Questa tecnica è particolarmente utile in scenari come la raccomandazione di film su piattaforme come Netflix, dove le somiglianze tra utenti o film vengono misurate tramite metriche di distanza, utilizzate nel K-Nearest Neighbors. Ad esempio, può suggerire film simili a quelli che l’utente ha già visto, basandosi sulle preferenze di altri utenti con gusti affini.

2.2.3 Deep Learning nell’apprendimento supervisionato

Nell’apprendimento supervisionato, il Deep Learning permette ai modelli di NLP di comprendere meglio il contesto e le relazioni semantiche nei testi. Tecniche come le reti neurali convoluzionali (CNN), le reti neurali ricorrenti (RNN), le Long Short-Term Memory (LSTM) e i Transformers sono particolarmente efficaci per compiti di NLP grazie alla loro capacità di gestire grandi volumi di dati e catturare intricate relazioni tra parole e frasi.

Le reti neurali nel NLP

Dopo aver fornito una definizione delle reti neurali nel capitolo precedente, ci concentreremo ora sull’applicazione specifica di queste reti nel contesto del Natural Language Processing.

- **Reti Neurali Convoluzionali (CNN):** inizialmente progettate per l’elaborazione di immagini, le CNN vengono utilizzate anche nel linguaggio naturale per riconoscere pattern locali come frasi o combinazioni di parole, utili in attività come la classificazione del testo, il riconoscimento di entità e l’analisi del sentiment.
- **Reti Neurali Ricorrenti (RNN):** come discusso nel capitolo precedente, le RNN sfruttano una memoria interna che permette al modello di utilizzare informazioni provenienti dagli input precedenti. Questo le rende particolarmente adatte per compiti come la traduzione automatica, il riconoscimento vocale e il riconoscimento di sequenze. Tuttavia, hanno difficoltà nel mantenere informazioni rilevanti su sequenze molto lunghe.
- **Long Short-Term Memory (LSTM):** grazie alla loro struttura interna, sono capaci di mantenere informazioni rilevanti per lunghi periodi durante l’apprendimento. Questo le rende particolarmente efficaci nel gestire dipendenze a lungo termine in compiti come la generazione di testo e il riconoscimento del linguaggio.
- **Transformers:** hanno rivoluzionato l’elaborazione del linguaggio naturale grazie al loro meccanismo di self-attention. Questo approccio ha portato alla creazione di modelli come BERT e GPT, che hanno ridefinito gli standard nei campi della traduzione automatica, generazione di testo e riconoscimento del linguaggio naturale.

2.3 Applicazioni del NLP nell'AI

L'elaborazione del linguaggio naturale permette alle macchine di interagire con gli esseri umani in modo più naturale ed intuitivo. Tra le applicazioni più rilevanti troviamo:

- *Assistenti virtuali.*
Siri, Alexa, Google Assistant, sono un classico esempio. Utilizzano L'NLP per riconoscere il linguaggio naturale, comprendere l'intento dell'utente e generare risposte informative e pertinenti.
- *Chatbot.*
Come ChatGpt, i chatbot sono capaci di emulare conversazioni con gli utenti, gestendo interazioni complesse e memorizzando le risposte date in precedenza. Inoltre, offrono la possibilità di essere addestrati tramite uno specifico prompt engineer.
- *Traduzione automatica.*
L'NLP è utilizzato per tradurre testi da una lingua all'altra in modo automatico, preservando il significato e lo stile del testo originale.
- *Analisi del sentiment.*
Permette l'estrazione di informazioni su opinioni da testi in diversi contesti, quali social media, recensioni, etc.. Viene utilizzato in particolar modo nelle aziende per comprendere meglio le opinioni dei clienti.
- *Riconoscimento vocale.*
Utilizzato in dispositivi di assistenza e applicazioni mobili per convertire il parlato in testo. Viene impiegato per attivare comandi vocali, trascrivere conversazioni, e migliorare l'accessibilità per persone con disabilità, permettendo una navigazione hands-free.
- *Generazione di testi ed estrazione di informazioni.*
Utilizzati rispettivamente per automatizzare la creazione di contenuti, come la scrittura di articoli o report, e per identificare dati rilevanti all'interno di documenti, come nomi, date o entità chiave.

Capitolo 3

I Large Language Models

In questo capitolo verrà offerta una panoramica generale sui modelli di linguaggio di grandi dimensioni (LLM), spiegando come questi modelli siano addestrati su vasti corpus di dati testuali per sviluppare una comprensione profonda del linguaggio umano.

Successivamente, tratteremo le differenze tra modelli proprietari, sviluppati e mantenuti da grandi organizzazioni come OpenAI, Google e Anthropic, e modelli open source, che offrono un'alternativa accessibile e flessibile per la ricerca e lo sviluppo in ambito AI. Esploreremo i vantaggi e le sfide di entrambi gli approcci, inclusi aspetti come il supporto tecnico, i costi, la trasparenza e la possibilità di personalizzazione.

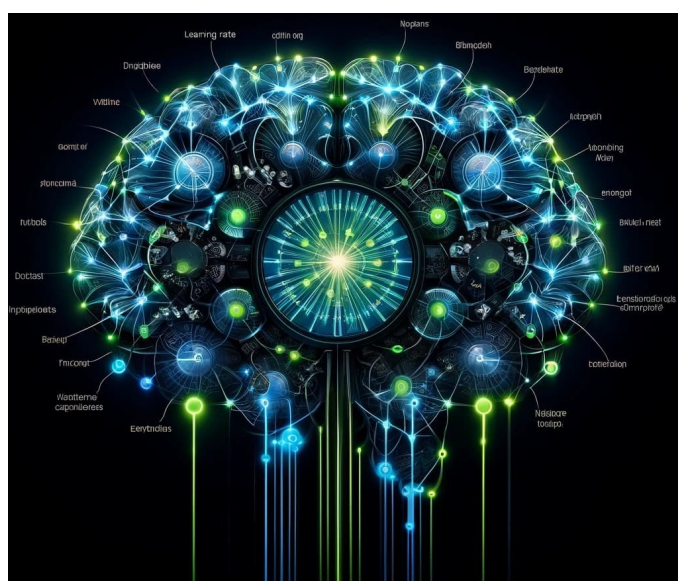
In particolare, approfondiremo la piattaforma Hugging Face, evidenziando i suoi punti chiave e il ruolo che svolge nella comunità open source. Discuteremo di come Hugging Face abbia rivoluzionato il campo dell'NLP e abbia reso accessibili le tecnologie avanzate di AI ad un pubblico ampio e diversificato.

Infine, affronteremo le implicazioni etiche e sociali associate all'uso diffuso degli LLM. Discuteremo dei potenziali bias e discriminazioni che possono emergere dall'uso di questi modelli, della privacy e sicurezza dei dati utilizzati per il loro addestramento, della trasparenza e responsabilità nelle decisioni prese dai modelli, e dell'impatto lavorativo e sociale dell'automazione basata sugli LLM.

I modelli di linguaggio di grandi dimensioni sono modelli avanzati nell'ambito del NLP, addestrati su vasti corpus di dati testuali. Grazie a questo, hanno sviluppato diverse capacità, come la generazione di testi coerenti, la risposta a domande e la traduzione automatica, dimostrando una comprensione profonda del linguaggio umano. Sono inoltre caratterizzati da una notevole complessità in termini di dimensioni e numero di parametri.

"Gli LLM hanno il potenziale per trasformare il modo in cui interagiamo con le informazioni e le tecnologie, rendendo l'accesso al sapere più democratico e l'interazione con le macchine più naturale ed intuitiva. - [11]"

Esempi notevoli includono i modelli GPT di OpenAI, il modello Gemini di Google e i modelli Llama di Meta. Il loro addestramento richiede risorse significative, motivo per cui solitamente viene affidato a grandi organizzazioni come Microsoft e OpenAI. Tuttavia, tecniche come il Fine-Tuning e il *Retrieval-Augmented Generation (RAG)* ¹ permettono l'ottimizzazione del modello per specifiche necessità.



Ogni nodo rappresenta un iperparametro connesso che simboleggia il flusso di informazioni. Al centro, il nucleo che funge da "cuore", dove avviene il processo di decisione e apprendimento.

¹Il Retrieval-Augmented Generation è una tecnica che combina due fasi: il recupero di informazioni rilevanti da una base di dati o documenti e la generazione di testo. Utilizza un sistema di ricerca per trovare contenuti utili e li incorpora nella generazione finale del testo.

3.2 Modelli Proprietari VS modelli Open Source

Gli LLM si suddividono in due principali categorie: modelli proprietari e modelli open source. La scelta tra questi due approcci comporta implicazioni significative in termini di accessibilità, flessibilità, supporto e costo.

3.2.1 Modelli proprietari

I modelli proprietari sono sviluppati, mantenuti e controllati da organizzazioni o individui che detengono i diritti esclusivi sul codice sorgente. Sono inoltre, generalmente protetti da diritti di proprietà intellettuale. Le principali caratteristiche che li contraddistinguono sono le seguenti:

- **Accesso limitato:** il codice sorgente è privato, quindi la visualizzazione, la modifica e la distribuzione del codice sono limitate ai detentori dei diritti.
- **Supporto e aggiornamenti:** garantiscono assistenza continua.
- **Controllo e sicurezza:** poiché solo il team di sviluppo interno ha accesso al codice, c'è un maggiore controllo sulla qualità e la sicurezza del software.
- **Costo:** il loro utilizzo può essere costoso, poiché l'accesso e l'uso del software sono generalmente soggetti a pagamento e regolati da licenze restrittive.

Tra i modelli proprietari più di rilievo troviamo, GPT per OpenAI, Gemini per Google e Claude per Anthropic.

OPENAI - GPT

OpenAI è un'organizzazione di ricerca e sviluppo nel campo dell'AI, riconosciuta come leader nel settore. Fondata nel 2015 come un'associazione no-profit, si è trasformata nel 2019 in una società a scopo di lucro limitato per attrarre investimenti e accelerare i progressi nella ricerca e nell'applicazione delle tecnologie avanzate. La serie di modelli GPT (Generative Pre-training Transformer) sviluppati da OpenAI ha segnato importanti traguardi nell'evoluzione dell'intelligenza artificiale. Relativamente al campo dell'NLP ricordiamo:

- **GPT-1 (2018).**

Il primo modello ha utilizzato l'architettura Transformer e quindi meccanismi di self-attention, dimostrando capacità di poter svolgere vari compiti di elaborazione del linguaggio naturale senza bisogno di essere addestrato specificamente per ciascuno di essi.

- **GPT-2 (2019).**

Con 1,5 miliardi di parametri, GPT-2 ha migliorato significativamente la coerenza del testo generato, imparando da una quantità maggiore di dati e gestendo meglio concetti complessi. Questo ha permesso al modello di comprendere meglio il contesto e le relazioni tra le parole.

- **GPT-3 (2020).**

Il vero salto qualitativo è avvenuto con il lancio di GPT-3, che, grazie ai suoi 175 miliardi di parametri, ha notevolmente migliorato la capacità di apprendere nuove attività attraverso un approccio di *apprendimento "few shot"*.²

- **GitHub Copilot (2021).**

GitHub Copilot, sviluppato da GitHub in collaborazione con OpenAI, è un assistente di programmazione progettato per aiutare gli sviluppatori a scrivere codice in modo più rapido e preciso. Utilizza i modelli GPT-3 e GPT-4 per generare suggerimenti di codice in tempo reale su vari linguaggi e contesti di programmazione.

- **GPT-3.5 (2022).**

Utilizzato per il lancio di ChatGPT, GPT-3.5 è stato progettato per sostenere conversazioni su vari argomenti e rispondere in modo più accurato e pertinente alle intenzioni degli utenti. Tuttavia, ha sollevato questioni di copyright riguardo alle fonti utilizzate per il suo addestramento, con accuse di utilizzo non autorizzato di materiale. [4]

²Tecnica di apprendimento automatico che permette a un modello di imparare a svolgere nuove attività o riconoscere nuovi concetti con un numero molto limitato di esempi di addestramento.

- **GPT-4 (2023).**

Questo modello è stato il primo LLM ad incorporare capacità multimediali per gestire input di testo e immagini, migliorando il ragionamento avanzato e l'apprendimento. Successivamente è stato introdotto GPT-4 Turbo, una versione ottimizzata, più veloce e meno costosa.

- **GPT-4o (Maggio 2024).**

GPT-4o offre prestazioni superiori in termini di velocità e capacità rispetto ai suoi predecessori. È più rapido ed economico, con notevoli miglioramenti nella gestione di testo e voce, eccellendo nella comprensione e spiegazione dei contenuti. La versione più leggera, GPT-4oMini, è progettata per dispositivi a risorse limitate, mantenendo molte funzionalità di GPT-4o in modo più efficiente e accessibile.

- **GPT-4o1 (Settembre 2024).**

I modelli o1 e la sua versione più leggera o1-mini sono i prodotti di punta di OpenAI. Sono progettati per risolvere problemi avanzati, ideali in scenari che richiedono analisi approfondite, come la generazione di codice e la revisione di documenti legali. Questi modelli utilizzano un processo chiamato "chain of thought", che consente di eseguire un ragionamento più profondo prima di fornire una risposta, migliorando così la qualità delle soluzioni offerte. Rispetto a GPT-4o, però, manca attualmente di funzionalità come la navigazione web e la generazione di immagini, probabilmente aggiunte in futuri aggiornamenti. [20]

GOOGLE

Google è una multinazionale tecnologica statunitense riconosciuta come leader nel campo dell'intelligenza artificiale e del machine learning. Fondata nel 1998, Google ha apportato contributi significativi nella ricerca e nello sviluppo di tecnologie avanzate e ha integrato progressivamente l'AI in molti dei suoi prodotti e servizi, diventando una delle principali forze trainanti a livello globale. I principali modelli sviluppati nel settore del NLP sono:

- **Bert (2018):** Bidirectional Encoder Representations from Transformers, un modello che ha rivoluzionato l'elaborazione del linguaggio naturale comprendendo il *contesto bidirezionale di parole in una frase*³.
- **T5 (2019):** Text-To-Text Transfer Transformer, progettato per trasformare tutti i compiti di NLP in input-output basati su testo.
- **Meena (2020):** un modello di conversazione avanzato con 2,6 miliardi di parametri.
- **LaMDA (2021):** Language Model for Dialogue Applications, progettato per comprendere e generare dialoghi più naturali e contestuali.
- **PaLM (2022):** Pathways Language Model, progettato per ottimizzare l'efficienza su diverse attività e domini, con un'architettura che facilita l'addestramento e l'*inferenza*⁴.
- **Google Bard (2023):** un modello avanzato per la generazione di testo e conversazione, sviluppato per competere con altri grandi modelli di linguaggio come ChatGPT, utilizzando la tecnologia LaMDA per migliorare la qualità delle risposte.
- **Gemini (2023):** Gemini 1.0 è la versione iniziale della serie, disponibile in tre varianti: Ultra, Pro, e Nano, oltre ad una versione Basic gratuita, la quale rappresenta il primo modello multimodale, accessibile gratuitamente. Si distingue per le sue capacità avanzate di ragionamento, superando molti benchmark accademici e tecnologici. Successivamente, è stato introdotto Gemini 1.5, con miglioramenti nelle capacità multimodali e una comprensione avanzata di testo, immagini e video.
- **Gemma (2024):** è stato lanciato come una famiglia di modelli open-source più leggeri, basati sulla stessa ricerca che ha portato alla creazione di Gemini. E' stato rilasciato in diverse dimensioni, inclusi modelli da 2,9 e 27 miliardi di parametri.

³Il modello non si limita a leggere le parole in una sola direzione, ma analizza contemporaneamente sia il contesto precedente che quello successivo di ogni parola.

⁴Nel contesto del machine learning, si riferisce al processo di esecuzione di un modello addestrato per fare previsioni o prendere decisioni basate su nuovi dati.

ANTHROPIC - Claude

Anthropic è un'azienda specializzata nella ricerca e sviluppo di modelli di intelligenza artificiale, fondata nel 2021 da ex dipendenti di OpenAI. L'azienda si distingue per il suo impegno nella creazione di modelli AI sicuri ed affidabili. Tra i principali modelli di Anthropic troviamo la serie Claude:

- **Claude 2 (2023):** è stato sviluppato con l'obiettivo di competere con modelli come ChatGPT, distinguendosi per le sue avanzate caratteristiche di sicurezza dell'AI e le elevate prestazioni. Il modello enfatizza la sicurezza attraverso l'integrazione di metodologie avanzate per ridurre i bias e promuovere la trasparenza.
- **Claude 3 (2024):** comprende 3 varianti principali:
 - **Claude Haiku:** il modello più rapido e compatto della famiglia, progettato per risposte quasi istantanee. Ideale per compiti che richiedono alta velocità e basso costo, è in grado di elaborare documenti complessi, come articoli scientifici, in pochi secondi, mantenendo un'accuratezza elevata nonostante le sue dimensioni ridotte.
 - **Claude Sonnet:** un equilibrio tra intelligenza e velocità, ottimizzato per carichi di lavoro aziendali su larga scala. Sonnet eccelle nell'elaborazione di grandi quantità di dati, come il recupero di informazioni e la generazione automatica di contenuti.
 - **Claude Opus:** il modello più potente della famiglia Claude 3, progettato per affrontare compiti complessi come la ricerca scientifica e l'analisi strategica. Eccelle nella gestione di contesti lunghi ed offre un'accuratezza quasi perfetta nel richiamo di informazioni.

Anthropic si distingue per il suo impegno nella riduzione dei *bias*⁵ nei modelli e nella promozione della trasparenza e sicurezza, implementando metodologie avanzate come la "*Constitutional AI*"⁶.

