

ALMA MATER STUDIORUM · UNIVERSITÀ DI BOLOGNA

SCUOLA DI SCIENZE

Corso di Laurea (Magistrale) in Matematica

INTELLIGENZA ARTIFICIALE
TRA RICERCA, DIDATTICA
E COMUNICAZIONE

Tesi di Laurea in Didattica della Matematica

Relatore:
Chiar.mo Prof.
Benvenuti Silvia

Presentata da:
Bastianini Michele

Anno Accademico 2023-2024

"Transire suum pectus mundoque potiri"

Archimede

Introduzione

Negli ultimi anni il tema delle intelligenze artificiali è tornato al centro del dibattito quotidiano in seguito alla pubblicazione online del chatbot ChatGPT, capace di generare risposte in linguaggio naturale in modo autonomo. La fruibilità di un'AI capace di rispondere a domande in maniera umana e generare porzioni di testo pone le basi per un dibattito sulla didattica.

Scopo di questa tesi è introdurre l'argomento delle intelligenze artificiali: si è deciso di iniziare parlando della loro storia, dal 1943 con l'invenzione del perceptrone fino al 2023 con l'introduzione di ChatGPT, passando per DeepBlue e AlphaGO, i due supercomputer che sono stati capaci di battere Garry Kasparov e Lee Sedol, rispettivamente a scacchi e Go.

Il secondo capitolo è stato dedicato alla comprensione delle strutture alla base delle reti neurali basandoci sulle pubblicazioni che le hanno definite, studiando il modello fondamentale del perceptrone per poi passare a modelli più complessi come quello del trasformatore, che definisce le deep neural network come quelle utilizzate per il famoso chatbot di OpenAI. L'ultima parte del secondo capitolo è stata dedicata al dibattito sui rischi e sui benefici che possono portare le intelligenze artificiali nell'ambiente didattico, analizzando le decisioni politiche di alcuni stati dell'Unione Europea e consultando diverse fonti. Tra le fonti consultate figura l'uscita di Settembre della rivista italiana Prisma, che si occupa di matematica in ogni suo aspetto, ponendo particolare attenzione a tre interviste a esperti del settore e docenti.

L'ultimo capitolo tratta della mia esperienza come divulgatore per conto di Calista al festival del gioco Play, tenutosi a Modena dal 17 al 19 maggio 2024, dove ho avuto l'occasione di portare un laboratorio didattico sulle intelligenze artificiali ed il gioco del

Go. Giocare delle partite contro il pubblico del Play mi ha permesso di sperimentare vari metodi per spiegare le regole del gioco del Go e di stimolare il ragionamento degli avversari. Nell'ultima parte del terzo capitolo si definiscono termini quali laboratorio didattico, situazione a-didattica e problem solving e si sostiene, attraverso alcune ricerche, l'ipotesi che un approccio ludico alla matematica possa favorire l'apprendimento e l'interesse degli studenti nei confronti della materia.

Indice

Introduzione	i
1 Storia dell’Intelligenza Artificiale	1
1.1 Il primo calcolatore meccanico e la prima Intelligenza Artificiale	1
1.2 Dal Turco Meccanico ad Alpha Go, passando per Deep Blue	3
2 Parte teorica	9
2.1 La teoria dietro il Machine Learning	9
2.2 Modello trasformatore per il deep learning	15
2.3 AI e didattica	18
3 La mia esperienza al Play	23
3.1 Play Festival del Gioco 2024	23
3.2 Approccio ludico al problem solving	28
Conclusioni	33
Bibliografia	35

Elenco delle figure

1.1	Cobb, William (2002). The Book of Go. Sterling Publishers. ISBN 978-0-8069-2729-9	5
2.1	Una rete neurale può essere rappresentata come un grafo orientato	9
2.2	Vari livelli di una rete neurale	10
2.3	Un neurone artificiale: il percettrone	11
2.4	Formula attivazione neurone	12
2.5	Esempio legge attivazione percettrone	13
2.6	Visualizzazione unidimensionale discesa del gradiente	14
2.7	Schema di un trasformatore	16
3.1	Classico esempio della regola del <i>ko</i>	25
3.2	Esempio di gruppi vivi e gruppi morti	26

Elenco delle tabelle

1.1	Complessità scacchi, shogi, go	4
-----	--	---

Capitolo 1

Storia dell'Intelligenza Artificiale

1.1 Il primo calcolatore meccanico e la prima Intelligenza Artificiale

L'era tecnologica in cui viviamo è caratterizzata dal rapporto uomo-macchina, generalmente improntato alla necessità dell'uomo di asservire la macchina ai suoi bisogni. Così la macchina è giunta a fare delle cose che risparmiano all'uomo tempo e fatica, o che addirittura sono per lui semplicemente impossibili, senza però mai mettere in dubbio la superiorità gerarchica del suo «padrone» umano [1].

Così inizia "I giocatori artificiali" di Paolo Ciancarini (1992), evidenziando come l'essere umano ha sempre tentato di automatizzare parte della propria vita: fra le prime "dimostrazioni" di questo fatto possiamo trovare la Macchina di Anticitera, databile tra il II e I secolo a.C. e rinvenuta nel 1900 sui fondali dell'omonima isola al largo della Grecia. Ritenuto il più antico calcolatore meccanico, è composto da diverse ruote dentate che messe in movimento permettono di calcolare le fasi solari e lunari, il movimento dei cinque pianeti allora conosciuti (Venere, Mercurio, Marte, Giove, Saturno) e le date dei giochi olimpici [2].

Per arrivare alla prima intelligenza artificiale dobbiamo fare un salto in avanti di qualche secolo fino al 1770 quando, alla Corte di Maria Theresa d'Austria, fece il suo ingresso il migliore giocatore di scacchi del suo tempo, rimasto imbattuto per 84 anni consecutivi. Si trattava appunto di un automa, denominato "il Turco meccanico" che, per i successivi

84 anni, diede prova delle sue abilità sulla scacchiera battendo ogni sfidante del pubblico, incluso celebrità dell'epoca come Napoleone Bonaparte e Benjamin Franklin.