⁵Il termine "bias" si riferisce a pregiudizi o distorsioni che possono essere presenti nei modelli di intelligenza artificiale. Questi bias possono derivare dai dati di addestramento utilizzati per sviluppare il modello, che possono riflettere pregiudizi esistenti nella società.

⁶Un metodo sviluppato da Anthropic che definisce principi e regole che guidano il comportamento dei modelli di AI, assicurando che operino in modo sicuro, trasparente e senza comportamenti discriminatori.

3.2.2 Modelli Open Source

I modelli open source offrono un'alternativa accessibile e flessibile per la ricerca e lo sviluppo in ambito AI. Infatti, a differenza dei modelli proprietari, presentano una serie di vantaggi che possono essere fondamentali per ricercatori, sviluppatori ed organizzazioni:

- *Accessibilità*: i modelli open source sono gratuiti e disponibili a chiunque.
- *Trasparenza*: il codice sorgente è pubblicamente accessibile, consentendo agli sviluppatori di comprendere il funzionamento del modello e di adattarlo alle proprie esigenze.
- *Collaborazione*: la comunità open source può collaborare allo sviluppo ed al miglioramento del modello, favorendo l'innovazione e la condivisione delle conoscenze.
- *Flessibilità*: gli utenti possono personalizzare ed adattare i modelli open source alle proprie esigenze specifiche.

Tali modelli sono fruibili tramite diverse piattaforme open, tra cui Hugging Face, che si distingue per la sua vasta libreria di modelli di machine learning e strumenti di sviluppo accessibili a tutti.

Hugging Face

Hugging Face è stata fondata nel 2016, inizialmente come una semplice repository di modelli pre-addestrati per TensorFlow e PyTorch, due librerie open source per il machine learning. Nel 2018, ha acquisito popolarità grazie al rilascio di Transformers, una libreria open source per l'implementazione di modelli di deep learning basati sull'architettura Transformer.

Oggi, Hugging Face è un ecosistema completo ed offre una vasta gamma di componenti, tra cui una libreria ricca di modelli pre-addestrati, strumenti di sviluppo intuitivi ed una comunità attiva che promuove la collaborazione e l'innovazione.

Tale piattaforma ha reso disponibili modelli come Bert, T5, GPT-1 e versioni ridotte di GPT-2 per favorire la ricerca e l'innovazione, rendendo accessibili le tecnologie di AI avanzate ad un pubblico ampio e diversificato.

Tra i punti chiave ricordiamo che la piattaforma offre una serie di *Strumenti di sviluppo*, come, librerie essenziali per l'implementazione e l'addestramento di modelli di intelligenza artificiale. Ecco una panoramica delle principali componenti:

- *Hugging Face Hub*: una piattaforma web che consente agli utenti di cercare, condividere ed utilizzare modelli di AI. Supporta anche il versioning e la collaborazione sui modelli, facilitando l'accesso ad un'ampia comunità di sviluppatori e ricercatori.

- *Librerie di base:* una vasta gamma di librerie per l'implementazione di modelli di intelligenza artificiale. Tra queste, oltre alla sopra citata Transformers, tra le più rilevanti vi sono Tokenizers e Dataset, strumenti fondamentali per la manipolazione dei dati, l'utilizzo ed il relativo addestramento di modelli pre-addestrati.
- *Strumenti per il fine-tuning tools:* utili per adattare modelli pre-addestrati a dati specifici. Questi strumenti includono librerie come Transformers e PEFT (Parameter-Efficient Fine-Tuning) che rendono il processo di adattamento dei modelli ai propri dati più efficiente e meno dispendioso in termini di risorse computazionali.

Inoltre, un ulteriore aspetto fondamentale è il *Supporto alla Comunità* open source, tramite organizzazione di eventi, workshop e collaborazioni con istituzioni accademiche ed aziende tecnologiche.

3.2.3 Confronto tra modelli Proprietari e Open Source:

Abbiamo appena visto le caratteristiche di ciascun tipo di modello; possiamo perciò affermare che, mentre i modelli proprietari offrono soluzioni complete e preconfigurate con supporto dedicato, i modelli open source rappresentano una risorsa preziosa per la comunità scientifica e tecnologica.

I *modelli proprietari* sono progettati per essere utilizzati con un **minimo sforzo di configurazione**, fornendo **supporto tecnico** e **aggiornamenti regolari** per garantire stabilità ed affidabilità. Questi vantaggi li rendono ideali per le aziende che necessitano soluzioni pronte all'uso e possono permettersi i relativi costi. Tuttavia, l'accesso limitato al codice sorgente e le restrizioni sulla personalizzazione possono frenare l'innovazione.

D'altra parte, i *modelli open source*, promuovono **collaborazione** ed **innovazione**. Tale approccio permette a ricercatori, sviluppatori ed aziende di adattare ed ottimizzare i modelli per esigenze specifiche. La trasparenza del codice sorgente aumenta la fiducia e la verifica dei risultati, contribuendo ad un progresso scientifico condiviso. Inoltre, la possibilità di collaborare su una piattaforma globale, accelera lo sviluppo di nuove tecnologie, rendendo l'intelligenza artificiale più inclusiva ed accessibile.

3.3 Implicazioni etiche e sociali

Fino ad ora abbiamo esplorato gli LLM esaminando le caratteristiche dei modelli proprietari ed open source, analizzando i principali modelli più potenti in uso oggi, come quelli sviluppati da OpenAI. Tuttavia, è cruciale anche considerare le implicazioni etiche e sociali associate a queste tecnologie, le quali meritano particolare attenzione. L'adozione diffusa degli LLM comporta non solo benefici, ma anche sfide che devono essere affrontate per garantire un uso responsabile ed etico di queste tecnologie avanzate. Le implicazioni etiche e sociali riguardano diversi aspetti, tra cui:

- **Bias e discriminazione.**

Poiché gli LLM vengono addestrati su enormi quantità di dati, possono riflettere ed amplificare i pregiudizi presenti nella società. Senza un adeguato controllo, potrebbero generare risultati ingiusti e discriminatori, contribuendo alla diffusione di stereotipi dannosi ed all'emarginazione di determinati gruppi sociali.

- **Privacy e sicurezza dei dati.**

L'addestramento degli LLM richiede l'accesso a grandi quantità di dati, spesso sensibili. È quindi essenziale sviluppare sistemi robusti per proteggere la privacy e la sicurezza degli individui, prevenendo l'uso improprio di queste tecnologie, che potrebbero portare anche alla generazione di deepfake.

- **Trasparenza e responsabilità.**

La complessità degli LLM rende difficile per gli utenti comprendere come vengono prese le decisioni. La mancanza di trasparenza può limitare la fiducia degli utenti e solleva questioni sulla responsabilità in caso di errori o abusi.

- **Impatto lavorativo.**

L'avvento degli LLM comporta anche sfide di natura sociale. L'automazione dei compiti ripetitivi potrebbe portare alla perdita di posti di lavoro, con conseguenze negative sull'occupazione e sulla coesione sociale.

Per affrontare queste sfide, è necessario adottare un approccio responsabile e consapevole. È inoltre fondamentale educare il pubblico sui potenziali rischi; solo così potremo costruire un futuro in cui gli LLM vengano utilizzati per il bene comune, migliorando la nostra vita e la nostra società.

Capitolo 4

L'impiego degli LLM nelle applicazioni mediche

L'introduzione dei modelli linguistici di grandi dimensioni ha segnato un punto di svolta significativo nel settore medico. Questi avanzati strumenti di elaborazione del linguaggio naturale stanno trasformando la gestione delle informazioni, la pratica clinica, la ricerca e l'educazione medica. Nonostante il loro enorme potenziale, l'implementazione clinica di questi modelli solleva questioni cruciali riguardo all'accuratezza, all'affidabilità e alla sicurezza, considerando la natura critica delle informazioni sanitarie.

4.1 Innovazione tecnologica degli LLM

Nel 2023, l'introduzione di modelli avanzati come GPT-4 di OpenAI, Bio-MedLM, LLama Serie2, Mistral e Med-PaLM2 ha rappresentato un salto tecnologico nel campo degli LLM. [3] Questi modelli sono stati addestrati su vasti corpus di testi, inclusi dati biomedici provenienti da fonti come PubMed e PubMed Central, per acquisire una comprensione approfondita del linguaggio medico. Nonostante GPT-4 non sia stato specificamente sviluppato per applicazioni mediche, ha dimostrato un'eccellente adattabilità a compiti diagnostici e formativi.

Studi scientifici, come quelli riportati nell'articolo "Large language models encode clinical knowledge" [19], hanno evidenziato come GPT-4 si distingua nell'ambito medico, sottolineando la sua versatilità e il potenziale nel fornire supporto in contesti sanitari. Oltre a migliorare la comunicazione clinica, questi modelli possono gestire ampi set di dati con grande efficacia, fornendo risposte precise e contestualmente pertinenti durante le interazioni con i pazienti o nelle consultazioni mediche. Ad esempio, GPT-4 è stato utilizzato per categorizzare parametri clinici non strutturati e suggerire trattamenti basati sulle informazioni ricevute, dimostrando la sua utilità in situazioni cliniche reali.

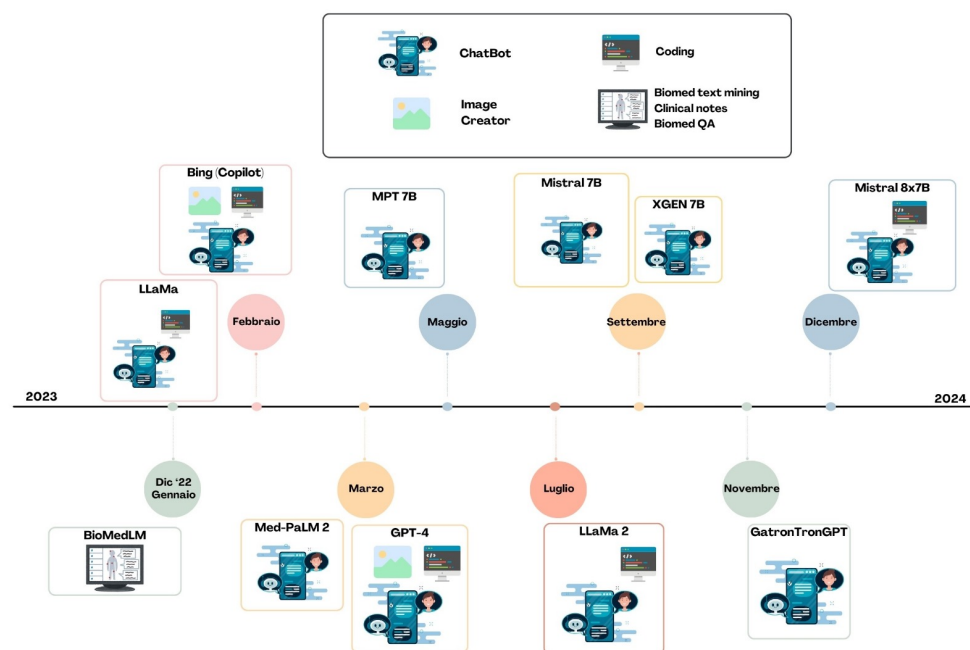


Figura 4.1: Linea temporale dei modelli in continua evoluzione dal 2023 al 2024.

4.2 Benchmarking e confronto tra modelli

Il confronto tra vari modelli di LLM è un aspetto cruciale per valutarne le prestazioni in ambito medico. L'articolo "Evaluating the Feasibility of ChatGPT in Healthcare: An Analysis of Multiple Clinical and Research Scenarios" [2] e altre ricerche hanno esplorato il confronto tra modelli come ChatGPT, Flan-PaLM e Med-PaLM attraverso la pratica del benchmarking.

Il benchmarking è un processo di valutazione delle prestazioni di modelli o sistemi utilizzando metriche standardizzate. Questo processo coinvolge l'uso di dataset specifici e criteri di valutazione per misurare l'efficacia dei modelli in determinati compiti, aiutando a identificare punti di forza e debolezza e a guidare lo sviluppo e l'ottimizzazione dei sistemi.

BioMedLM, noto anche come PubMedGPT 2.7B, rappresenta un significativo avanzamento nella tecnologia degli LLM, essendo stato progettato per comprendere e interpretare il linguaggio biomedico. Questo modello, basato su una versione adattata di GPT-2, utilizza un tokenizer specifico per il dominio biomedico, dimostrando precisione nell'*Open Domain Question Answering*

(*OpenQA*)¹ su dataset medici. Tuttavia, ogni utilizzo clinico di questi modelli richiede una scrupolosa validazione per garantire l'integrità e l'affidabilità delle informazioni.

Un innovativo strumento di benchmarking introdotto in questo contesto è MultiMedQA, presentato nell'articolo "Large language models encode clinical knowledge". MultiMedQA combina sei dataset esistenti di question answering medico e introduce un nuovo dataset, HealthSearchQA, che raccoglie domande mediche comunemente ricercate online. Questo benchmark testa la capacità degli LLM di fornire risposte accurate, comprensibili e sicure, valutando anche la loro comprensione, il ragionamento critico e la capacità di mitigare pregiudizi o danni.

Utilizzando MultiMedQA, è stato possibile dimostrare l'efficacia di BioMedLM nell'interpretare e rispondere a quesiti medici complessi, evidenziando al contempo la necessità di una continua verifica clinica per assicurare che le informazioni fornite siano sempre accurate e affidabili.

4.3 Da PaLM a Med-PaLM: innovazioni e sfide nell'ambito medico.

Nell'articolo "Large language models encode clinical knowledge", i ricercatori hanno valutato il modello PaLM e la sua versione potenziata, Flan-PaLM, utilizzando *tecniche di prompting* come few-shot, chain-of-thought e self-consistency.²

Flan-PaLM ha mostrato miglioramenti nel dataset MedQA, ma ha evidenziato limiti nella conformità al consenso scientifico e nella prevenzione di informazioni dannose. E' stato osservato che solo il 61,9% delle risposte dettagliate di Flan-PaLM rispecchiava il consenso scientifico. Inoltre, il 29,7% delle risposte è stato ritenuto potenzialmente dannoso.

Per superare queste problematiche, gli autori hanno proposto l'utilizzo dell'*instruction prompt tuning*, una tecnica che ha notevolmente migliorato la conformità di Flan-PaLM agli standard clinici, portando alla creazione di una nuova versione del modello, denominata **Med-PaLM**. Questo approccio ha significativamente perfezionato la generazione di risposte estese, con il 92,6% delle risposte in linea con il consenso scientifico, un risultato paragonabile al 92,9% delle risposte fornite dai professionisti del settore medico e al solo 5,9% di risposte ritenute dannose.

¹OpenQA è un'area di ricerca nel campo dell'AI focalizzata su sistemi capaci di rispondere a domande su qualsiasi argomento, a differenza dei sistemi chiusi che si limitano a domande su temi specifici.

²Le tecniche di prompting sono metodi utilizzati per guidare le risposte generate dai modelli, sfruttando l'input fornito per migliorare la pertinenza e la qualità delle risposte.

- Few-shot Prompting: è necessario fornire al modello pochi esempi di domande e risposte.
- Chain-of-Thought Prompting: serve esplicitare il ragionamento passo dopo passo.
- Self-consistency: genera più risposte ad una domanda per selezionare la risposta migliore.

4.4 Applicazioni degli LLM nel settore sanitario

L'articolo "The future landscape of large language models in medicine" [6] esplora le numerose applicazioni degli LLM nel settore sanitario, dimostrando come questi modelli stiano trasformando la cura dei pazienti, la ricerca e l'educazione medica. Grazie alla loro capacità di analizzare e generare testi complessi, gli LLM migliorano l'efficienza clinica e facilitano la redazione di documentazione scientifica di alta qualità.

Questa prospettiva è ulteriormente supportata dall'articolo "Adapted Large Language Models Can Outperform Medical Experts in Clinical Text Summarization" [23], che mostra come gli LLM, specialmente se adattati a contesti clinici specifici, possano superare gli esperti umani nella sintesi di documenti complessi, come rapporti radiologici e note di progresso.

L'adozione di questi modelli non solo riduce significativamente il carico di lavoro dei medici, ma consente anche di migliorare l'assistenza ai pazienti, permettendo ai medici di concentrarsi maggiormente sulle esigenze cliniche dirette piuttosto che sulla gestione della documentazione.

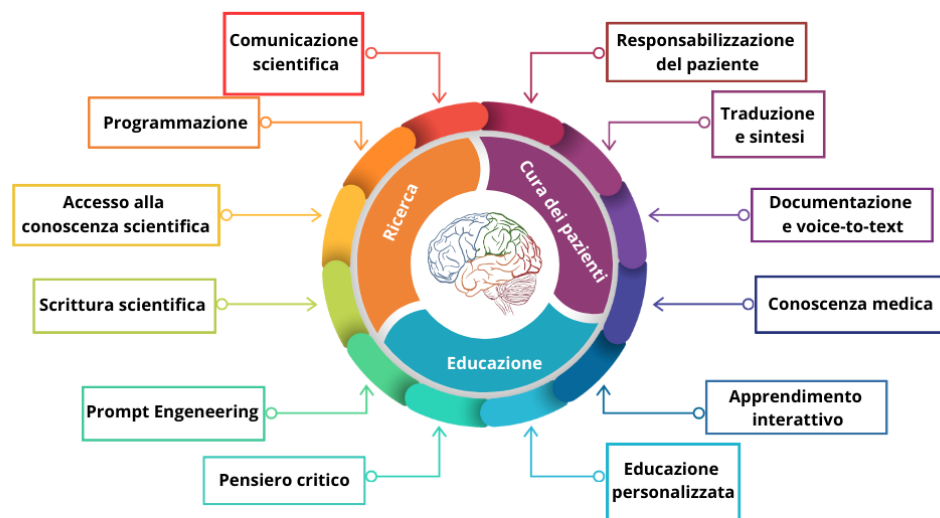


Figura 4.2: L'integrazione dell'AI nel campo medico.

Come mostrato in figura l'intelligenza artificiale supporta diverse aree della sanità, dalla ricerca scientifica alla cura personalizzata dei pazienti.

4.4.1 Supporto alla pratica clinica

Gli LLM, offrono un supporto significativo e sempre più avanzato nella pratica clinica, con applicazioni che spaziano dalla generazione di note mediche alla categorizzazione dei parametri clinici. Sebbene questi modelli dimostrino buone capacità, mostrano limitazioni nella comprensione delle relazioni causali complesse. Di conseguenza, possono generare risposte imprecise o fuorvianti in contesti clinici complessi, a causa della mancanza di una comprensione profonda dei meccanismi patologici. Per questo motivo, è fondamentale una supervisione umana per garantire la precisione e la sicurezza delle raccomandazioni generate dal modello. Inoltre, questioni etiche e legali, come la privacy dei pazienti e la responsabilità medica, richiedono un'attenta considerazione quando si utilizzano LLM nella pratica clinica.

4.4.2 Scrittura scientifica

L'uso di ChatGPT nella scrittura scientifica ha mostrato la capacità di comprendere e sintetizzare articoli scientifici. In test con articoli pubblicati su riviste di alto livello come il New England Journal of Medicine (NEJM), ChatGPT ha prodotto riassunti coerenti e ha dimostrato di poter generare testo di alta qualità, facilitando il lavoro dei ricercatori. Secondo l'articolo su Nature, il modello è stato in grado di analizzare le sezioni di background, metodi e risultati degli studi, generando conclusioni che fornivano una panoramica accurata dei principali risultati e del contesto dello studio. Questa capacità di generare testi accademici di qualità contribuisce a rendere più efficiente la produzione di documentazione scientifica, consentendo ai ricercatori di dedicare più tempo all'analisi dei dati e alla sperimentazione. Sebbene gli LLM possano accelerare il processo di scrittura, possono anche introdurre inesattezze e bias. Pertanto, è essenziale garantire l'integrità scientifica attraverso una revisione attenta da parte degli autori, includendo la verifica e la validazione delle informazioni.

4.4.3 Educazione medica

Gli LLM svolgono un ruolo fondamentale anche nell'educazione medica, facilitando l'apprendimento interattivo e personalizzato. Possono essere utilizzati per creare materiali didattici, simulazioni cliniche e strumenti di valutazione che aiutano studenti e professionisti medici ad approfondire la loro conoscenza e competenza. Ad esempio, ChatGPT può essere impiegato per generare scenari clinici complessi che gli studenti possono utilizzare per esercitarsi nella diagnosi e nel trattamento di pazienti virtuali. Questo approccio interattivo non solo migliora l'apprendimento, ma prepara anche meglio gli studenti per situazioni cliniche reali. Tuttavia, è importante garantire che le informazioni fornite siano aggiornate e basate su evidenze scientifiche, per evitare la diffusione di conoscenze obsolete o inesatte.

Capitolo 5

Ottimizzazione di LLM per l'ipertensione: Fine-Tuning con risorse Open Source

Il progetto si pone l'obiettivo di migliorare le risposte in ambito medico, con un focus specifico sull'ipertensione, attraverso la tecnica del fine-tuning su un modello linguistico di grandi dimensioni preesistente, utilizzando risorse open source. Gli strumenti impiegati includono Google Colab con GPU T4, la piattaforma Hugging Face con la libreria Transformers, PyTorch, Pandas e Ludwig, tutti disponibili gratuitamente.

Prima di approfondire i dettagli dell'implementazione, è necessario comprendere il concetto di fine-tuning. Il **fine-tuning** [21] è una tecnica di apprendimento automatico che consente di adattare un modello preaddestrato su un ampio dataset generico, ad un compito specifico, utilizzando un dataset più piccolo e mirato al dominio di interesse. Questo approccio sfrutta le conoscenze generali acquisite durante il preaddestramento, riducendo notevolmente il tempo e le risorse necessarie rispetto all'addestramento da zero, migliorando al contempo la precisione del modello per applicazioni specifiche.

Di seguito verrà presentata una panoramica generale sull'importanza dell'utilizzo dei chatbot nella gestione dell'ipertensione, spiegando le motivazioni che hanno guidato lo sviluppo di questo progetto.

Saranno poi illustrati i passaggi chiave seguiti per la sua realizzazione, partendo dalla creazione e preparazione del dataset, fino al fine-tuning del modello e all'analisi dei risultati ottenuti. Questi passaggi comprendono l'implementazione di script per la gestione dei dati, la configurazione e ottimizzazione del modello, l'uso di tecniche specifiche per il fine-tuning, e la valutazione delle performance.

5.1 I Chatbot come strumento di supporto per il paziente iperteso

L'ipertensione, comunemente nota come "pressione alta", è una condizione cronica che colpisce milioni di persone in tutto il mondo. Caratterizzata da un costante aumento della pressione del sangue nelle arterie, questa patologia richiede un monitoraggio regolare e una gestione accurata per prevenire complicanze gravi come malattie cardiovascolari o ictus. A causa della sua natura cronica, i pazienti ipertesi devono tenere costantemente sotto controllo i loro valori e aderire scrupolosamente alle terapie prescritte.

Nell'ambito medico, il progresso tecnologico ha portato all'introduzione di strumenti innovativi, come i chatbot basati su modelli di linguaggio di grandi dimensioni. Questi strumenti sono progettati per assistere i pazienti nella gestione quotidiana della loro condizione, fornendo promemoria per il monitoraggio della pressione sanguigna, motivando all'adesione della terapia e offrendo supporto emotivo. I chatbot, grazie alla loro capacità di comprendere e generare linguaggio naturale, possono interagire in modo empatico con i pazienti, migliorando l'esperienza complessiva di autogestione della malattia. L'obiettivo del progetto è quindi quello di migliorare ulteriormente l'efficacia delle risposte di questi chatbot nel contesto medico, con un focus specifico sull'ipertensione. Questo non significa che modelli come GPT-4, tra i più avanzati sul mercato, non siano in grado di offrire risposte empatiche e un valido supporto ai pazienti. Tuttavia, esistono seri problemi di privacy che non possono essere trascurati, come discusso nel lavoro *Data Decentralisation of LLM-Based Chatbot Systems in Chronic Disease Self-Management* [15], che mette in luce l'impatto dei modelli su larga scala in termini di sicurezza e gestione dei dati. Questi modelli, poiché operano tramite API esterne, catturano informazioni sensibili dalle interazioni con i pazienti aumentando i rischi di violazione della privacy e potenzialmente non rispettando normative come il GDPR in Europa o l'HIPAA negli Stati Uniti.

Per ovviare a queste problematiche, si è deciso di lavorare su modelli open source, che oltre ad essere gratuiti, offrono un maggiore controllo sulla gestione dei dati sensibili. Questo approccio permette di eseguire i modelli localmente, trattando i dati direttamente in locale senza dover utilizzare chiamate API verso server remoti, riducendo così i rischi di "data breach".

In questo modo, il progetto mira a sviluppare soluzioni su misura per le esigenze dei pazienti ipertesi, fornendo un supporto che rispetti la riservatezza dei loro dati e migliori la gestione della loro condizione cronica.

5.2 Creazione del Dataset

Il dataset è stato creato tramite uno script in Python, utilizzando le API di OpenAI e il modello "gpt-4o". Si è partiti da un set iniziale di 100 quesiti messi a disposizione. Per ciascuno di questi, sono state generate 10 domande analoghe, creando così un set di dati composto indicativamente da 1000 righe. Ogni quesito è stato accompagnato da una risposta corrispondente, generata tramite lo stesso script.

5.2.1 Implementazione dello Script

Lo script mostrato nella porzione di codice, segue un approccio standard comunemente suggerito nella guida ufficiale di OpenAI, per la creazione di chatbot, utilizzando le API di GPT-4. In questo contesto, viene sfruttata la tecnica del prompting, fornendo al modello istruzioni specifiche per definire sia il ruolo del chatbot come "assistente virtuale" specializzato in cardiologia e ipertensione, sia il contesto della conversazione. Le frasi, ad esempio, vengono istruite affinché siano brevi, coerenti, empatiche e chiudano sempre con un punto. Questo approccio consente di gestire le conversazioni configurando in modo preciso i ruoli di sistema e utente, facilitando la creazione di risposte rilevanti e adattate al contesto della richiesta.

```
def generate_response(self, new_questions):
    response = openai.ChatCompletion.create(
        model="gpt-4o",
        messages=[
            {"role": "system", "content": "Sei un assistente virtuale"},
            {"role": "user", "content": "In qualità di esperto di
                ipertensione e cardiologia, dovresti rispondere alle
                richieste dei pazienti per fornire diagnosi o consigli.
                Le risposte devono essere empatiche, di senso compiuto
                e TERMINARE CON UN PUNTO.
                Segui l'esempio della risposta alla seguente domanda:
                DOMANDA: Come posso migliorare il monitoraggio dei
                dati della mia frequenza cardiaca?
                RISPOSTA: Assicurati di calibrare regolarmente i tuoi
                dispositivi e di verificare la loro precisione
                confrontandoli con metodi di misurazione tradizionali.
                {new_questions}"},
        ],
        max_tokens=60,
        n=1,
        stop=None,
        temperature=0.8,
    )
    answer = response.choices[0].message.content.strip()
```


5.3 Caricamento e preparazione del Dataset

Al termine della generazione del dataset, il passo successivo è stato caricarlo su Google Colab. Da qui, si sono effettuate operazioni fondamentali per l'addestramento del modello, tra cui, la data augmentation per migliorarne la generalizzazione, la suddivisione in set di training, validation e test per una valutazione accurata, ed un'analisi dettagliata delle caratteristiche del dataset per ottimizzare le prestazioni del modello.

5.3.1 Data Augmentation

La data augmentation [9] è una tecnica utilizzata per ampliare e diversificare il dataset, permettendo al modello di apprendere da un numero maggiore di esempi. Questo processo rafforza la capacità del modello di generalizzare e di fare previsioni accurate su dati nuovi e non visti, migliorandone la robustezza. La tecnica applicata, chiamata "data augmentation basata su sinonimi", consiste nel sostituire alcune parole nel testo originale con i loro sinonimi. Questo approccio mantiene il significato generale del testo, ma introduce variazioni lessicali.

Per realizzarla è stata utilizzata la libreria *nlpaug* e il modulo *SynonymAug* basato su WordNet, strumenti facili da integrare e usare, che offrono risultati immediati. Inoltre, questa tecnica è poco dispendiosa in termini di risorse computazionali.

 Dimensione del dataset originale (bilanciato): 1434

 Dimensione del dataset dopo l'augmentazione: 2868

Figura 5.1: Nella figura viene evidenziato come siano aumentate le dimensioni del dataset di partenza.

5.3.2 Suddivisione del Dataset

La suddivisione del dataset in set di training, validation e test è essenziale per garantire che il modello non solo apprenda correttamente dai dati di training, ma sia anche in grado di generalizzare efficacemente su dati non visti, permettendo una valutazione più accurata delle sue prestazioni finali. Ogni set ha un ruolo specifico nel processo di addestramento del modello:

- **Training Set**

Questo set è utilizzato per addestrare il modello. Include la maggior parte dei dati disponibili e permette di apprendere i *pattern*¹ e le relazioni presenti nei dati, migliorando le sue prestazioni.

- **Validation Set**

Questo set è essenziale per ottimizzare il modello, garantendo che non solo apprenda dai dati di training, ma sia anche in grado di generalizzare efficacemente su dati non visti, mantenendo un equilibrio tra accuratezza e capacità di generalizzazione. Aiuta inoltre a prevenire l'*overfitting*², ottimizzare gli iperparametri e decidere il momento ottimale per fermare l'allenamento. Al termine del training, il modello che performa meglio sul validation set viene spesso scelto come modello finale.

- **Test Set**

Questo set serve a valutare le prestazioni finali di un modello su dati completamente nuovi, utilizzato esclusivamente dopo il completamento del training e della validazione.

5.3.3 Analisi del Dataset

Dopo aver suddiviso il dataset, è fondamentale analizzare le caratteristiche delle domande e delle risposte per garantire una preparazione adeguata del modello. Il calcolo della lunghezza media fornisce informazioni importanti per regolare i parametri del modello, come la lunghezza massima delle sequenze di input e output, rendendo il processo di addestramento più efficiente e assicurando che il modello gestisca e generi risposte in modo ottimale.

¹I pattern sono schemi o comportamenti ricorrenti nei dati che il modello impara a riconoscere e usare per fare previsioni.

²Overfitting è un termine usato per descrivere una situazione in cui un modello di Machine Learning si adatta troppo strettamente ai dati di training, imparando non solo i pattern generali ma anche errori e dettagli irrilevanti o casuali del dataset. Di conseguenza, il modello performa molto bene sui dati di training, ma generalizza male su dati nuovi e non visti.

5.4 Caricamento del Modello

Per assicurare prestazioni elevate e affidabilità, sono stati utilizzati modelli preaddestrati disponibili su Hugging Face, caricati e gestiti tramite la libreria Transformers. Questa, facilita il caricamento e la gestione dei modelli, supportando varie architetture e configurazioni. In particolare, la libreria, include strumenti come il tokenizer e il supporto per modelli quantizzati, che ottimizzano le risorse computazionali e migliorano le prestazioni complessive del modello.

La combinazione di PyTorch [10] e Transformers [5] permette di sfruttare appieno la gestione efficiente della memoria, fondamentale per lavorare con modelli di Deep Learning complessi.

5.4.1 Il Tokenizer

Il tokenizer è uno strumento essenziale nei modelli di linguaggio naturale. Esso converte il testo in input (stringhe di caratteri) in una sequenza di token numerici che il modello può elaborare. Questo processo è fondamentale perché i modelli di Machine Learning non possono elaborare testo grezzo e richiedono input numerici per funzionare.

Prima della tokenizzazione, il testo viene spesso normalizzato: convertito in minuscolo, rimosso di punteggiatura o trattato per caratteri speciali. Questo garantisce che il testo sia in una forma standardizzata per il tokenizer. Una volta tokenizzato, ogni token viene mappato ad un identificatore numerico utilizzando un vocabolario predefinito.

Vengono inoltre gestiti dei token speciali come <PAD>, <SEP> e <EOS>, che sono usati per diverse funzioni nei modelli di linguaggio. Ad esempio, <PAD> viene usato per il padding di sequenze di lunghezza variabile, <SEP> separa segmenti di testo diversi e <EOS> indica la fine di una sequenza di testo.

```
tokenizer = AutoTokenizer.from_pretrained(  
    model_path,  
    model_max_length=512,  
    trust_remote_code=True,  
    padding_side="left",  
    add_eos_token=True  
)
```

Questo script mostra il caricamento di un tokenizer preaddestrato da Hugging Face. Utilizza la funzione `AutoTokenizer.from_pretrained` per configurare il tokenizer con il percorso del modello, stabilendo la lunghezza massima della sequenza di token (`model_max_length=512`), il lato del padding (`padding_side="left"`) e specificando che deve essere aggiunto il token di fine sequenza *EOS*: `add_eos_token=True`.

5.4.2 Quantizzazione del modello

La quantizzazione [16] è una tecnica utilizzata per ridurre le dimensioni e la complessità computazionale di modelli di Deep Learning, permettendone l'uso su dispositivi con risorse limitate come GPU di fascia bassa. Questo processo implica la riduzione della *precisione numerica dei parametri del modello*³, ad esempio passando da 8 bit a 4 bit.

Sebbene si verifichi una perdita di informazione, questa è generalmente minima e non compromette significativamente le prestazioni del modello. Infatti, riducendo la complessità computazionale, l'impatto sull'accuratezza risulta contenuto, con una diminuzione solitamente inferiore all'1-2%.

Esistono diverse metodologie di quantizzazione, e in questo caso è stata utilizzata la 'quantizzazione post-training', tecnica avanzata che consente di ridurre la precisione numerica di un modello già addestrato senza necessitare di un nuovo ciclo di addestramento completo. Per fare ciò è stata utilizzata la classe **BitsAndBytesConfig** per la configurazione: una libreria che si integra con Tanformers per gestire la quantizzazione dei modelli.

```
bnb_config = BitsAndBytesConfig(  
    load_in_4bit=True,  
    load_in_8bit=False,  
    llm_int8_threshold=6.0,  
    llm_int8_has_fp16_weight=False,  
    bnb_4bit_compute_dtype="float16",  
    bnb_4bit_use_double_quant=True,  
    bnb_4bit_quant_type="nf4"  
)
```

La porzione di codice mette in evidenza la quantizzazione a 4 bit tramite **BitsAndBytesConfig**. I parametri specificati includono l'abilitazione della quantizzazione a 4 bit (`load_in_4bit=True`), che riduce significativamente la memoria richiesta e accelera i tempi di inferenza. L'uso della doppia quantizzazione (`bnb_4bit_use_double_quant=True`) migliora ulteriormente la compressione dei pesi del modello attraverso due livelli: il primo livello riduce la precisione dei parametri, mappandoli da una rappresentazione a 8 bit a una a 4 bit, riducendo la quantità di dati necessari e velocizzando i calcoli; il secondo livello affina ulteriormente questi valori, riducendo l'errore e mantenendo più informazioni dal modello originale.

Altri parametri definiscono il tipo di dati per i calcoli a 4 bit (`float16`), e il tipo di quantizzazione (`nf4`).

³La "precisione numerica dei parametri del modello" si riferisce al numero di bit utilizzati per rappresentare i valori dei parametri (come pesi e bias). Una maggiore precisione numerica permette una rappresentazione più accurata dei valori, mentre una precisione inferiore riduce la memoria e la complessità computazionale.