Il Turco Meccanico era composto da due parti: un manichino con l'aspetto ed il vestiario di un uomo mediorientale, compreso di turbante, capace di muoversi e decidere le proprie mosse in autonomia grazie all'insieme di ingranaggi presenti all'interno del baule in legno sottostante. Prima dell'inizio di ogni partita veniva data prova dell'autenticità della macchina: il presentatore mostrava al pubblico l'interno della macchina aprendo i diversi sportelli, che permettevano di vedere una sezione all'interno del baule, mostrandone il contenuto di ingranaggi e leve. Spesso per dimostrare che la macchina non fosse influenzata da fenomeni magnetici veniva posizionata sulla parte superiore una grande calamita. Dopo diversi tour in tutto il mondo e diversi passaggi di proprietà, l'attenzione attorno al Turco Meccanico crebbe e con questa anche i dubbi e le accuse di truffa, dovuti anche ad un articolo di giornale scritto da Edgar Allan Poe [3].

Questo crescente sospetto generato attorno al Turco Meccanico portò John Kearsley Mitchell, all'epoca proprietario del marchingegno, a donarlo al Museo di Philadelphia, dove passò il resto dei suoi giorni fino al 5 luglio 1854, quando un incendio raggiunse il museo. Nonostante nessuno sia riuscito a dimostrare che il Turco Meccanico fosse effettivamente una truffa nel periodo in cui era in funzione, dopo l'incendio Silas Mitchell, figlio di John Mitchell, decise di rompere il silenzio pubblicando una serie di articoli che illustravano il trucco: come molti sospettavano, all'interno del marchingegno era presente un giocatore di scacchi umano.

Mentre il presentatore della macchina apriva gli sportelli per mostrare il meccanismo, il giocatore all'interno si spostava nella sezione nascosta dallo sportello stesso del baule, così da non esser visto. Dopo che il presentatore aveva mostrato l'interno della scatola ed il pubblico si era convinto che questa fosse vuota, il giocatore apriva un compartimento segreto accanto a se, contenente una scacchiera, una candela ed una leva. Questa leva era un pantografo modificato che permetteva al giocatore di muovere il braccio del Turco sulla scacchiera in tutte e tre le dimensioni dello spazio e di chiuderla così da afferrare i pezzi.

Sopra il giocatore era presente un ripiano segreto che, rimosso, dava la possibilità di vedere la parte inferiore della scacchiera posizionata sul tavolo, sotto ogni casella era

posizionato un piccolo magnete appeso ad un filo di rame, collegato magneticamente con ogni pezzo della scacchiera grazie ad un foglio metallico nascosto tra la scacchiera ed il ripiano della scatola in legno e dei magneti sotto ogni pezzo della scacchiera.

I magneti sopra cui non era presente alcuna pedina erano "a riposo", appesi per il filo di rame, mentre gli altri erano a contatto con il fondo della scacchiera.

Quando lo sfidante del Turco alzava una pedina per spostarla, il magnete corrispondente all'interno del baule si sarebbe allontanato dal fondo della scacchiera, mostrando così al giocatore segreto quale pedina si fosse mossa; una volta posizionata la pedina, il magnete a riposo si avvicinava alla scacchiera, mostrando al giocatore dove fosse stato posizionato il pezzo precedentemente alzato.

1.2 Dal Turco Meccanico ad Alpha Go, passando per Deep Blue

Dopo il Turco Meccanico, ci furono molti altri automi, fasulli e non, capaci di giocare a scacchi, anche se la successiva svolta nel mondo dei giocatori artificiali si ebbe nel 1997 quando il computer Deep Blue di IBM batté il campione del mondo di scacchi Garry Kasparov in un match "al meglio di sei" secondo le regole e le tempistiche utilizzate nei tornei ufficiali. Prima di questo epocale risultato ci furono alcuni tentativi infruttuosi, Deep Blue non è che il punto di arrivo di una storia durata più di dieci anni, composta da sconfitte e continui aggiornamenti e miglioramenti.

La costruzione della prima versione di Deep Blue iniziò nel 1985 con il nome di ChipTest all'Università Carnegie Mellon da parte del dottorando Feng-hsiung Hsu; due anni dopo si aggiudicò il primo premio al North American Computer Chess Championship. Nel 1988 il team di Hsu iniziò a lavorare sul successore di ChipTest, riuscendo a creare Deep Thought, una versione con due processori costruiti appositamente per l'occorrenza ed una maggiore velocità di ricerca. Il nome Deep Thought è un riferimento al supercomputer della serie di romanzi Guida galattica per gli autostoppisti di Douglas Adams, capace di fornire "*la risposta alla domanda fondamentale sulla vita, l'universo e tutto quanto*". Deep Thought si piazzò in prima posizione nel North American Computer Chess Championship dal 1988 al 1990, nonostante nel 1989 perse due partite contro Garry Kasparov:

campione del mondo di scacchi fino al 2000. La sconfitta con Kasparov evidenziò alcune criticità di Deep Thought che portarono alla creazione di Deep Blue, che riuscì dove il proprio predecessore aveva fallito, battendo Kasparov 3.5 a 2.5 su sei partite. Deep Blue oltre a ricevere aggiornamenti tecnici ricevette un'istruzione superiore rispetto a Deep Thought, aggiornando la propria lista delle aperture grazie al parere esperto di giocatori quali Joel Benjamin, Miguel Illescas, John Fedorowicz e Nick De Firmian. Dopo la sconfitta, Kasparov descrisse Deep Blue come un "avversario alieno" ma successivamente avanzò dubbi riguardo un possibile intervenuto umano in aiuto dell'elaboratore, data l'inusuale creatività nelle mosse della macchina, accusa smentita dal team di IBM, sottolineando come gli unici interventi umani non siano avvenuti durante le partite ma nell'intervallo fra queste [4].

Nonostante i sospetti di un nuovo Turco Meccanico, i passi in avanti nel mondo dell'informatica e dell'intelligenza artificiale erano innegabili. Successivamente a Deep Blue, molti ricercatori iniziarono a lavorare su computer in grado di competere con giocatori umani in giochi da tavolo, iniziando da giochi simili agli scacchi, come dama o shogi (l'equivalente giapponese degli scacchi).