5.4.3 Caricamento del Modello Quantizzato

Il caricamento del modello utilizza diverse tecniche avanzate per ottimizzare le risorse hardware e migliorare le prestazioni complessive.

```
model = AutoModelForCausalLM.from_pretrained(  
    model_path,  
    device_map="auto",  
    torch_dtype=torch.float16,  
    offload_folder="offload",  
    trust_remote_code=True,  
    low_cpu_mem_usage=True,  
    quantization_config=bnb_config  
)
```

La configurazione mostrata nel codice utilizza vari parametri per ottimizzare l'uso delle risorse hardware.

Tra questi, la quantizzazione con `bnb_config` e l'offload delle parti del modello sulla CPU (`offload_folder="offload"`) per risparmiare memoria sulla GPU. In questo modo, i calcoli o i dati che non richiedono necessariamente l'elaborazione sulla GPU vengono spostati sulla CPU.

Inoltre il parametro `device_map` è impostato su "auto" per assegnare automaticamente le diverse parti del modello ai dispositivi hardware disponibili, come CPU e GPU, garantendo un utilizzo efficiente della memoria e della potenza di calcolo.

Un altro parametro importante è `torch_dtype=torch.float16`, che imposta i dati di PyTorch a 16 bit in virgola mobile. Questo tipo di dato viene utilizzato per le operazioni di calcolo (come ad esempio le previsioni), anche se i pesi del modello possono essere quantizzati a 4 bit. In questo modo, si mantiene un equilibrio tra precisione e prestazioni, sfruttando l'efficienza del calcolo in virgola mobile a 16 bit, riducendo al contempo l'occupazione di memoria e migliorando la velocità di calcolo.

5.5 Ottimizzazione della memoria

L'ottimizzazione della memoria è una pratica essenziale quando si lavora con modelli di Deep Learning su hardware con risorse limitate, come GPU con memoria ridotta. Queste limitazioni possono rappresentare un serio ostacolo durante l'addestramento di reti neurali complesse, che richiedono molta memoria per archiviare *pesi, attivazioni e variabili intermedie*.⁴

5.5.1 Tecniche di ottimizzazione utilizzate

- **Gradient Checkpointing.**

Riduce l'uso della memoria durante l'addestramento, permettendo di gestire modelli molto grandi su GPU con memoria limitata, senza perdere precisione.

Normalmente, durante l'addestramento di un modello, vengono salvate molte informazioni intermedie durante la fase di propagazione in avanti (forward pass) per essere utilizzate nella fase di propagazione all'indietro (backward pass).⁵

- **Liberare memoria.**

Funzioni come `torch.cuda.empty_cache()` e `gc.collect()` aiutano a liberare memoria GPU e risorse di sistema, riducendo il rischio di esaurimento della memoria.

```
model.gradient_checkpointing_enable()  
# Free up memory  
torch.cuda.empty_cache()  
gc.collect()
```

⁴I pesi sono i parametri appresi dalla rete neurale durante l'addestramento e determinano l'influenza di un neurone sull'altro.

Le attivazioni sono i valori calcolati in ogni strato della rete che determinano quali neuroni si "attivano" per trasmettere informazioni al livello successivo della rete neurale.

Le variabili intermedie sono i dati temporanei generati durante le operazioni interne della rete, utilizzati per ulteriori calcoli

⁵Nel contesto del Deep Learning e dell'addestramento dei modelli di reti neurali, ci sono due fasi principali che sono cruciali per il funzionamento del modello.

Fase di Forward Pass: durante questa fase, i dati di input attraversano la rete strato per strato fino a generare l'output, utilizzando i pesi correnti.

Fase di Backward Pass: dopo aver ottenuto l'output, la backward pass viene utilizzata per aggiornare i pesi della rete neurale. Questo processo inizia calcolando l'errore dell'output rispetto al valore atteso, propagandolo poi all'indietro attraverso la rete.

5.6 Fine-Tuning con Ludwig

Il passo successivo è stato addestrare il modello nel dominio dell'ipertensione per migliorare la precisione delle risposte. A tal fine, è stata utilizzata Ludwig, una libreria open source che facilita la costruzione e l'addestramento di modelli di Machine Learning, ottimizzando l'uso della memoria e delle risorse computazionali. Grazie alle configurazioni YAML, Ludwig rende il processo di definizione e addestramento dei modelli più semplice e automatizzato, risultando particolarmente utile in ambienti con risorse limitate come Google Colab. È stata adottata una configurazione specifica per ridurre l'overfitting e garantire una buona generalizzazione sui nuovi dati.

5.6.1 Tecniche usate durante il Fine-Tuning

Prompting: Few-shot learning

Questo approccio permette al modello di estrapolare i pattern dagli esempi forniti e di applicare efficacemente le conoscenze acquisite a situazioni nuove, migliorando la sua capacità di generalizzazione anche con dati limitati.

Preprocessing e configurazione per la generazione del testo

Il preprocessing prevede la suddivisione casuale del dataset in training, validation e test per valutare le prestazioni del modello. Durante la generazione del testo, la `temperatura` a 0.1, rende il modello più conservativo, privilegiando le parole con probabilità più alta e producendo risposte più precise. Il parametro `top_p`, impostato a 0.9, regola la varietà selezionando solo le parole con probabilità cumulata fino al 90%, bilanciando coerenza e diversità.

LoRa (Low-Rank Adaptation)

LoRa è una tecnica utilizzata per migliorare l'efficienza nell'adattamento di modelli di grandi dimensioni.

Integrata in Ludwig, permette di ridurre significativamente il carico computazionale e l'uso della memoria durante il fine-tuning. Infatti, anziché aggiornare tutti i parametri del modello preaddestrato, LoRA introduce un numero limitato di nuovi parametri che catturano i cambiamenti essenziali necessari per adattare il modello ad un nuovo compito, rendendo così l'aggiornamento complessivo più efficiente.

Quantizzazione dei pesi a 4 bit

Si tratta della stessa tecnica utilizzata in precedenza per caricare il modello. Come già discusso, è fondamentale in scenari con limitazioni di memoria, poiché riduce lo spazio necessario e velocizza i calcoli, mantenendo un buon equilibrio tra precisione e prestazioni.

Iperparametri specifici

Il fine-tuning del modello è configurato con parametri specifici, con l'obiettivo di **migliorare l'efficienza e prevenire l'overfitting**. Tra questi il **numero di epoche** indica quante volte l'algoritmo di apprendimento elabora l'intero set di dati. In questo caso, l'algoritmo utilizzato è il **Gradient Descent**, supportato da parametri come il learning rate, la strategia di `gradient_accumulation_steps` e il learning rate scheduler, che ne regolano l'efficienza e la stabilità.

Il parametro di **`gradient_accumulation_steps`** consente di accumulare i *gradienti*⁶ su più batch prima di aggiornare i pesi del modello, sommandoli per applicare un unico aggiornamento complessivo. Questo riduce il consumo di memoria durante l'addestramento e aiuta a minimizzare la funzione di perdita in modo più efficiente.

Il **learning rate** controlla l'ampiezza dei cambiamenti apportati ai pesi ad ogni passaggio. Un learning rate fisso può risultare inefficiente: se troppo alto, il modello potrebbe oscillare tra soluzioni non ottimali; se troppo basso, rallenta il processo di apprendimento, rendendo difficile la convergenza.

Il **learning rate scheduler** risolve questo problema variando dinamicamente il learning rate, utilizzando diverse strategie.

In questo caso è stata utilizzata la strategia Cosine Annealing, che riduce progressivamente la velocità di apprendimento seguendo una curva coseno, favorendo un adattamento più graduale verso la fine dell'addestramento.

Una frazione di **warmup** dell'1% aumenta gradualmente il learning rate all'inizio dell'addestramento, permettendo al modello di adattarsi con maggiore precisione e dandogli il tempo necessario per apprendere in modo corretto prima di applicare cambiamenti più significativi.

Infine, la **dimensione dei batch** indica quanti campioni vengono elaborati contemporaneamente prima di aggiornare i parametri del modello, il che contribuisce a rendere l'allenamento più efficiente.

Early stopping

Si tratta di una tecnica utilizzata per **prevenire l'overfitting**. Durante l'addestramento del modello, viene monitorata una metrica di validazione, come la perdita di validazione. Se le prestazioni del modello su un set di dati di validazione non migliorano dopo un certo numero di epoche consecutive, l'addestramento viene interrotto, risparmiando risorse computazionali. Questo approccio garantisce che il modello non solo apprenda dai dati di addestramento, ma generalizzi bene su nuovi dati.

⁶Il gradiente può essere definito come un vettore in cui ogni elemento rappresenta la direzione e l'entità del cambiamento necessario per ridurre l'errore. Viene utilizzato dagli algoritmi di ottimizzazione per aggiornare i pesi del modello in modo efficiente.

Callback

In questa sezione sono definite le funzioni che vengono eseguite automaticamente durante diverse fasi del processo di addestramento di un modello. Questi callback sono utili per monitorare e intervenire nel comportamento del processo di addestramento in tempo reale. Nel progetto sono stati utilizzati callback come la visualizzazione della barra di avanzamento, il salvataggio periodico dei checkpoint del modello e l'early stopping, che consentono di monitorare il progresso dell'addestramento e di interromperlo automaticamente se le prestazioni non migliorano.

5.7 Valutazione del modello

La valutazione del modello è una fase cruciale per determinare quanto efficacemente il modello di linguaggio naturale risponda alle domande e generi testi. Per effettuare tali valutazioni nel progetto sono state utilizzate diverse metriche per misurare vari aspetti delle prestazioni durante e dopo l'addestramento: loss, perplexity, token accuracy e BERTScore.

1. **Loss:** viene detta anche funzione di perdita ed è una misura che quantifica quanto l'output generato da un modello si discosti dal valore atteso o reale. In pratica, rappresenta l'errore che il modello compie nelle sue previsioni. Durante l'addestramento di un modello di Machine Learning, l'obiettivo è minimizzare la loss, in modo che il modello produca previsioni il più possibile accurate rispetto ai dati reali.

Valutare la perdita su set di dati diversi (training, validation e test) è fondamentale per comprendere come il modello stia apprendendo e generalizzando. Questo processo aiuta anche a individuare fenomeni come l'overfitting e l'underfitting.

L'**overfitting** si manifesta quando la validation loss è significativamente più alta della training loss (ad esempio di oltre 0.15). Questo indica che il modello sta memorizzando i dati di addestramento, invece di imparare pattern generali, e quindi fatica a generalizzare su nuovi set di dati. D'altro canto, l'**underfitting** si verifica quando sia la training loss che la validation loss sono alte, tipicamente superiori a valori come 1.2. In questo caso, il modello non è in grado di apprendere correttamente dai dati, suggerendo che la sua struttura è troppo semplice o inadeguata per il compito.

2. **Perplexity:** [7] si tratta di una metrica utilizzata principalmente per valutare modelli di linguaggio, che quantifica quanto un modello è "incerto" o "confuso" rispetto alle parole che sta predicendo in una sequenza. La perplessità è definita come la log-verosimiglianza negativa media esponenziata di una sequenza. Se abbiamo una sequenza tokenizzata $X = (x_0, x_1, \dots, x_t)$, la perplessità di X è calcolata come:

$$\text{PPL}(X) = \exp \left\{ -\frac{1}{t} \sum_{i=0}^t \log p_{\theta}(x_i | x_{<i}) \right\}$$

Dove:

- $p_{\theta}(x_i | x_{<i})$: probabilità condizionale che il modello assegna al token x_i , dato che ha già visto i token precedenti della sequenza $x_{<i}$. In altre parole, è la probabilità che il modello predica correttamente il prossimo token basandosi sui token precedenti.
- $\log p_{\theta}(x_i | x_{<i})$: log-verosimiglianza, ovvero il logaritmo della probabilità condizionale. L'uso del logaritmo facilita i calcoli, trasformando la moltiplicazione delle probabilità in somme, rendendo i calcoli più stabili e semplici, specialmente quando si lavora con numeri molto piccoli. ($\log(ab) = \log(a) + \log(b)$)
- $\frac{1}{t} \sum_{i=0}^t \log p_{\theta}(x_i | x_{<i})$: media della log-verosimiglianza su tutti i token della sequenza.
- **exp**: l'esponenziale riporta il valore della log-verosimiglianza negativa media su una scala positiva, rendendo più intuitiva l'interpretazione della perplessità.

Una perplessità bassa indica che il modello è relativamente sicuro delle parole successive nella sequenza, mentre una perplessità alta indica maggiore incertezza e molteplici possibili scelte per la parola successiva. Un range indicativo per valutare la perplessità è il seguente:

- $\text{PPL} < 10$: eccellente, il modello prevede molto bene il testo.
- PPL tra 10-50: accettabile, con buone performance ma migliorabile.
- PPL tra 50-100: problemi evidenti nella capacità di previsione.
- $\text{PPL} > 100$: non ottimale, il modello fatica a catturare i pattern linguistici.

Per arrivare a tali risultati è però fondamentale che il dataset sia di alta qualità e che il fine-tuning del modello venga eseguito correttamente. Tuttavia, va ricordato che la perplessità valuta solo la coerenza interna delle previsioni del modello, senza considerare la qualità effettiva del contenuto generato. Questo significa che una perplessità più bassa non garantisce necessariamente risposte migliori.

3. **Token Accuracy.** Includere questa metrica offre una visione più completa delle prestazioni del modello, in quanto permette di valutare con quale frequenza il modello genera esattamente il token corretto in una sequenza di testo.

Un token può essere una parola completa o, in modelli che utilizzano la *subword tokenization* ⁷, può rappresentare una parte di una parola. In pratica, la token accuracy calcola la percentuale di token predetti correttamente rispetto ai token reali presenti nel testo di riferimento.

Non esiste un intervallo "ottimale" universale per la token accuracy nei modelli NLP, poiché dipende dal compito specifico e dalla complessità del modello. Tuttavia, una token accuracy tra l'80% e il 90% è spesso considerata buona. Nei compiti più complessi, come la generazione di testo libero, una token accuracy superiore al 70% può comunque essere accettabile, poiché è più difficile mantenere un'alta precisione sui singoli token.

Insieme alla perplexity, questa misura fornisce una valutazione equilibrata sia della fluidità del testo che della precisione delle risposte.

4. **BERTScore:** misura la somiglianza tra frasi generate dal modello e frasi di riferimento, valutandole non solo a livello di parole, ma anche a livello semantico. Tale metrica è particolarmente utile nei compiti di generazione di testo in cui la comprensione e la coerenza del significato sono cruciali.

BERTScore, quindi, offre un modo più avanzato di valutare la qualità del testo generato rispetto a metriche più semplici che considerano solo la corrispondenza superficiale tra parole, come BLEU e ROUGE.

Nel progetto, il suo utilizzo prevede due passaggi principali: la generazione delle risposte con il modello fine-tuned e il calcolo del BERTScore utilizzando la funzione score della libreria per confrontare le risposte generate con quelle di riferimento. In questo caso, per un confronto più dettagliato, il BERTScore è stato calcolato sia prima che dopo l'operazione di fine-tuning, permettendo di valutare chiaramente le prestazioni del modello prima e dopo il miglioramento.

Il calcolo del BERTScore si basa sulle metriche di Precision, Recall e F1, che offrono una valutazione più completa delle prestazioni del modello.

Precision misura quante delle parole generate dal modello sono corrette rispetto alla frase di riferimento, valutando la qualità delle parole prodotte. **Recall** valuta quante delle parole rilevanti della frase di riferimento sono state generate dal modello, concentrandosi sulla capacità del modello di recuperare le informazioni importanti. **F1** combina Precision e Recall, bilanciando entrambe le metriche per fornire una visione complessiva della qualità del testo generato.

⁷Tecnica utilizzata per suddividere le parole in unità più piccole chiamate "subword" o "token", che possono essere intere parole, prefissi, suffissi o sequenze di caratteri.

5.8 Analisi dei risultati per modello

Per ottenere risultati chiari, sono stati condotti vari esperimenti su diversi modelli selezionati su Hugging Face:

- **GPT-2:** un modello generico noto per le sue capacità di generazione di testo, sviluppato da OpenAI. È in grado di produrre testi coerenti e creativi in vari contesti, basandosi su milioni di parametri.
- **Mistral-7b:** un modello ottimizzato per vari compiti di NLP, basato su 7 miliardi di parametri.
- **BioMistral-7B:** un modello progettato specificamente per il dominio biomedico, caratterizzato da 7 miliardi di parametri e parte della famiglia Mistral. Questo modello è utilizzato in compiti come la diagnosi assistita e la ricerca scientifica in ambito medico.

Ogni modello è stato sottoposto ad una serie di test distinti, ciascuno eseguito con un numero crescente di epoche di addestramento. Questo approccio incrementale ha consentito di monitorare come le prestazioni del modello migliorassero progressivamente con l'aumentare dei cicli di addestramento. Le prestazioni sono state valutate utilizzando le seguenti metriche: loss, perplexity, token accuracy e BERTScore.

5.8.1 GPT-2

Il modello di GPT-2 utilizzato è "*GroNLP/gpt2-small-italian-embeddings*", [8] basato su GPT-2 piccolo di OpenAI, con *Transformer layers*⁸ identici al modello originale in inglese, ma con il livello lessicale riaddestrato per il vocabolario italiano.

Risultati dei test

I test effettuati sul modello hanno coinvolto diverse fasi di addestramento, con valutazioni condotte dopo 1, 2, 3, 4, 5, 6, 80, 90, 100, 110, 120 e 130 epoche (le prime 6 necessarie per confrontare i risultati con i modelli che verranno successivamente analizzati).