Gioco	Scacchi	Shogi	Go
Dimensioni scacchiera	64	81	361
Numero pezzi	32	40	Fino a 361
Numero pezzi diversi	6	8	1
Posizioni possibili	10^{47}	10^{71}	10^{170}
Giochi possibili	10^{123}	10^{226}	10^{360}
Durata media partita (minuti)	70	115	150

Tabella 1.1: Complessità scacchi, shogi, go

Come possiamo vedere dalla tabella riportata qui sopra, shogi rappresenta un livello di complessità superiore rispetto agli scacchi, a causa del maggior numero di pezzi, di pedine diverse e delle dimensioni superiori del reticolo della scacchiera e conseguentemente del numero di posizioni possibili per ogni partita.

Lo sviluppo di giocatori artificiali proseguì per gradi andando a studiare giochi dalla complessità sempre maggiore: dopo lo shogi i ricercatori dedicarono le proprie energie a

costruire macchine capaci di giocare a Go. Il Go è un antico gioco da tavolo strategico per due giocatori, inventato in Cina più di 2500 anni fa. I due giocatori (rappresentati dai colori bianco e nero) posizionano a turno delle pedine (chiamate *pietre*) nelle intersezzioni di una scacchiera (detta *goban*) formata da una griglia 19×19 . Lo scopo del gioco è quello di catturare più territorio possibile, con la possibilità di catturare le pietre nemiche nel caso in cui si riescano a circondare (quindi "bloccare tutte le vie di ritirata" come mostrato in figura).

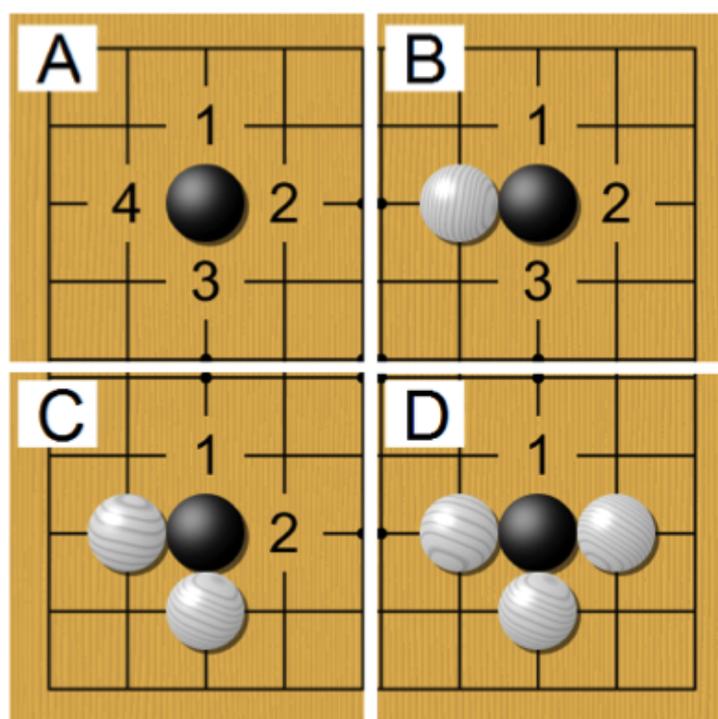


Figura 1.1: Cobb, William (2002). The Book of Go. Sterling Publishers. ISBN 978-0-8069-2729-9

Il Go è uno di quei giochi la cui difficoltà non risiede nelle regole (come ad esempio gli scacchi) ma nel gioco stesso: per questo riuscire a costruire un programma capace di giocare a Go rappresentava un importante passo avanti nella storia delle Intelligenze Artificiali. Il primo programma capace di giocare a Go è del 1968, scritto da Albert Lindsey Zobrist per la sua tesi sul riconoscimento di pattern; nei successivi 40 anni furono scritti altri programmi per il gioco del Go, ma spesso questi si restringevano a griglie

(*goban* in giapponese) più piccole rispetto a quelle considerate standard e, oltre al danno anche la beffa, questi programmi non erano nemmeno "eccelsi giocatori", riuscendo nel migliore dei casi a raggiungere un rango di 20-15 *kyu* solo attorno al 2003. Un punteggio superiore a 20 *kyu* lo possiamo attribuire a giocatori occasionali (volendo tracciare un parallelo con gli scacchi, un rango di 20 *kyu* corrisponde circa ad un punteggio di 100 nel sistema di valutazione Elo).

La maggiore difficoltà del gioco del Go rese necessario affrontare il problema con un'ottica totalmente nuova, un nuovo approccio che arrivò nel 2007 dopo la pubblicazione di un articolo da parte di Rémi Coulom dal titolo "*Efficient Selectivity and Backup Operators in Monte-Carlo Tree Search*" [5].

In questo articolo veniva introdotto un nuovo algoritmo che espande l'albero di ricerca delle mosse con risultato migliore con campionamento casuale dello spazio di ricerca grazie al metodo Monte Carlo. Con il nuovo algoritmo si sviluppa quindi un punto di osservazione diverso rispetto al gioco degli scacchi. Per gli scacchi il gioco aveva inizio consultando il libro delle aperture (una banca dati contenente le combinazioni di apertura, inserite da umani) e proseguiva scegliendo la mossa ritenuta migliore, decisa assegnando un punteggio grazie alle regole pari a +1, 0 o -1, a seconda che la disposizione dei pezzi sulla scacchiera sia vantaggiosa, neutra o svantaggiosa per il giocatore, fino alla fine della partita.

Per il gioco del Go non è sempre possibile determinare se una mossa sia vantaggiosa o meno per il giocatore: da qui l'ispirazione per il Monte-Carlo Tree Search (in seguito abbreviato con MCTS). Il programma sceglie un ramo dell'albero e gioca un numero determinato di partite che iniziano con quella mossa, selezionando le mosse successive in modo casuale; una volta completate le partite fa un "bilancio" tra partite vinte e partite perse, così da assegnare un punteggio "di qualità" alla mossa iniziale. Il punteggio assegnato (denominato "peso") rappresenta la probabilità di scegliere la mossa in questione in successive partite.