Fin dalle prime epoche, i modelli hanno mostrato miglioramenti significativi nelle metriche di loss e perplexity.

Di seguito i dettagli:

⁸L'architettura utilizzata da GPT-2 è quella del Transformer, composta da diversi strati che elaborano i dati in ingresso, consentendo al modello di apprendere e rappresentare relazioni complesse tra le parole nel testo.

- **Loss**

Il grafico mostra un calo costante della loss nel corso delle epoche per tutti e tre i set di dati, suggerendo che il modello sta progressivamente migliorando la sua capacità di apprendimento. Tuttavia, è interessante notare che la loss del training set parta da un valore molto alto, scendendo rapidamente nelle prime epoche, per poi diminuire in modo instabile a partire dall'epoca 20, mantenendosi costantemente superiore rispetto ai set di validation e test. Questo comportamento è insolito e potrebbe indicare che il modello fatica a ridurre l'errore sui dati di addestramento, non riuscendo a catturare correttamente i pattern presenti nei dati, suggerendo un possibile underfitting.

Al contrario, la loss dei set di validazione e test inizialmente scende rapidamente, per poi diminuire in modo più costante e parallelo, suggerendo che il modello sta generalizzando meglio sui dati non visti.

Tuttavia, i valori della loss rimangono complessivamente troppo elevati per tutti i set, confermando la presenza di underfitting. Ciò significa che il modello è troppo semplice o non ha sufficiente capacità per rappresentare la complessità del problema, rendendolo incapace di apprendere correttamente i pattern nei dati di training e di generalizzare adeguatamente sui set di validation e test.

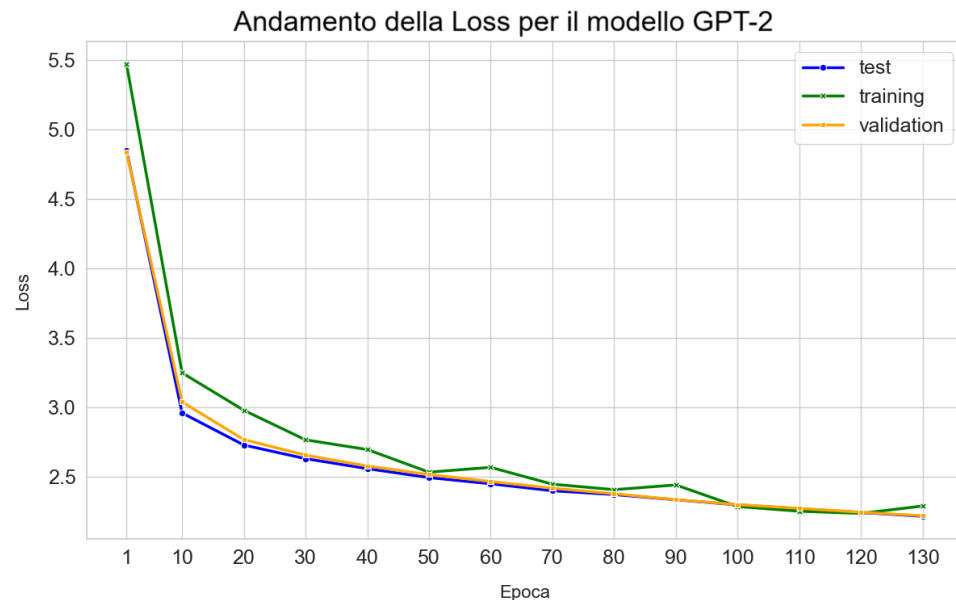


Figura 5.2: Andamento della loss durante le fasi di addestramento del modello GPT-2.

Il grafico mostra una diminuzione costante della loss in tutti i set, segno di una buona generalizzazione. Tuttavia, l'andamento instabile nel training set e i valori complessivamente troppo elevati della loss indicano la presenza di underfitting.

- **Perplexity**

Il grafico mostra che, nelle prime epoche, la loss del training set (in verde) scende rapidamente, per poi assumere un andamento irregolare nelle epoche successive. Ci sono oscillazioni significative, con un picco intorno all'epoca 90, seguito da una riduzione e un leggero rialzo intorno all'epoca 130. Questo comportamento potrebbe indicare che il modello inizialmente è migliorato sui dati di training, ma che successivamente ha incontrato difficoltà, forse a causa di problemi di instabilità nell'ottimizzazione.

D'altro canto, la perplexity del validation set (in arancione) e del test set (in blu) mostra una diminuzione rapida nelle prime epoche, per poi stabilizzarsi nelle epoche successive, mantenendo valori molto simili tra loro. Sebbene questo sia un segnale positivo, che indica un miglioramento nella generalizzazione del modello sui dati non visti, il fatto che i valori di perplexity rimangano comunque su livelli elevati è un aspetto negativo. Ciò suggerisce che il modello ha ancora difficoltà a ridurre significativamente l'incertezza nelle sue previsioni e potrebbe non essere sufficientemente ottimizzato per garantire prestazioni soddisfacenti.

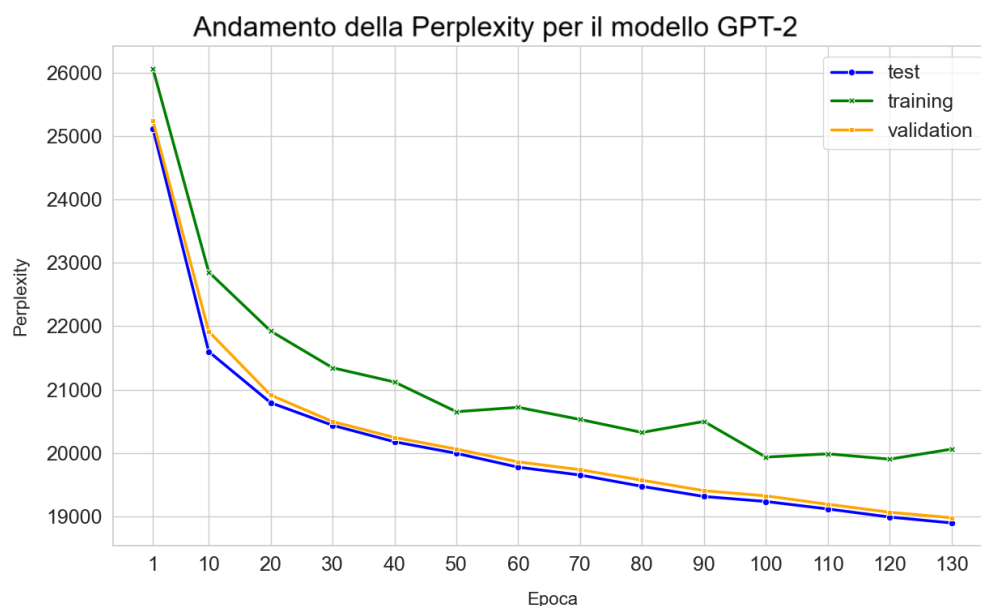


Figura 5.3: Andamento della perplexity durante le fasi di addestramento del modello GPT-2.

Dal grafico emergono difficoltà nell'apprendimento, evidenziate dalle significative oscillazioni nel set di training e dai valori ancora troppo elevati. Anche se il validation set e il test set mostrano un andamento più stabile, i valori complessivamente alti indicano che il modello fatica a migliorare in modo efficace.

- **Token Accuracy**

Nel grafico, la token accuracy per il training set (in verde) mostra una forte diminuzione nelle prime epoche, seguita da una marcata instabilità, caratterizzata da oscillazioni significative che spesso causano una diminuzione del valore anziché un progressivo miglioramento.

Per quanto riguarda il validation set (arancio) e il test set (blu), si osserva una forte diminuzione iniziale, seguita da una marcata stabilizzazione con lievi picchi durante l'addestramento.

Sebbene la stabilità possa sembrare positiva, il fatto che i valori rimangano bloccati su livelli molto bassi è preoccupante.

Questo comportamento suggerisce che il modello soffra di underfitting, poiché non riesce a catturare correttamente i pattern presenti nei dati. Infatti, le oscillazioni nel training set indicano un processo di apprendimento instabile, mentre i valori estremamente bassi per tutti i set, attorno allo 0.005, indicano che solo lo 0.5% dei token viene previsto correttamente, segnalando gravi difficoltà del modello nel fare previsioni accurate.

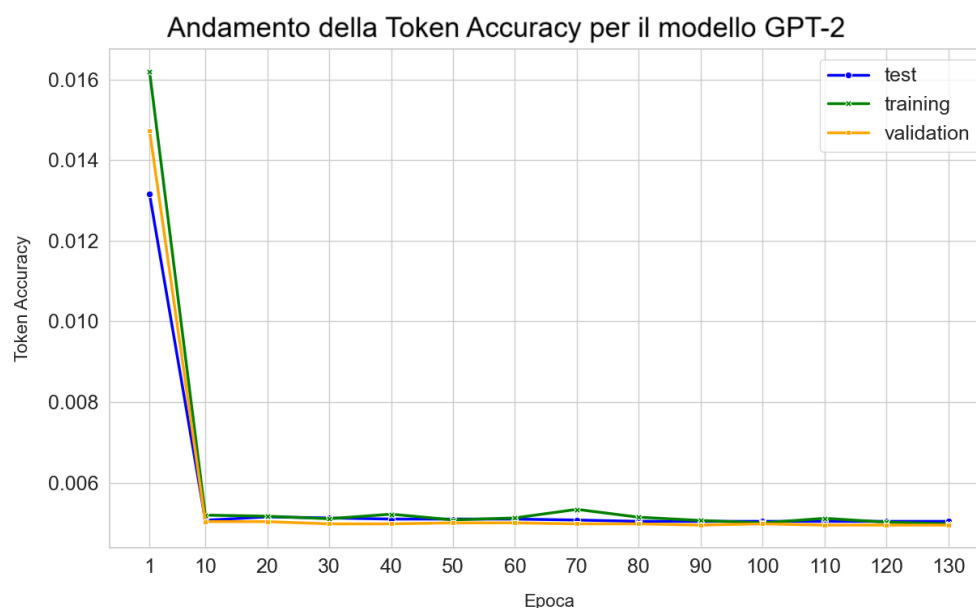


Figura 5.4: Andamento della token accuracy durante le fasi di addestramento del modello GPT-2.

Il grafico evidenzia l'instabilità nella token accuracy del training set, caratterizzata da oscillazioni significative. Al contrario, sia il validation set che il test set mostrano un andamento più stabile, con valori bloccati su livelli molto bassi. Questo, insieme al mantenimento di valori bassi per tutti i set, è un chiaro segno di underfitting.

- **BERTScore**

Come illustrato nella tabella, il processo di fine-tuning ha portato ad un significativo miglioramento delle prestazioni rispetto al modello di base, con un aumento evidente in tutte le metriche considerate. Tuttavia, si osservano diminuzioni notevoli nelle epoche 3 e 4, durante le quali tutte le metriche subiscono un brusco calo prima di riprendersi nelle epoche successive. Dopo queste flessioni, le prestazioni del modello si stabilizzano e continuano a migliorare gradualmente fino alla 130^a epoca. Nonostante le fluttuazioni iniziali, il modello dimostra una notevole capacità di adattamento e di recupero, riuscendo infine a mantenere un buon equilibrio tra accuratezza e completezza nelle fasi successive dell'addestramento.

Epoche	Mean Precision	Mean Recall	Mean F1
Base	0,5644	0,6962	0,6229
1	0,5865	0,5975	0,5912
2	0,4956	0,5023	0,4985
3	0,0946	0,0864	0,0901
4	0,1937	0,1826	0,1875
5	0,6457	0,6292	0,6364
6	0,7084	0,6879	0,6967
80	0,7782	0,7410	0,7586
90	0,7821	0,7589	0,7699
100	0,7821	0,7551	0,7679
110	0,7809	0,7523	0,7658
120	0,7852	0,7578	0,7707
130	0,7884	0,7588	0,7729

Tabella 5.1: **Prestazioni del Modello GPT-2 nelle 130 epoche.**
 La tabella mostra l'evoluzione delle prestazioni del modello all'aumentare del numero di epoche di allenamento, evidenziando un calo iniziale, con una successiva ripresa fino all'ultima epoca rispetto al modello di base.

Domanda:	Che dispositivi posso utilizzare per monitorare e registrare la mia frequenza cardiaca?
Epoche	Risposta
Modello base	L'agopuntura è l'apparecchio a cui vengono posizionati tutte le informazioni relative al tuo paziente; si basa sull'uso del suo midollo osseo o nella sua funzione vasocostrizione oppure sul sistema immunitario come indicato dall'autore della rivista <i>La Nuova American Journal of Medical Sciences</i> ; pubblicato nel 1978 dalla National Association of Agriculture and Biologics negli Stati Uniti d'America....
Dataset	Esistono diversi dispositivi che puoi utilizzare per monitorare e registrare la tua frequenza cardiaca, come i misuratori di pressione arteriosa automatici, gli orologi fitness con sensori ottici e le fasce toraciche che utilizzano la tecnologia Bluetooth per trasmettere dati direttamente al tuo smartphone o ad un computer.
1	la pressione arteriosa è una pressione arteriosa che va incontro a un'alterazione di pressione arteriosa. La pressione arteriosa è una pressione arteriosa che va incontro a un'alterazione di pressione arteriosa. La pressione arteriosa è una pressione arteriosa che va incontro a un'alterazione di pressione arteriosa...
2	la pressione arteriosa è una pressione arteriosa che va trasmessa da un sistema nervoso centrale, che controlla la pressione arteriosa. La pressione arteriosa è una pressione che va trasmessa da un sistema nervoso centrale, che controlla la pressione arteriosa. La pressione arteriosa è una pressione che va trasmessa da un sistema nervoso centrale, che controlla la pressione arteriosa.
3	NaN
4	NaN
5	L'apparecchio è un dispositivo di misurazione della pressione arteriosa, che permette di misurare la frequenza cardiaca, la pressione sanguigna e la pressione sanguigna.
6	Sì utilizzare dispositivi per monitorare e registrare la frequenza cardiaca, come un display a cristalli liquidi, un display a cristalli liquidi o un display a cristalli liquidi.
80	Per monitorare la tua frequenza cardiaca, è importante utilizzare dispositivi di misura come dispositivi di misuratore, che possono essere più affidabili e più adattabili per gestire la pressione arteriosa.

Epoche	Risposta
90	Esistono diversi dispositivi che possono essere utilizzati per monitorare la tua frequenza cardiaca, tra cui dispositivi per misurare la frequenza cardiaca, che possono essere utilizzati per misurare la tua frequenza cardiaca, come un dispositivo per misurare la frequenza di un braccio o un braccio a terra.
100	Esistono diversi dispositivi che possono essere utilizzati per monitorare la frequenza cardiaca, tra cui dispositivi di misuratore, che possono essere utilizzati per misurare la frequenza cardiaca, o per misurare la frequenza di un medico.
110	Esistono diversi dispositivi che possono essere utilizzati per monitorare la tua frequenza cardiaca, tra cui dispositivi di misura come il telelaser, che possono essere utilizzati per misurare la tua frequenza cardiaca.
120	Esistono diversi dispositivi che possono essere utilizzati per monitorare la tua frequenza cardiaca, tra cui dispositivi di misuratore, che possono essere utilizzati per misurare la frequenza cardiaca.
130	Per monitorare la tua frequenza cardiaca, è importante utilizzare dispositivi come smartwatch, che consentono di misurare la frequenza cardiaca in modo automatico e indipendente.

Osservando le risposte generate dal modello durante il processo di fine-tuning, possiamo notare un chiaro riscontro con quanto descritto in precedenza riguardo alle metriche di precision, recall e F1. Il miglioramento delle prestazioni del modello è evidente, partendo da risposte incoerenti nel modello base fino ad arrivare a risposte sempre più pertinenti e utili nelle ultime epoche.

Nelle prime epoche, soprattutto fino alla 4^a, le risposte rimangono ripetitive e prive di senso, coerentemente con il brusco calo delle metriche. A partire dalla 5^a epoca, con la risalita della precision e della F1, si nota un graduale miglioramento nella qualità delle risposte, sebbene rimangano ancora imprecise e ripetitive fino alla 120^a epoca. Nonostante questo miglioramento rispetto alle epoche iniziali, le risposte non sono del tutto soddisfacenti, continuando a mostrare ripetizioni e mancanza di coerenza. Tuttavia, nell'ultima epoca, la 130^a, il modello raggiunge la sua miglior performance, con una precision di 0.7884 e un recall di 0.7588, fornendo risposte finalmente più coerenti e pertinenti, come dimostrato dalla menzione di dispositivi concreti come smartwatch.

Questo risultato finale conferma che, nonostante le fluttuazioni durante l'addestramento, il modello è riuscito a mantenere un buon equilibrio tra accuratezza e completezza, dimostrando la sua capacità di adattamento e miglioramento nel tempo.

5.8.2 Mistral-7B

Il modello di Mistral utilizzato è "*MaziyarPanahi/Mistral-7B-Instruct-Aya-101*", [18] una variante del modello Mistral-7B, addestrato specificamente per generare testo in un contesto di istruzione o assistenza. Questo modello è ottimizzato per rispondere a domande e generare contenuti informativi, ed è stato addestrato utilizzando un ampio dataset multilingue chiamato "Aya Dataset," che include dati in 101 lingue diverse.

Risultati dei test

I test sul modello, condotti dopo 6 epoche, hanno mostrato un calo costante della loss, che però è rimasta su valori troppo elevati. Anche le metriche di perplexity e token accuracy sono rimaste insoddisfacenti, indicando che il modello continua a incontrare difficoltà nel generare previsioni accurate nella configurazione attuale.

Le metriche utilizzate per il calcolo del BERTScore (Precision, Recall e F1) hanno registrato un aumento significativo, indicando un miglioramento parziale del modello durante l'addestramento, in particolare nella sua capacità di produrre risposte più coerenti e semanticamente rilevanti.