Grazie a questo nuovo algoritmo, nel 2009 un programma basato su MCTS ottenne il grado di primo *dan* (circa 2100 nel sistema di valutazione Elo) avvicinando il traguardo di un'Intelligenza Artificiale capace di giocare a Go ai livelli di un professionista. Nonostante il metodo MCTS abbia portato ad un sostanziale miglioramento delle capacità dei

programmi in grado di giocare a Go, risulta evidente come giocare un campione casuale di partite fino alla fine per stabilire la bontà di una mossa sia un processo lungo e non esaustivo, suggerendo la possibilità di miglioramento ed ottimizzazione degli algoritmi di ricerca della mossa migliore.

Il successivo traguardo fu raggiunto nel 2015 con l'avvento di AlphaGo, un programma rilasciato da Google DeepMind che univa al metodo MCTS metodi di *deep learning* utilizzando un vasto campione di partite giocate da umani e AI. L'idea di implementare il *deep learning* si rivelò vincente, tanto che nel corso di 500 partite contro altri programmi di GO, AlphaGo riuscì a vincere ogni partita eccetto una. Il team di Google DeepMind capitanato da David Silver dopo questo successo decise di puntare più in alto, sfidando il campione europeo di GO Fan Hui nell'ottobre 2015, vincendo cinque a zero, e successivamente il giocatore professionista coreano Lee Sedol, ritenuto al tempo il migliore giocatore di GO.

La partita AlphaGo vs. Lee Sedol si tenne dal 9 al 15 marzo 2016 e fu trasmessa in diretta davanti a centinaia di milioni di spettatori e commentata da giocatori 9 *dan* di Go: il grado massimo corrispondente a 2940 punti sulla scala Elo; le partite sono disponibili per intero sul canale YouTube di Google DeepMind insieme ad un documentario che analizza tutte e cinque le partite. Il risultato fu di quattro a uno per AlphaGo, con partite caratterizzate da mosse "creative" e "squisite" da parte di entrambi i giocatori (come commentato durante la diretta) e continui scambi di vantaggi. In seguito all'esito della sfida la Hanguk Kiwon (la federazione coreana di GO) assegnò ad AlphaGo il rango 9 *dan*.

La storia di AlphaGo però non si conclude con la vittoria contro Lee Sedol: il 19 ottobre 2017 il team di DeepMind pubblicò su *Nature* un articolo dove annunciava la creazione di AlphaGo Zero, una versione di AlphaGo allenata rimuovendo il fattore umano dal campione di *deep learning*; giocando solo partite contro se stesso, AlphaGo Zero è riuscito a raggiungere e superare le capacità dei suoi predecessori in circa 40 giorni. Quando un'AI apprende meglio e più rapidamente senza l'aiuto dell'essere umano, una diretta conclusione può essere quella di vedere il contributo umano come incompleto se non addirittura erroneo: sorge quindi la domanda filosofica sul perché giocare quando esistono programmi con qualità simil-divine. La risposta possiamo trovarla in un articolo uscito

su *arXiv.org* nel novembre 2022 dal titolo "*Adversarial Policies Beat Superhuman Go AIs*" [6] successivamente discusso all'International Conference on Machine Learning nel 2023 [7].

Nell'articolo Tony T. Twant et al. (2022) sono riusciti a trovare delle falle in KataGo (AI open source capace di battere AlphaGo il 99.9995% delle volte) ed a sfruttarle per vincere il 97% delle partite giocate. L'incredibile risultato qui sopra riportato dimostra che l'essere umano non è ancora da considerarsi totalmente obsoleto in confronto alle AI, anche se è probabile che il sorpasso avverrà in un futuro non così lontano.

Capitolo 2

Parte teorica

2.1 La teoria dietro il Machine Learning

L'AI è tornata molto a far parlare di se con l'uscita del chatbot ChatGPT a novembre del 2022 ma nonostante questo la nascita di questa disciplina può essere fatta risalire al 1956, quando al Dartmouth College nel New Hampshire si tenne un convegno per verificare i risultati ottenuti da un team di ricercatori formato l'anno precedente sul tema dell'intelligenza artificiale [8]. Dopo la conferenza si dedusse che "ogni aspetto dell'apprendimento o ogni altra caratteristica dell'intelligenza possa, in linea di principio, essere descritta così precisamente da poter creare una macchina capace di simularlo": uno dei primi passi fu quello di caratterizzare l'apprendimento umano e tentare di simularlo meccanicamente.

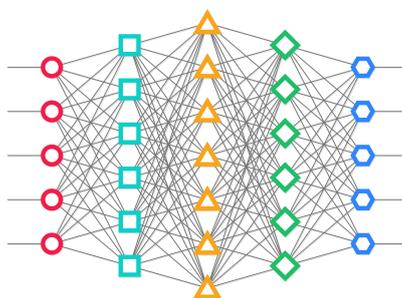


Figura 2.1: Una rete neurale può essere rappresentata come un grafo orientato

Nel 1957 presso il Cornell Aeronautical Laboratory vede la luce la prima macchina con rete neurale artificiale per opera di Frank Rosenblatt, denominata Mark I Perceptron, basata sul modello del percettrone [9]. Nel modello di percettrone venivano gettate le basi di una teoria della struttura cerebrale, questo fu diffuso nel 1943 da Warren McCulloch e Walter Pitts nell'articolo "A Logical Calculus of the Ideas Immanent in Nervous Activity" [10]; nonostante questo l'idea di applicarlo ad un apprendimento artificiale si ebbe solo intorno al 1955. Possiamo immaginare una rete neurale artificiale come un insieme di neuroni artificiali, ognuno dei quali capace di attivarsi nel caso in cui una determinata condizione venga raggiunta; questi neuroni sono divisi in livelli, ognuno dei quali comunica con il livello successivo ed è specializzato in un compito.

Per introdurre l'argomento, parleremo adesso di Feedforward Neural Network (abbreviate in FNN), ovvero reti neurali senza cicli. Il numero di neuroni e di livelli varia a seconda della rete neurale trattata e ordinando i neuroni in livelli possiamo dividere la rete neurale in *input layer*, *output layer* e *hidden layers*. Come riportato in figura, con *input layer* si intende il primo livello della rete, in cui questa riceve i dati in ingresso, con *output layer* identifichiamo l'ultimo livello della rete che fornisce i dati in output dopo aver elaborato i dati iniziali negli *hidden layers*: i livelli centrali.

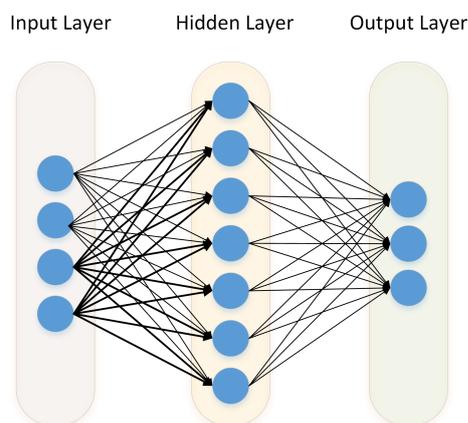


Figura 2.2: Vari livelli di una rete neurale

L'attivazione di un neurone in un livello comporta l'attivazione di altri neuroni nel livello successivo, come in un vero cervello umano; per capire però come avviene questo processo dobbiamo spiegare più nel dettaglio la loro composizione. I neuroni possono

essere interpretati come un oggetto che riceve in input alcuni valori binari (nella nostra immagine ad esempio, x_1 , x_2 e x_3) e restituisce un output denominato *attivazione* a seconda che questi siano superiori ad un determinato valore di soglia.

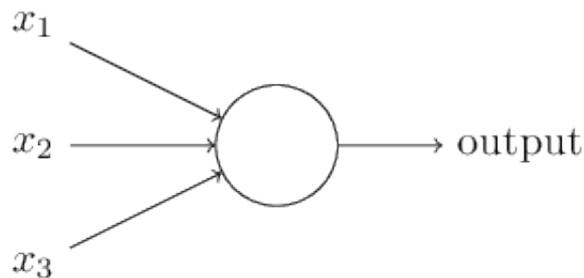


Figura 2.3: Un neurone artificiale: il perceptrone

I neuroni nei layer successivi hanno a loro volta una soglia di attivazione e ricevono in input i valori di ogni neurone del layer precedente, tarati in base alla loro importanza, denominata *peso*, w_1 , w_2 e w_3 ; questi perceptron si attivano se la somma pesata $\sum_{i=1}^n w_i a_i$ è superiore alla soglia di attivazione, dove con a_i intendiamo i valori in output dei neuroni del livello precedente. Da ciò deriva il fatto che un determinato perceptrone si attiva quando $\sum_{i=1}^n w_i a_i > t$ con t soglia di attivazione. Dal momento che avere disequazioni $> t$ risulta più scomodo di avere disequazioni > 0 , definiamo il *bias* $b := -t$.

Prendiamo ad esempio il primo neurone del secondo livello di una rete neurale, abbiamo indicato con a l'attivazione di ogni neurone (quindi il valore che trasmettono ai livelli successivi), con il numero ad apice indichiamo il livello della rete neurale mentre con il numero a pedice indichiamo il numero del neurone nel livello. Per i pesi, il primo numero a pedice rappresenta il neurone di arrivo, mentre il secondo numero rappresenta il neurone di partenza. Per il primo neurone del secondo livello possiamo quindi scrivere la sua funzione di attivazione come $w_{(0,0)}a_0^{(0)} + \dots + w_{(0,4)}a_0^{(4)} + b_0 > 0$; è importante ricordare come ogni neurone del secondo livello abbia una funzione di attivazione.

Nel nostro esempio, per i neuroni del secondo livello avremo un sistema di quattro disequazioni e per i livelli successivi un numero di disequazioni proporzionali al numero di neuroni del livello stesso, i cui parametri di attivazione dipendono dai risultati delle disequazioni del livello precedente.

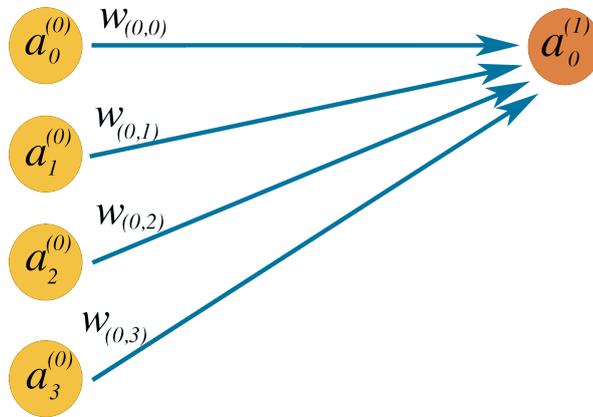


Figura 2.4: Formula attivazione neurone

Per semplificare questa notazione possiamo scrivere \mathbf{W} e \mathbf{A} come vettori colonna con componenti i pesi ed i valori di uscita di ogni perceptrone del livello precedente, rendendo così la somma pesata un prodotto scalare $\mathbf{W} \cdot \mathbf{A}$, definiamo inoltre il vettore colonna \mathbf{B} , contenente i bias di ogni neurone del livello. Mettendo insieme le nuove notazioni, per capire se un perceptrone si attiva, abbiamo la disequazione fra vettori $\mathbf{W} \cdot \mathbf{A} + \mathbf{B} > 0$. I perceptron quindi non sono altro che funzioni, che ricevono in input un array di valori e, in seguito al calcolo di $\mathbf{W} \cdot \mathbf{A} + \mathbf{B} > 0$, restituiscono in output un valore di 0 o 1.

Il principale problema di una rete di perceptron è che piccole variazioni in input possono portare a grandi variazioni in output: questa particolarità rende difficile lavorare con i perceptron, per questo motivo si usano spesso reti neurali basate su altri modelli di neuroni, che utilizzano funzioni particolari per restituire un valore in output meno suscettibile a variazioni in input. Alcune funzioni utilizzate a questo scopo sono la sigmoide, definita come $\sigma(x) = \frac{1}{1+e^{-x}}$, e la ReLU (Rectified Linear Unit), definita come $\text{ReLU}(x) = \max(0, x)$. Nell'immagine a pagina seguente possiamo vedere i valori di output delle tre funzioni esposte al variare dei valori in input.

Dopo aver analizzato la struttura di una rete neurale dobbiamo necessariamente parlare del suo funzionamento: possiamo iniziare domandandoci cosa significhi "apprendere" per una rete neurale. Per una rete neurale l'apprendimento è caratterizzato dal continuo aggiornare pesi e bias per minimizzare l'errore commesso.