Di seguito i dettagli:

- **Loss**

Il grafico evidenzia una costante diminuzione della loss su tutti e tre i set di dati nel corso delle 6 epoche, indicando un miglioramento continuo nella capacità del modello di apprendere e predire con maggiore accuratezza.

In particolare, il training set (verde) mostra una rapida diminuzione nelle prime tre epoche, seguita da un calo più graduale fino alla sesta epoca. Le curve del validation set (arancione) e del test set (blu) seguono un andamento più costante, con valori leggermente più alti rispetto al training set, suggerendo che il modello sta adattandosi meglio ai dati.

Non ci sono quindi segnali chiari di overfitting, poiché la validation e test loss restano simili alla training loss, in particolare nelle ultime epoche, il che indica un buon equilibrio tra i vari set. Tuttavia, i valori di loss rimangono ancora troppo elevati per essere considerati ottimali, suggerendo che c'è margine per ulteriori miglioramenti.

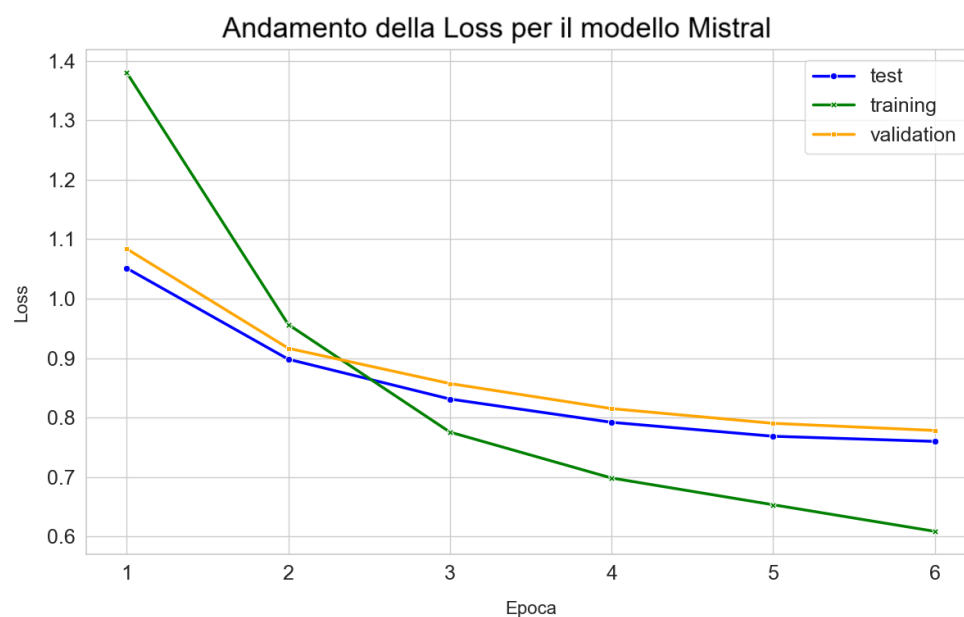


Figura 5.5: Andamento della loss durante le fasi di addestramento del modello Mistral7B.

Il grafico mostra una riduzione progressiva della loss su tutti i set di dati, mantenendo però valori troppo elevati.

- **Perplexity**

Il grafico mostra una diminuzione costante della perplexity su tutti i set di dati nel corso delle epoche, un segnale positivo che indica come il modello stia diventando progressivamente più sicuro nelle sue previsioni durante l'addestramento.

Infatti, nelle prime epoche si nota un divario significativo tra il training set (verde) e i set di validation e test, suggerendo che il modello inizialmente ha avuto difficoltà ad imparare dai dati di addestramento. Tuttavia, questo divario si riduce gradualmente con l'avanzare delle epoche, segno che il modello sta cominciando a comprendere meglio i pattern linguistici.

Inoltre i valori di validation set (arancio) e il test set (blu) rimangono molto vicini, suggerendo una buona capacità del modello di generalizzare sui dati non visti.

Nonostante questa riduzione, i valori assunti della perplexity restano ancora molto elevati. In contesti di generazione di testo, valori più bassi sarebbero preferibili, poiché valori così alti indicano che il modello fatica a produrre previsioni affidabili.

Pur mostrando segni di miglioramento, il modello non ha ancora raggiunto una performance ottimale.

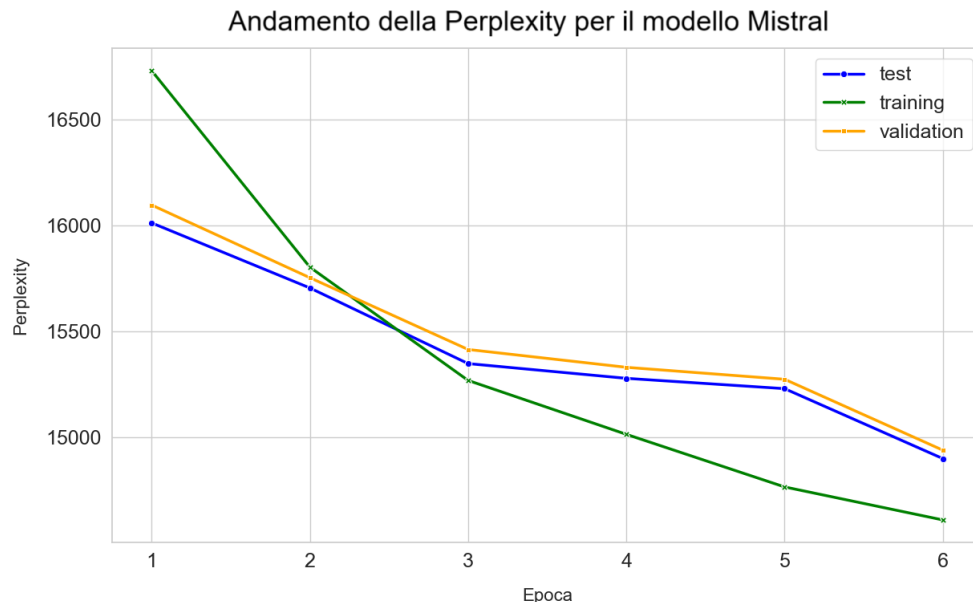


Figura 5.6: Andamento della perplexity durante le fasi di addestramento del modello Mistral7B.

Nel grafico si osserva un miglioramento costante in tutti i set, pur mantenendo un valore di perplexity troppo elevata.

- **Token Accuracy**

L'andamento dei valori di token accuracy per i set di test, training, e validation evidenzia aspetti prevalentemente problematici, legati alle prestazioni del modello.

I valori sono estremamente bassi per tutti i set, oscillando tra 0.0001 e 0.0002, il che indica che il modello sta prevedendo correttamente una quantità minima di token, e di conseguenza le sue previsioni complessive sono molto imprecise.

Nel training set (verde), la token accuracy varia nel corso delle epoche, senza però mostrare un chiaro trend di miglioramento.

Nei set di validation (arancio) e test (blu), i valori rimangono stabili, ma su livelli molto bassi, senza alcun segno di progresso.

Questa mancanza di miglioramento su tutti i set suggerisce che il modello non sta apprendendo correttamente i pattern dai dati, probabilmente a causa di underfitting.

Inoltre le oscillazioni osservate nel training set indicano che il modello non sta convergendo in modo stabile, sottolineando ulteriori difficoltà nell'addestramento.

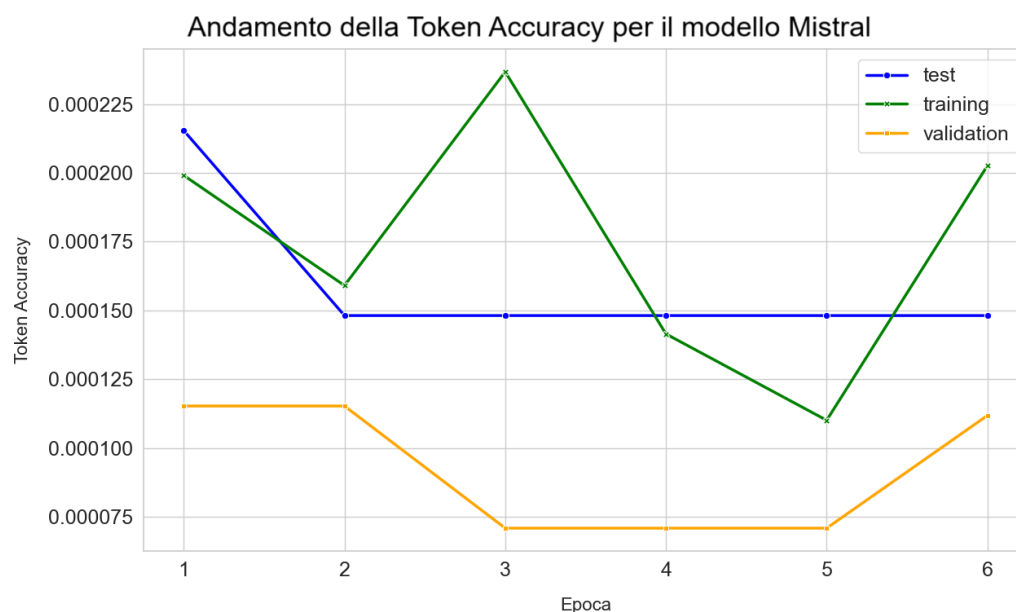


Figura 5.7: Andamento della token accuracy durante le fasi di addestramento del modello Mistral7B.

Il grafico evidenzia fluttuazioni nella token accuracy nel set di training, mentre i set di validation e test mostrano una grande stabilità. Questo andamento riflette l'instabilità del modello durante l'addestramento, con assenza di miglioramenti significativi nella capacità di predizione e nella precisione.

- **BERTScore**

Come evidenziato nella tabella, l'addestramento del modello Mistral ha portato ad un significativo miglioramento delle prestazioni rispetto al modello di base, con un chiaro incremento in tutte le metriche principali.

La Mean Precision è aumentata sensibilmente, dimostrando una maggiore accuratezza nelle risposte del modello. Tuttavia, alla quinta epoca si osserva un leggero calo, che potrebbe indicare una fluttuazione temporanea nel processo di addestramento. Infatti, il successivo recupero alla sesta epoca, con un incremento della precisione, suggerisce che il modello ha continuato a migliorare, superando una fase di instabilità momentanea.

D'altra parte, la Mean Recall ha continuato a migliorare costantemente, ma anch'essa ha subito un leggero calo alla terza epoca. Questo potrebbe significare che, sebbene il modello stia migliorando nella capacità di identificare correttamente i casi rilevanti, c'è stata una fase in cui non è riuscito a migliorare in modo significativo la copertura delle previsioni. Questo calo può essere legato a un piccolo momento di instabilità o a un problema temporaneo nell'ottimizzazione del modello.

Questi cali hanno influenzato anche il Mean F1, che ha seguito un andamento simile, indicando una temporanea perdita di equilibrio tra precision e recall.

Epoche	Mean Precision	Mean Recall	Mean F1
Base	0,5562	0,6950	0,6175
1	0,5883	0,6012	0,5923
2	0,8100	0,7889	0,7988
3	0,8130	0,7814	0,7963
4	0,8141	0,7955	0,8040
5	0,8131	0,7959	0,8038
6	0,8205	0,7987	0,8089

Tabella 5.2: Prestazioni del Modello Mistral7B nelle 6 epoche.

La tabella mostra l'evoluzione delle prestazioni del modello man mano che aumenta il numero di epoche di allenamento, evidenziando un miglioramento generale nelle metriche fino a 6 epoche rispetto al modello di base.

Domanda:	Potresti spiegarmi cosa significa avere la pressione alta?
Epoche	Risposta
Modello base	L'omeopatia è uno studio della medicina basata su studi clinici condotti a partire dai primi anni del secolo XIX presso l'Università degli Studi di Roma "La Sapienza", sulla base dei dati raccolti dall'Istituto Superiore di Sanità al fine di valutare le condizioni patologiche delle persone affette da il diabete mesotelico o analeidalita', nel periodo compreso tra il 1888 ed il 1918....
Dataset	Certamente, avere la pressione alta, o ipertensione, significa che l'antagonismo forza del sangue contro le pareti delle arterie è troppo elevata, il che può portare a problemi di salute come malattie cardiache.
1	# # # # # # # #
2	Sì, certo. La pressione alta significa che il cuore deve lavorare più duramente per pompare il sangue attraverso il tuo corpo, che può portare a problemi di salute come malattie cardiache, ictus e insufficienza renale.
3	L'ipertensione significa che la pressione del sangue nelle vene è troppo alta, che può portare a problemi di salute come malattie cardiache, ictus e insufficienza renale.
4	L' avere la pressione alta significa che il cuore deve lavorare più duramente per pompare il sangue attraverso il corpo, che può portare a problemi di salute come malattie cardiovascolari e ictus.
5	L'ipertensione significa che la pressione del sangue contro le pareti delle tue arterie è troppo alta, che può portare a danni ai vasi sanguigni e ai loro organi, come il cuore e i reni.
6	È importante capire che la pressione alta, o ipertensione, significa che il cuore lavora più duramente per pompare il sangue, mettendo a dura prova il sistema cardiovascolare e aumentando il rischio di malattie cardiache, ictus e insufficienza renale.

Come evidenziato nella tabella, l'addestramento del modello Mistral ha portato a un significativo miglioramento delle prestazioni rispetto al modello di base. Questo progresso è rispecchiato nelle risposte generate dal modello durante le diverse fasi di addestramento.

Nelle prime epoche, il modello ha mostrato difficoltà significative, con risposte completamente irrilevanti e prive di senso. Tuttavia, man mano che l'addestramento è proseguito, si è notato un miglioramento tangibile. Già dalla seconda epoca, le risposte sono diventate più pertinenti e strutturate, dimostrando che il modello iniziava a comprendere meglio il contesto.

La Mean Precision è aumentata sensibilmente durante le prime epoche, come si evince dalle risposte sempre più precise. Tuttavia, alla quinta epoca si osserva una leggera flessione, che si traduce in risposte meno accurate in termini di precisione, pur mantenendo una certa coerenza.

D'altra parte, la Mean Recall ha subito un piccolo calo alla terza epoca, con risposte meno fluide rispetto a quelle delle epoche successive, indicando che il modello ha faticato a recuperare tutte le informazioni rilevanti in quella fase. L'andamento incrementale dei risultati si nota in particolar modo nell'ultima risposta, che dimostra come il modello abbia compreso meglio il concetto, offrendo previsioni più precise e complete. Questo progresso corrisponde all'aumento costante delle metriche di Precision, Recall, e F1 score, evidenziando come il modello sia diventato progressivamente più accurato e dettagliato nelle sue risposte.

5.8.3 BioMistral-7B

Il modello utilizzato è "*BioMistral-7B*" [1], una variante altamente specializzata del modello Mistral 7B, progettata specificamente per il dominio medico. BioMistral-7B è stato ulteriormente addestrato su un vasto corpus di letteratura medica proveniente da PubMed Central, rendendolo particolarmente abile nel trattare terminologie, concetti e contesti medici. È stato sottoposto a una valutazione multilingue su larga scala per garantirne l'efficacia in diversi contesti, non limitandosi quindi alla lingua inglese.

Risultati dei test

I test effettuati sul modello, valutati su un totale di 6 epoche di addestramento, evidenziano, nel complesso un miglioramento parziale. Sebbene la loss e la perplexity siano diminuite costantemente, i loro valori restano ancora troppo elevati per indicare prestazioni ottimali. La token accuracy è rimasta molto bassa, suggerendo difficoltà nel fare previsioni accurate. Tuttavia, le metriche di BERTScore (Precision, Recall e F1) hanno raggiunto buoni livelli, indicando che il modello è riuscito a produrre risposte più coerenti e rilevanti nelle fasi successive dell'addestramento.

- **Loss**

Il grafico evidenzia una costante diminuzione della loss su tutti i set di dati, segnalando un miglioramento continuo nella capacità del modello di apprendere e ridurre l'errore nelle sue previsioni.

In particolare, la curva del training set (verde) mostra un rapido calo nelle prime epoche, suggerendo che il modello sta acquisendo rapidamente i pattern fondamentali. Questo calo prosegue in modo più graduale, segnalando che il modello sta perfezionando il suo apprendimento senza incorrere in overfitting.

Anche la loss del validation set (arancione) e del test set (blu) segue un andamento in discesa, mantenendo valori leggermente superiori rispetto al training set, come previsto. Questo indica una buona capacità di generalizzazione, applicando ciò che ha appreso ai dati non visti.

La vicinanza tra le curve di validation e test conferma la stabilità del modello nelle sue prestazioni predittive, dimostrando coerenza e precisione anche su dati nuovi.

Tuttavia, nonostante questa riduzione, i valori assunti della loss rimangono elevati, indicando che il modello continua a commettere errori significativi. Per una performance ottimale, dovrebbe diminuire ulteriormente, per garantire una maggiore accuratezza predittiva.