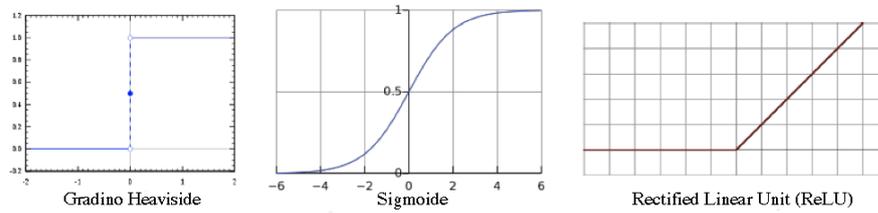


Figura 2.5: Esempio legge attivazione perceptrone

I primi passi verso questo risultato consistono nell'impostare dei pesi e dei bias iniziali (spesso scelti in modo random) e fornire alla rete un *training set*, ovvero un insieme di esempi sui quali la rete possa esercitarsi aggiustando pesi e bias. La rete è programmata per modificare pesi e bias in modo da minimizzare l'errore quadratico medio tra la risposta data e la risposta corretta: se rappresentiamo con y il vettore contenente gli output della rete e con z il vettore contenente la risposta corretta, possiamo scrivere $\frac{1}{2n} \sum_{i=1}^n (y_i - z_i)^2$, dove il 2 a denominatore serve per semplificare i calcoli quando andremo a derivare questa funzione, nel nostro caso l'MSE prende il nome di *funzione costo*. Per minimizzare la funzione costo possiamo studiarne la derivata. Questo procedimento, quando possibile, risulta comunque difficile e dispendioso in termini di tempo e di capacità di calcolo; spesso l'alternativa che viene utilizzata è il metodo della *discesa del gradiente* [11].

La discesa del gradiente è un metodo numerico iterativo che ci permette di calcolare punti di massimo e minimo in una funzione in più variabili. Si sceglie casualmente una soluzione iniziale per la funzione di cui vogliamo trovare il minimo e ci spostiamo in un intorno di x_0 , nel caso in cui per una nuova soluzione x_1 valesse $f(x_1) < f(x_0)$ allora ripetiamo il processo, usando come nuova soluzione di partenza x_1 ; il metodo giunge a conclusione quando, spostandoci di una determinata grandezza p detta *passo*, troviamo solo valori x_i tali che $f(x_i) \geq f(x_f)$, con x_f ultimo valore trovato. Nel caso unidimensionale possiamo visualizzare il tutto immaginandoci una sfera che rotola su delle colline (dove la sfera è il nostro punto di studio x_i e le colline sono il nostro grafico); posizionando la sfera su un'altura tenderà a rotolare verso valle (ovvero un punto di minimo locale) per poi stabilizzarsi lì.

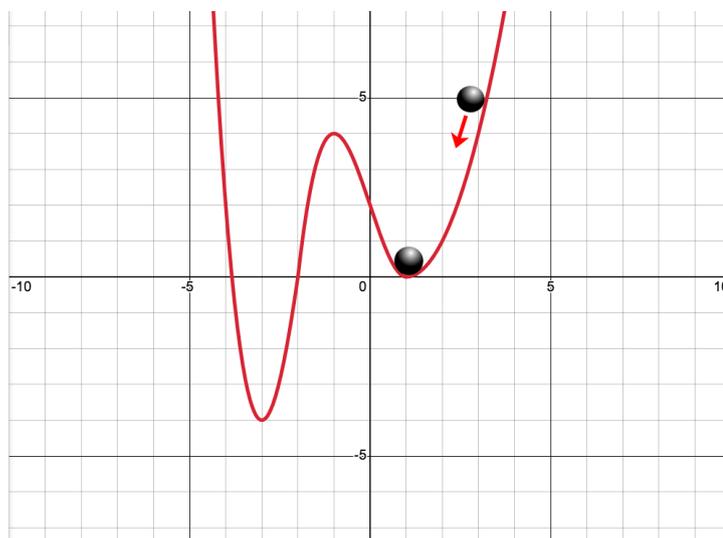


Figura 2.6: Visualizzazione unidimensionale discesa del gradiente

Nel caso bidimensionale possiamo immaginarci la stessa cosa ma su un piano, generalizzando a più dimensioni ed usando l'analisi, stiamo calcolando il gradiente della funzione nel punto x_i e ci stiamo "muovendo" nella direzione opposta. Il problema di questo metodo risiede nel fatto che $\nabla f(x_i)$ non sempre risulta semplice da calcolare: per questo motivo introduciamo l'algoritmo della *backpropagation*.

Con *backpropagation* si intende una famiglia di algoritmi, spesso usati in reti neurali con apprendimento supervisionato, che permettono alla rete di aggiustare pesi e bias per ottenere i risultati desiderati nella fase di training. Il processo è il seguente: quando la rete neurale compie un errore nella fase di training, viene calcolata la funzione costo con l'obiettivo di minimizzarla, intuitivamente la rete neurale cerca di far avvicinare il più possibile i valori ottenuti in output ai valori che avrebbe dovuto ottenere.

Per modificare il valore di uno dei neuroni in output, la rete neurale può modificare i bias ed i pesi dei neuroni del livello precedente, ma questa modifica influenza solo in parte il risultato finale; un'altra componente che possiamo modificare è la soglia di attivazione dei neuroni del livello precedente, per fare questo possiamo ripetere il processo sopraelencato: modificando pesi e bias dei neuroni del livello precedente e così via, fino ad arrivare al livello in input. Questo procedimento viene effettuato per ogni neurone in output e conseguentemente per ogni neurone della rete, dopo aver aggiustato i pesi

e bias per un determinato errore del network, questa riprende la fase di training fino al prossimo errore, dove ripeterà l'algoritmo.

2.2 Modello trasformatore per il deep learning

Un particolare tipo di rete neurale che ha recentemente preso d'assalto il mondo dopo una ricerca di Vaswani A. del 2017 dal titolo "Attention Is All You Need" prende il nome di *trasformatore* [12]. Questo particolare tipo di rete neurale è il modello che sta alla base di intelligenze artificiali capaci di trasformare una porzione di testo in voce artificiale (*Text To Speech*, abbreviato come TTS), una descrizione in un'immagine (*Text To Image*) o generare una risposta a partire da una porzione di testo (chatbot, come ad esempio il famoso ChatGPT).

Prendendo proprio ad esempio ChatGPT, possiamo osservare alcuni particolari: l'acronimo GPT sta per *Generative Pre-trained Transformer*, il primo termine fa riferimento alla capacità dell'assistente virtuale di generare porzioni di testo, il secondo spiega come per ottenere questo risultato finale sia stato necessario addestrare precedentemente l'AI e Transformer indica il tipo di rete neurale.

Il funzionamento di un trasformatore mantiene la struttura fondamentale delle reti neurali viste in precedenza, apportando però alcune importanti modifiche necessarie a causa della maggiore complessità del compito. Viene fornita alla rete neurale una lista di parole, le quali vengono rappresentate da una matrice, denominata *matrice di embedding*, in cui ogni colonna rappresenta una singola parola della lista; il word embedding è un insieme di tecniche che permette alla macchina di memorizzare informazioni semantiche e sintattiche di ogni parola. Quando un trasformatore riceve un input, lo divide in parti più piccole denominate *token*: alcuni esempi di token in un trasformatore come ChatGPT possono essere le sillabe delle parole ed i segni di punteggiatura.

Come ogni AI anche i trasformatori si basano sull'utilizzo di matrici: ad ogni token è associato un vettore colonna dove vengono salvati dei valori che rappresentano principalmente il rapporto con gli altri token presenti nel vocabolario della rete, questi vettori vengono poi processati attraverso un blocco denominato *attention block*, che permette a questi di aggiornare i propri valori influenzandosi a vicenda. Successivamente, i vettori

passano attraverso un altro blocco denominato *perceptrone multistrato*, simile alle reti neurali viste nel capitolo precedente, dove non comunicano tra loro ed eseguono tutti lo stesso processo in parallelo.

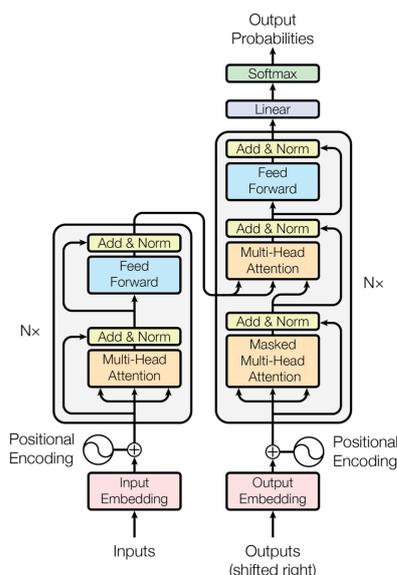


Figura 2.7: Schema di un trasformatore

Ad ogni token nella frase in input è associata una colonna della matrice di embedding, quindi l'ultima colonna rappresenta l'ultimo token della frase; una volta finito il ciclo di attention block e perceptrone multistrato la colonna finale conterrà dei valori che rappresentano la correlazione con le altre parole del vocabolario. Per ottenere la probabilità che un determinato token sia il successivo rispetto alla parola finale della frase segue un procedimento di *unembedding*. Questo procedimento è caratterizzato da una prima fase nella quale la colonna dell'ultimo token viene moltiplicata per la matrice di embedding ed una seconda fase che, tramite una funzione denominata *softmax*, normalizza i valori del prodotto matriciale così da ottenere una distribuzione di probabilità. La funzione softmax riceve in input un vettore z e restituisce in output un valore compreso tra 0 ed 1 dopo aver applicato ad ogni elemento del vettore la funzione esponenziale ed averlo diviso per $\sum_{i=1}^n e^{z_i}$ per normalizzarne il valore.

Nel campo del machine learning, viene spesso aggiunto un termine T come divisore delle

funzioni esponenziali detto *temperatura*, più il valore di T è grande, più la funzione softmax darà importanza a valori minori rendendo la distribuzione più uniforme, mentre un valore basso di T restituirà una distribuzione molto sbilanciata verso i valori più grandi. Entriamo ora un po' più nel dettaglio sul funzionamento di questi due blocchi, per semplicità possiamo immaginarci che ogni token corrisponda ad una parola della nostra frase. Nell'attention block, il primo passo che compie la macchina è quello di scrivere i vettori relativi ad ogni singolo token, utilizzando come dato solo la singola parola e la sua posizione nella frase.

Il nostro obiettivo è avere vettori che contengono informazioni non solo della singola parola ma anche delle precedenti, per questo motivo definiamo il vettore di *query*: un vettore contenente, sotto forma di numeri, le informazioni rilevanti che la parola può ricevere per comprendere il contesto entro cui è inserita, come ad esempio se nella frase sono presenti aggettivi riferiti alla parola, preposizioni o altro. Il vettore query è il risultato del prodotto righe per colonne della *matrice di query*, che possiamo immaginare come una matrice contenente tutte le informazioni possibili che può ricevere una parola, per il vettore della singola parola in esame.

Un altro vettore associato alla parola è il vettore delle *keys*, contenente tutte le informazioni che la parola può dare per chiarire il contesto a parole successive; anche questo vettore è il risultato di un prodotto fra la *matrice delle keys*, contenente tutte le informazioni che una generica parola può fornire, ed il vettore della parola in esame. Possiamo immaginare il vettore delle query come una serie di domande che la parola pone alle parole precedenti per capire il contesto in cui si trova ed il vettore delle keys come le risposte che la parola può dare alle risposte che le vengono poste dalle parole successive, per capire quanto una risposta sia appropriata per una domanda eseguiamo un prodotto scalare tra il vettore delle keys ed il vettore delle query. Il risultato del prodotto scalare sarà un numero reale, per renderlo una distribuzione di probabilità applichiamo l'algoritmo softmax; la funzione attenzione utilizzata nel paper di Vaswani A. è $Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V$ dove d_k è la dimensione delle matrici di query e keys e il termine a denominatore $\sqrt{d_k}$ serve a rendere il metodo numericamente stabile. Nella fase di training, anziché fornire alla rete delle frasi chiedendole di dedurre la parola successiva, risulta più efficiente utilizzare quelle già scritte come base per l'addestramen-

to oscurando tutte le parole eccetto la prima, chiedendo di scrivere la successiva per poi rivelare la risposta, permettendo così alla rete di aggiustare i propri pesi. Da un punto di vista tecnico, questo metodo di addestramento implica che le parole non possono influenzare parole precedenti e prende il nome di *masking*. Ma come fanno i vettori dei token a comunicare fra di loro ed aggiornare il proprio significato in base al contesto?