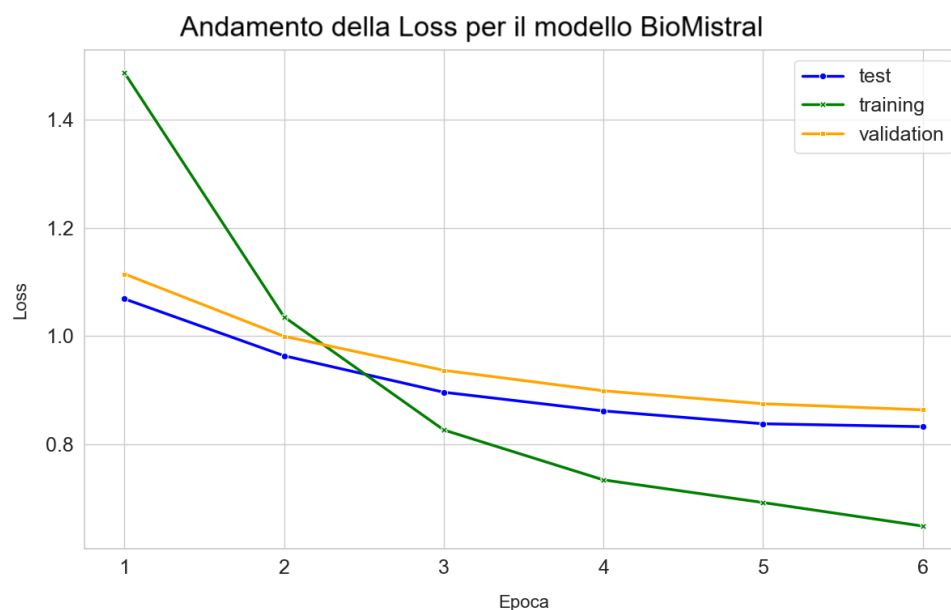


Figura 5.8: Andamento della loss durante le fasi di addestramento del modello BioMistral7B.

Il grafico evidenzia una diminuzione della loss su tutti i set, indicando un miglioramento nell'apprendimento del modello. Tuttavia, i valori rimangono ancora troppo elevati, indicando che il modello continua a commettere errori significativi.

- **Perplexity**

Il grafico evidenzia una riduzione significativa della perplexity su tutti i set, segnalando che il modello sta progressivamente migliorando la sua capacità di fare previsioni più sicure.

I valori della perplexity per il test set (blu) e il validation set (arancione) sono molto vicini, indicando che il modello generalizza bene e non ci sono segnali di overfitting.

Tuttavia, nonostante il miglioramento, i valori assunti della perplexity rimangono troppo elevati su tutti i set, segnalando che il modello fatica ancora a cogliere in modo efficiente i pattern linguistici presenti nei dati. Sarebbero quindi necessarie ulteriori ottimizzazioni per ridurre questi valori e migliorarne l'accuratezza.

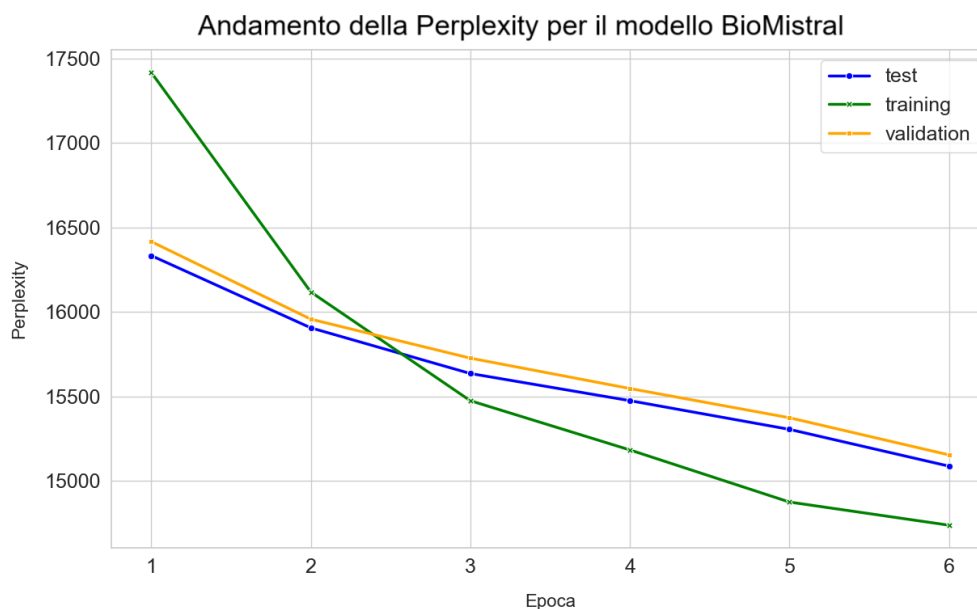


Figura 5.9: Andamento della perplexity durante le fasi di addestramento del modello BioMistral7B.

Il grafico evidenzia una significativa riduzione della perplexity su tutti i set, ma i valori restano comunque troppo elevati, indicando che il modello fatica ancora a cogliere in modo efficiente i pattern linguistici presenti nei dati.

- **Token Accuracy**

L'andamento dei valori di token accuracy in tutti i set presenta diversi aspetti problematici. I valori sono estremamente bassi, attorno allo 0.0001, indicando che il modello ha gravi difficoltà a fare previsioni accurate, compromettendo la sua capacità di cogliere i pattern nei dati.

Nel training set (verde), la token accuracy mostra un picco alla terza epoca, seguito da un calo nelle epoche successive. Queste oscillazioni segnalano un processo di apprendimento instabile, in cui il modello non riesce a mantenere i progressi fatti.

Nel test set (blu), si osserva un piccolo picco, seguito da una stabilizzazione su livelli comunque molto bassi.

Nel validation set (arancione), la token accuracy ha un picco iniziale, si stabilizza brevemente, ma poi cala nelle ultime epoche, evidenziando difficoltà nel mantenere buone performance su dati non visti.

I valori risultano quindi troppo bassi per considerare le prestazioni del modello accettabili.

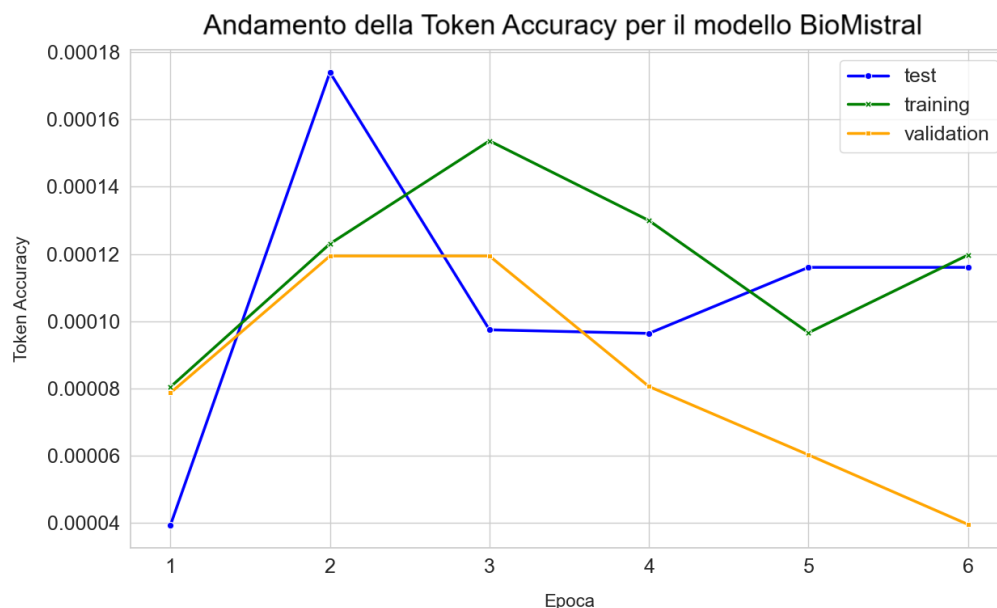


Figura 5.10: Andamento della token accuracy durante le fasi di addestramento del modello BioMistral7B.

Il grafico evidenzia fluttuazioni significative, segnalando un'instabilità del modello e problemi dovuti ai valori troppo bassi della token accuracy.

- **BERTScore**

Biomistral mostra un comportamento molto simile a Mistral.

La Mean Precision è aumentata sensibilmente nelle prime epoche, dimostrando una maggiore accuratezza nelle risposte del modello. Tuttavia, anche qui, alla quinta epoca si osserva un leggero calo, che può essere interpretato come una fluttuazione temporanea nel processo di addestramento. Il recupero alla sesta epoca, con un incremento della precisione, suggerisce che il modello ha superato questa fase di instabilità momentanea, continuando a migliorare.

Anche qua, la Mean Recall ha mostrato un miglioramento costante, con un lieve calo alla terza epoca, seguito da una pronta ripresa.

Questi cali si riflettono anche nel Mean F1, che a seguito di un andamento simile, evidenzia una temporanea perdita di equilibrio tra precision e recall, per poi recuperarla nelle epoche successive.

Epoche	Mean Precision	Mean Recall	Mean F1
Base	0,5562	0,6950	0,6175
1	0,7823	0,7534	0,7670
2	0,8074	0,7810	0,7934
3	0,8088	0,7440	0,7746
4	0,8180	0,7954	0,8060
5	0,8158	0,7939	0,8041
6	0,8751	0,8714	0,8732

Tabella 5.3: Prestazioni del modello Biomistral7B in 6 epoche.

La tabella mostra l'evoluzione delle prestazioni del modello man mano che aumenta il numero di epoche di allenamento, evidenziando un miglioramento generale nelle metriche fino a 6 epoche rispetto al modello di base.

Domanda:	Che suggerimenti puoi darmi riguardo alla mia ipertensione di oggi?
Epoche	Risposta
Modello base	Questa prova di empatia è particolarmente utile per valutare la coerenza delle prime frasi e la complessità delle successive. Da notare è il fatto che la dimostrazione di empatia (ad esempio “Senti come ti siano fatti un pezzo più grandi?”)....
Dataset	Mi dispiace sapere che stai avendo problemi con l'ipertensione oggi. È importante che tu cerchi di rilassarti e di ridurre lo stress, magari con tecniche di respirazione profonda o meditazione.
1	Suggerisci di monitorare regolarmente la tua pressione arteriosa e mantenere un stile di vita sano, con una dieta equilibrata, ricca di frutta, verdura e cereali integrali, e un'attività fisica regolare.
2	È importante mantenere uno stile di vita sano, come seguire una dieta equilibrata ricca di frutta, verdura e cereali integrali, limitare il consumo di sale e grassi saturi, e fare regolarmente esercizio fisico.
3	Per gestire l'ipertensione di oggi, è importante mantenere una dieta equilibrata, ridurre il consumo di sale e assicurati,.
4	È importante monitorare la pressione arteriosa regolarmente, mantenere una dieta equilibrata e ridurre il consumo di sale. Assicurati di fare esercizio fisico regolarmente e di evitare l'alcol e il tabacco.
5	È importante monitorare la tua pressione arteriosa regolarmente e mantenere i livelli sotto 140/90 mmHg. Assicurati di seguire una dieta sana, ricca di frutta, verdura e povera di sale. Evita l'alcol e il tabacco.
6	Per gestire l'ipertensione, è importante seguire una dieta equilibrata ricca di frutta, verdura e cereali integrali, riducendo il consumo di sale e grassi saturi. Inoltre, è importante mantenere un regolare ritmo di attività fisica, come camminare.

Dopo l'addestramento del modello Biomistral7B, si osserva un miglioramento significativo nella qualità delle risposte rispetto al modello di base, come evidenziato dalla tabella. Inizialmente, le risposte generate erano completamente irrilevanti e prive di coerenza, incapaci di comprendere il contesto delle domande. Con il progredire dell'addestramento, le risposte sono diventate più pertinenti e strutturate, mostrando una comprensione migliore del contesto e una maggiore accuratezza.

Alla terza epoca, tuttavia, si verifica un calo della Mean Recall, che si riflette in risposte incomplete, segnalando che il modello ha faticato a recuperare tutte le informazioni necessarie per fornire una risposta completa. Nonostante ciò, il modello mostra un rapido miglioramento nelle epoche successive, con risposte più ricche e articolate.

5.8.4 Llama-8B

Il modello Llama-8B "meta-llama/Meta-Llama-3.1-8B" [14] è una delle versioni più recenti della serie di modelli LLaMA sviluppata da Meta AI. Questo modello, appartenente alla terza generazione di LLaMA, è stato progettato per offrire capacità avanzate di generazione del linguaggio naturale su larga scala, grazie ad un numero di parametri aumentato a 8 miliardi.

Risultati

Durante il tentativo di addestramento del modello Llama-8B su Google Colab, sono emersi significativi problemi legati alla memoria GPU disponibile, impedendo il completamento del processo. Nella versione gratuita, la memoria GPU è piuttosto limitata, con una disponibilità che solitamente varia tra i 12 GB e i 16 GB. Tuttavia, il modello di Meta, essendo molto più grande rispetto ad altri modelli come BioMistral-7B, richiede una quantità di memoria decisamente superiore per poter gestire l'archiviazione dei pesi, dei gradienti e delle attivazioni durante l'addestramento.

Nello specifico, con oltre 8 miliardi di parametri, il modello ha facilmente superato la capacità di memoria disponibile, specialmente durante la fase di aggiornamento dei parametri del modello, dove è necessario memorizzare temporaneamente le attivazioni dei layer (info intermedie) per il calcolo dei gradienti. Questo ha portato a un errore "Out of Memory", come evidenziato nell'immagine 5.16, dove il sistema ha tentato senza successo di allocare ulteriori 112 MiB di memoria, esaurendo così le risorse disponibili. Questo evento sottolinea le sfide tecniche che si incontrano quando si lavora con modelli di grandi dimensioni su piattaforme con risorse hardware limitate.

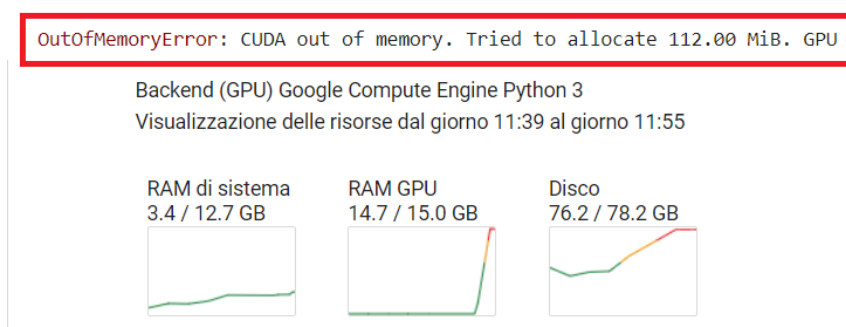


Figura 5.11: Allenamento del modello Llama 8B.

5.9 Discussione dei risultati

Dopo aver esplorato in dettaglio il complesso processo di sviluppo e fine-tuning degli LLM, giungiamo ora a una riflessione conclusiva sui risultati ottenuti. La seguente analisi finale ci consente di fare un confronto tra i vari modelli.

- **Loss.** Come osservato in precedenza, i modelli Mistral e BioMistral mostrano un andamento simile, con una diminuzione della loss su tutti i set, segnalando un miglioramento nell'apprendimento. Tuttavia, i valori restano troppo elevati, indicando che entrambi continuano a commettere errori significativi. Mistral evidenzia valori di loss leggermente inferiori, dimostrandosi così il modello con prestazioni migliori. D'altra parte, sebbene la loss diminuisca per GPT-2, i valori su tutti i set rimangono più elevati rispetto ai modelli precedenti, suggerendo problemi di instabilità, difficoltà nell'addestramento e underfitting, compromettendo così le prestazioni complessive del modello.

Epoche	Training		
	Mistral	BioMistral	GPT-2
1	1.38032	1.48649	5.04123
2	0.95654	1.03562	4.16069
3	0.77560	0.82688	3.74203
4	0.69840	0.73469	3.58647
5	0.65342	0.69283	3.46349
6	0.60870	0.64938	3.38259
80	-	-	2.52780
90	-	-	2.36804
100	-	-	2.38145
110	-	-	2.35660
120	-	-	2.24878
130	-	-	2.28965

Epoche	Validation			Test		
	Mistral	BioMistral	GPT-2	Mistral	BioMistral	GPT-2
1	1.08388	1.11494	3.93882	1.05127	1.06888	3.93093
2	0.91656	0.99976	3.65496	0.89815	0.96384	3.63408
3	0.85727	0.93698	3.50546	0.83120	0.89650	3.47305
4	0.81511	0.89918	3.28798	0.79206	0.86205	3.27069
5	0.79037	0.87526	3.19569	0.76862	0.83827	3.17142
6	0.77839	0.86399	3.12904	0.75994	0.83295	3.10424
80	-	-	2.37796	-	-	2.37198
90	-	-	2.33510	-	-	2.33452
100	-	-	2.29943	-	-	2.29777
110	-	-	2.27286	-	-	2.26962
120	-	-	2.24476	-	-	2.24305
130	-	-	2.21929	-	-	2.21537

Tabella 5.4: Confronto della Loss tra i modelli.

- **Perplexity.** Anche in questo caso, i modelli Mistral e BioMistral mostrano un comportamento simile, con una riduzione della perplexity durante l'addestramento, ma mantenendo valori ancora troppo elevati per i contesti di generazione di testo, segnalando difficoltà nel produrre previsioni affidabili.

Nonostante i segni di miglioramento, nessuno dei due modelli ha raggiunto una performance ottimale. Tuttavia, Mistral presenta valori di perplexity leggermente inferiori rispetto a BioMistral, confermandosi il modello con le prestazioni migliori.

GPT-2, invece, presenta valori di perplexity molto più alti, suggerendo che, a causa della sua struttura più semplice, potrebbe non essere adatto al compito senza un'ulteriore ottimizzazione.

Epoche	Training		
	Mistral	BioMistral	GPT-2
1	16729	17417	25989
2	15802	16117	25166
3	15268	15473	24456
4	15013	15182	24041
5	14765	14873	23315
6	14610	14736	23226
80	-	-	20322
90	-	-	20496
100	-	-	19932
110	-	-	19984
120	-	-	19900
130	-	-	20058

Epoche	Validation			Test		
	Mistral	BioMistral	GPT-2	Mistral	BioMistral	GPT-2
1	16095	16415	24462	16011	16332	29322
2	15752	15956	23735	15704	15905	29336
3	15414	15725	23282	15347	15634	29334
4	15329	15545	22636	15277	15473	29361
5	15273	15372	22420	15229	15303	29353
6	14939	15152	22197	14899	15086	29348
80	-	-	19567	-	-	19470
90	-	-	19402	-	-	19310
100	-	-	19320	-	-	19231
110	-	-	19187	-	-	19113
120	-	-	19061	-	-	18985
130	-	-	18974	-	-	18892

Tabella 5.5: Confronto della Perplexity tra i modelli.

- **Token Accuracy.** GPT-2, sorprendentemente, mostra una token accuracy più alta rispetto agli altri due modelli, pur mantenendo valori complessivamente bassi.

I modelli Mistral e BioMistral, invece, si dimostrano instabili e con valori eccessivamente bassi. Tuttavia, Mistral presenta valori leggermente superiori rispetto a BioMistral e un andamento meno instabile nei set di validation e test, dove, pur non mostrando miglioramenti significativi, i valori non tendono a diminuire come accade con BioMistral.

In ogni caso, nessuno dei tre modelli presenta risultati che possano essere considerati accettabili, mostrando difficoltà significative nel fare previsioni accurate, come confermato dalle metriche precedenti di perplexity e loss.

Epoche	Training		
	Mistral	BioMistral	GPT-2
1	0.00020	0.00008	0.01201
2	0.00016	0.00012	0.00600
3	0.00024	0.00015	0.00530
4	0.00014	0.00013	0.00520
5	0.00011	0.00010	0.00530
6	0.00020	0.00012	0.00520
80	-	-	0.00514
90	-	-	0.00506
100	-	-	0.00500
110	-	-	0.00512
120	-	-	0.00502
130	-	-	0.00498

Epoche	Validation			Test		
	Mistral	BioMistral	GPT-2	Mistral	BioMistral	GPT-2
1	0.00012	0.00008	0.00521	0.00022	0.00004	0.00540
2	0.00012	0.00012	0.00512	0.00015	0.00017	0.00520
3	0.00007	0.00012	0.00533	0.00015	0.00010	0.00526
4	0.00007	0.00008	0.00495	0.00015	0.00010	0.00504
5	0.00007	0.00006	0.00551	0.00015	0.00012	0.00540
6	0.00011	0.00004	0.00558	0.00015	0.00012	0.00537
80	-	-	0.00498	-	-	0.00504
90	-	-	0.00495	-	-	0.00504
100	-	-	0.00498	-	-	0.00504
110	-	-	0.00495	-	-	0.00504
120	-	-	0.00495	-	-	0.00504
130	-	-	0.00495	-	-	0.00507

Tabella 5.6: Confronto della Token Accuracy tra i modelli.

- **BERTScore.** Nel complesso, tutti e tre i modelli hanno mostrato un miglioramento significativo durante il fine-tuning rispetto al modello di base, ma ciascuno ha seguito un percorso diverso. GPT-2 ha dimostrato una buona capacità di adattamento e recupero: nonostante un calo significativo nelle prime epoche, è riuscito a mantenere un trend positivo nei test successivi. Mistral ha registrato un forte miglioramento iniziale, superando BioMistral già dalla seconda epoca. Nonostante una leggera instabilità nella terza e quinta epoca, il modello si è ripreso, con un incremento sia della precision che del recall, dimostrando una buona capacità di recupero. BioMistral7B, dopo progressi iniziali promettenti, è stato superato da Mistral nelle epoche centrali. Tuttavia, dalla quarta epoca in poi ha mostrato un netto recupero, nonostante le instabilità, migliorando costantemente nelle metriche di precision e recall, fino a raggiungere i risultati migliori e dimostrandosi il modello con le performance complessive più elevate.

Epoche	Mistral		BioMistral		GPT-2	
	Precision	Recall	Precision	Recall	Precision	Recall
1	0.588	0.601	0.782	0.753	0.587	0.598
2	0.810	0.789	0.807	0.781	0.496	0.502
3	0.813	0.781	0.809	0.744	0.095	0.086
4	0.814	0.796	0.818	0.795	0.194	0.183
5	0.813	0.796	0.816	0.794	0.646	0.629
6	0.821	0.799	0.875	0.871	0.710	0.716
80	-	-	-	-	0.778	0.741
90	-	-	-	-	0.782	0.759
100	-	-	-	-	0.782	0.755
110	-	-	-	-	0.781	0.752
120	-	-	-	-	0.785	0.758
130	-	-	-	-	0.788	0.759

Tabella 5.7: Confronto del BERTScore tra i modelli.

In conclusione, le performance complessive di tutti i modelli non sono risultate ottimali, specialmente in termini di loss, perplexity, e token accuracy. Tuttavia, i risultati ottenuti dal BERTScore sono stati incoraggianti, soprattutto se confrontati con le prestazioni iniziali.

Possiamo affermare che Mistral si distingue come il modello con la migliore efficienza complessiva, specialmente per quanto riguarda loss, perplexity, e token accuracy, mentre BioMistral eccelle nel BERTScore, grazie alla sua specializzazione nel dominio medico. Questa differenza è probabilmente dovuta al fatto che BioMistral è stato pre-addestrato su testi specifici del settore medico, permettendogli di catturare meglio le sfumature semantiche e generare risposte più naturali e contestualmente appropriate.

Infine, GPT-2, a causa della sua architettura meno complessa e del numero ridotto di parametri, ha mostrato diverse limitazioni rispetto ai modelli più grandi. Il BERTScore ha subito un declino, recuperando solo nelle ultime epoche, ma senza raggiungere i risultati ottenuti nelle prime epoche da Mistral e BioMistral. Questi fattori suggeriscono che, con le risorse limitate tipiche degli strumenti open source, GPT-2 potrebbe non essere adeguato per compiti complessi e richiederebbe risorse più avanzate per un miglioramento significativo.

Ovviamente, anche modelli come Mistral e BioMistral, per raggiungere performance ottimali, avrebbero bisogno di risorse più avanzate rispetto a quelle open source utilizzate. L'ottimizzazione su larga scala e l'utilizzo di infrastrutture più potenti potrebbero permettere loro di esprimere al meglio il loro potenziale, superando le attuali limitazioni.

Di seguito viene mostrata una comparazione delle risposte dei modelli per evidenziare quanto discusso in precedenza.

<hr/> Domanda: Quali informazioni puoi fornirmi sui monitoraggi per l'ipertensione? <hr/> <hr/>	
Modello	Risposta
Mistral	Monitoraggi regolari sono fondamentali per gestire l'ipertensione. Misura la pressione arteriosa a casa utilizzando un bracciale sanguigno validato e segui le indicazioni del tuo medico per il monitoraggio.
Biomistral	Il monitoraggio per l'ipertensione è fondamentale per gestire la salute cardiovascolare. Utilizza un misuratore di pressione affidabile e rileva la pressione sanguigna in repousse, preferibilmente alla stessa ora ogni giorno.

Capitolo 6

Conclusione

Il progetto si basa sull'applicazione di modelli di linguaggio di grandi dimensioni (LLM) nel settore medico, con un focus specifico sulla gestione dell'ipertensione. Questa condizione cronica rappresenta una sfida significativa per milioni di pazienti in tutto il mondo, richiedendo un monitoraggio costante e una gestione continua per prevenire complicanze gravi. Pertanto, l'accesso a strumenti di supporto efficaci e facilmente accessibili è cruciale.

Si è scelto di utilizzare risorse open source per i numerosi vantaggi che offrono, come la riduzione dei costi e una maggiore flessibilità. Se eseguiti localmente infatti, questi modelli garantiscono una maggiore tutela della privacy, poiché i dati sensibili possono essere gestiti internamente senza dover essere trasmessi a terze parti. Questo aspetto è particolarmente rilevante nel settore medico, dove il rispetto delle normative sulla privacy è fondamentale. È importante sottolineare che i grandi modelli di mercato, come quelli offerti da aziende commerciali tramite API, non sempre offrono lo stesso livello di garanzia in termini di privacy. Questo accade perché i dati devono essere inviati a server esterni, il che può esporre le informazioni a potenziali rischi di sicurezza, soprattutto in ambiti sensibili come quello sanitario.

Inizialmente, si è sperimentato con modelli di dimensioni ridotte, come GPT-2 small, il quale ha permesso di allenare su un numero "elevato" di epoche. Tuttavia, i risultati ottenuti sono stati deludenti, evidenziando le limitazioni intrinseche date dall'architettura semplice di GPT-2. Per superare queste difficoltà, si è deciso di passare a modelli più avanzati e di dimensioni maggiori, come Mistral7b e BioMistral7b, che offrono una capacità di apprendimento superiore e una maggiore adattabilità al contesto medico. Tuttavia, nonostante le potenzialità di questi modelli, il processo di training è stato limitato a poche epoche, a causa delle risorse computazionali limitate, portando a risultati non ottimali.

Questo si riflette chiaramente nelle metriche di valutazione: la loss ha mostrato valori significativamente superiori a quelli considerati ottimali, suggerendo una difficoltà del modello nel ridurre l'errore durante l'apprendimento. Allo

stesso modo, la perplexity è risultata estremamente elevata, ben oltre il range accettabile di 50-100, evidenziando problemi nel generare previsioni coerenti e nell'apprendere correttamente i pattern linguistici. Infine, la token accuracy è rimasta bassa, raggiungendo un massimo dell'1,20% nel modello con la performance migliore, ben al di sotto del valore atteso di circa il 70% per un modello ben addestrato a svolgere questo tipo di compito.

Questi risultati indicano che, sebbene l'utilizzo di risorse open source offra vantaggi in termini di costi e privacy, le attuali limitazioni computazionali rendono difficile il loro impiego in contesti clinici avanzati.

La sfida rimane quindi quella di trovare un equilibrio tra accessibilità, privacy, potenza computazionale e costi limitati per ottenere modelli più performanti e adatti alle esigenze del settore medico.

Come prospettiva per lavori futuri, si potrebbe considerare l'utilizzo di strumenti più avanzati per il training, come cluster con GPU, per migliorare le capacità computazionali. Con un'infrastruttura più performante, sarebbe possibile sfruttare appieno i modelli di grandi dimensioni, ottenendo risultati significativamente migliori in termini di velocità di addestramento e accuratezza delle risposte. Ciò consentirebbe di ottimizzare l'uso delle risorse computazionali e di affrontare task più complessi, superando le limitazioni attuali.

L'obiettivo finale sarebbe la creazione di un chatbot altamente specializzato, progettato per la gestione dell'ipertensione, capace di fornire risposte accurate e personalizzate grazie alle tecnologie più avanzate. Un ulteriore passo potrebbe essere l'integrazione di dati clinici specifici per migliorare l'accuratezza e l'affidabilità delle risposte, aprendo la strada a una vera assistenza digitale nel settore medico.

Bibliografia

- [1] BioMistral. Biomistral-7b. <https://huggingface.co/BioMistral/BioMistral-7B>.
- [2] Marco Cascella, Jonathan Montomoli, Valentina Bellini, and Elena Bignami. Evaluating the feasibility of chatgpt in healthcare: An analysis of multiple clinical and research scenarios. *Journal of Medical Systems*, 47(1):33, March 2023.
- [3] Marco Cascella, Federico Semeraro, Jonathan Montomoli, Valentina Bellini, Ornella Piazza, and Elena Bignami. The breakthrough of large language models release for medical applications: 1-year timeline and perspectives. *Journal of Medical Systems*, 48(1):22, 2024.
- [4] Chambers and Partners. Copyright vs. chatgpt: Unravelling the legal dilemma. *Chambers and Partners*, 2023. Accessed: September 2023.
- [5] Murali e Vishwanath Venkatram e Somani Arun K. Chitty-Venkata, Krishna Teja e Emami. Neural architecture search for transformers: A survey. *Accesso IEEE*, 10.
- [6] Jan Clusmann, Fiona R. Kolbinger, Hannah Sophie Muti, Zunamys I. Carrero, Jan-Niklas Eckardt, Narmin Ghaffari Laleh, Chiara Maria Lavinia Löffler, Sophie-Caroline Schwarzkopf, Michaela Unger, Gregory P. Veldhuizen, Sophia J. Wagner, and Jakob Nikolas Kather. The future landscape of large language models in medicine. *Communications Medicine*, 3(1):141, 2023.
- [7] Hugging Face. Perplexity — transformers documentation. <https://huggingface.co/docs/transformers/perplexity>, 2024.
- [8] GroNLP. gpt2-small-italian-embeddings. <https://huggingface.co/GroNLP/gpt2-small-italian-embeddings>.
- [9] Deepali Joshi, Aryan Shinde, Shreya Das, Om Deokar, Dipasha Shetiya, and Simran Jagtap. Text data augmentation. In *2023 International Conference on Advances in Computation, Communication and Information Technology (ICAICCIT)*, pages 392–396, 2023.

- [10] Bumhwi Kim and Eunryung Lee. Medical artificial intelligence framework for the development of medical imaging artificial intelligence devices. In *2022 13th International Conference on Information and Communication Technology Convergence (ICTC)*, pages 2210–2212, 2022.
- [11] Yann LeCun. The present and future of artificial intelligence, 2023.
- [12] John McCarthy. The quarterly journal of mechanics and applied mathematics. *Automata studies*, 1955.
- [13] John McCarthy. What is artificial intelligence? *Stanford University Website*, 2006.
- [14] Meta. Meta-llama 3.1 8b. <https://huggingface.co/meta-llama/Meta-Llama-3.1-8B>.
- [15] Sara Montagna, Stefano Ferretti, Lorenz Cuno Klopfenstein, Antonio Florio, and Martino Francesco Pengo. Data decentralisation of llm-based chatbot systems in chronic disease self-management. In *Proceedings of the 2023 ACM Conference on Information Technology for Social Good*, page 205–212. Association for Computing Machinery, 2023.
- [16] Prateeth Nayak, David Zhang, and Sek Chai. Bit efficient quantization for deep neural networks. In *2019 Fifth Workshop on Energy Efficient Machine Learning and Cognitive Computing - NeurIPS Edition (EMC2-NIPS)*, pages 52–56, 2019.
- [17] Andrew Ng. Ai is the new electricity, 2017. Stanford University AI Conference.
- [18] Maziyar Panahi. Mistral-7b-instruct-aya-101. <https://huggingface.co/MaziyarPanahi/Mistral-7B-Instruct-Aya-101>.
- [19] Karan Singhal, Shekoofeh Azizi, Tao Tu, S. Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, Perry Payne, Martin Seneviratne, Paul Gamble, Chris Kelly, Abubakr Babiker, Nathanael Schärli, Aakanksha Chowdhery, Philip Mansfield, Dina Demner-Fushman, Blaise Agüera y Arcas, Dale Webster, Greg S. Corrado, Yossi Matias, Katherine Chou, Juraj Gottweis, Nenad Tomasev, Yun Liu, Alvin Rajkomar, Joelle Barral, Christopher Semturs, Alan Karthikesalingam, and Vivek Natarajan. Large language models encode clinical knowledge. *Nature*, 620(7972):172–180, 2023.
- [20] Steve Sweetman. Introducing o1: Openai’s new reasoning model series for developers and enterprises on azure. *Microsoft Azure Blog*, 2024. Accessed: 2024-09-14.

-
- [21] Nima Tajbakhsh, Jae Y. Shin, Suryakanth R. Gurudu, R. Todd Hurst, Christopher B. Kendall, Michael B. Gotway, and Jianming Liang. Convolutional neural networks for medical image analysis: Full training or fine tuning? *IEEE Transactions on Medical Imaging*, 35(5):1299–1312, 2016.
 - [22] Alan Turing. Computing machinery and intelligence. *Mind*, 1950.
 - [23] Dave Van Veen, Cara Van Uden, Louis Blankemeier, et al. Adapted large language models can outperform medical experts in clinical text summarization. *Nature Medicine*, 2024.

Ringraziamenti

Ed eccoci qua, alla fine di questo percorso.

È stato un cammino davvero impegnativo, fatto di momenti di euforia e altri di sconforto, di giornate felici e altre più difficili, e di tanti sacrifici fatti per conciliare al meglio studio e lavoro.

Oggi, però, vedo tutto questo come una sfida che sono riuscita a superare, una sfida che mi ha permesso non solo di imparare nuove cose, ma anche di scoprire lati di me stessa che non conoscevo.

In particolare, questo lavoro mi ha dato l'opportunità di esplorare il mondo dell'intelligenza artificiale, una tecnologia tanto affascinante quanto complessa, ricca di concetti stimolanti. Fin da subito, mi ha appassionato profondamente e, dopo questa esperienza così intensa, mi piacerebbe approfondire ulteriormente questo campo, con l'obiettivo di specializzarmi.

E quindi un ringraziamento speciale va alla mia relatrice, Sara Montagna, e al mio correlatore, Matteo Magnini, che con i loro consigli mi hanno aiutato a orientarmi e a scegliere la strada giusta, dopo tanti tentativi andati a vuoto.

Un grazie enorme va anche alla mia famiglia, che mi ha sempre sostenuto e che ha sempre cercato di aiutarmi concretamente nei momenti più difficili.

Infine, un grazie profondo va a Leandro, mio compagno di vita, che mi è sempre stato accanto con infinita pazienza e amore. Mi hai sostenuto nei momenti più difficili, quando la stanchezza e lo sconforto prendevano il sopravvento e sembrava impossibile andare avanti.

Quindi Leandro, grazie di cuore per aver sempre creduto in me, anche quando io stessa dubitavo. Mi hai incoraggiato a superare i miei limiti, a non accontentarmi mai e a guardare sempre al futuro con fiducia. Senza il tuo supporto costante, la tua comprensione e il tuo amore incondizionato, non sarei riuscita ad affrontare questa sfida con la stessa forza.

GRAZIE.