Definiamo una matrice denominata *matrice dei valori* che, moltiplicata per il vettore dei token restituisce come risultato il *vettore dei valori*. Possiamo immaginarci la matrice dei valori come una matrice capace di quantificare la rilevanza di una parola nella frase, per aggiustare il significato dei token successivi e fornirgli contesto.

Ogni vettore dei valori viene successivamente moltiplicato per ogni token, sommando le componenti del risultato della moltiplicazione si ottiene un vettore che rappresenta lo spostamento causato dal contesto e sommandolo ai singoli token otteniamo un valore aggiornato contenente informazioni derivanti dalla frase.

Questo concetto che abbiamo definito è chiamato di *auto-attenzione* e l'insieme delle matrici dei valori, embedding, query e keys prende il nome di *head of attention*; spesso nelle AI si utilizzano sistemi *multi-headed* che lavorano in parallelo fra loro, ognuno con le proprie matrici. Questo significa che ogni head ha una sua matrice dei valori distinta dalle altre che, moltiplicata per ogni token dell'input e sommato le componenti del risultato, fornisce lo spostamento causato dal contesto; per il singolo token, si somma ogni spostamento dato da ogni head per ottenere la matrice del token aggiornata. Questi due tipi di blocchi (attention block e perceptrone multistrato) vengono ripetuti fino ad ottenere un vettore di output, contenente numeri rappresentanti la probabilità di ogni parola nel vocabolario. La capacità di dedurre l'elemento successivo in una frase implica l'abilità di generare testi, dando come nuovo input la proposizione appena creata, in questo modo i chatbot riescono a scrivere risposte coerenti con l'input ricevuto.

2.3 AI e didattica

Come accennato nel capitolo precedente, a novembre 2022 è stato pubblicato online ChatGPT, un chatbot capace di imitare conversazioni umane al fine di automatizzare il supporto online, addestrato con una massiccia quantità di dati testuali e per questo

definito Large Language Model (LLM) [13]. All'inizio del 2023 sono uscite molte ricerche sul tema dell'AI e di ChatGPT che indagano sulle abilità del nuovo chatbot di OpenAI di svolgere abilità fino a poco tempo fa pensate impossibili per una macchina, come ad esempio superare esami di data analysis [14]. Altri articoli hanno proposto dei questionari a studenti universitari chiedendo loro se usano ChatGPT nella vita quotidiana o per compiti universitari e se questo utilizzo possa considerarsi al pari di copiare, sottolineando come il fatto che questi chatbot siano di pubblico utilizzo richieda una nuova discussione a livello educativo [15]. Altri articoli, come quello di Zhu et al. [16] evidenziano i pro nell'uso di ChatGPT nell'istruzione, come la possibilità di "generare risposte plausibili a domande in input", "fornire soluzioni e sostegno per compiti complessi" e "valutare incarichi e fornire feedback". La stessa ricerca evidenzia anche i contro, come "l'inclusione di informazioni imprecise, inventate o parziali nelle risposte" e "mancanza di pensiero critico e di comprensione delle risposte".

Nel complesso, ciò che emerge dalle ricerche è come ChatGPT sia uno strumento ed in quanto tale non possa essere definito di per sé positivo o negativo, ma come questa differenza risieda nell'utilizzo che se ne fa. In un'epoca in cui l'informazione e la tecnologia rivestono il ruolo di protagonisti, molti studiosi di didattica hanno iniziato a riformulare l'insegnamento sfruttando le potenzialità della *media education* come riportato nel lavoro di Rivoltella P. et al. [17] in cui si sottolinea la necessità da parte dei docenti di integrare elementi digitali all'interno delle proprie lezioni, rendendo partecipi gli studenti delle potenzialità e dei rischi che queste nuove forme di apprendimento portano con loro. Ad inizio del 2023 sono uscite molte ricerche il cui scopo è quello di indagare le opinioni delle persone nei confronti delle AI e di ChatGPT in vari ambiti, dall'istruzione [15], alla sanità [18], al marketing [19].

Le principali preoccupazioni relative ai chatbot riguardano la possibilità di utilizzarli per falsificare i compiti, per questo motivo sono usciti molti programmi capaci di identificare testi generati con AI. Le principali opportunità che possono dare i chatbot riguardano una didattica più digitale, basata su un insegnamento più personale, incentrato sugli studenti. È notizia delle prime settimane di Luglio che il ministro dell'Istruzione e del Merito, Giuseppe Valditara, abbia firmato una circolare che impone il divieto di utilizzo dei cellulari fino alle scuole secondarie di primo grado, anche per scopi didattici, lascian-

do però la possibilità di utilizzare pc e tablet sotto la supervisione dei docenti.

Nel numero di Settembre 2024 della rivista Prisma compaiono tre articoli sul tema dell'AI e della didattica, due a cura di Angelo Guerraggio ed uno di Paola Ricciardi. Il primo articolo di Guerraggio A. è un'intervista ad Alfio Quarteroni: matematico italiano al 48esimo posto nel mondo nella lista *Top Mathematichs Scientists* del 2022, esperto di matematica applicata in campi che spaziano dallo sport alla medicina [20]. In particolare, l'intervista condotta da Guerraggio tratta il tema delle AI e dei finanziamenti italiani alla ricerca in questo ambito: Quarteroni sostiene che l'Italia "abbia perso il treno delle AI" a causa della scarsità dei finanziamenti e delle commissioni specializzate e che occorra "sottolineare l'aspetto euristico, quasi sperimentale, della matematica: riscoprire il fascino del problema a cui dare soluzione e la conseguente ricerca di nuove strade".

Il secondo articolo di Guerraggio affronta il tema molto dibattuto dell'utilizzo del telefono cellulare in classe, chiedendo a docenti che insegnano dalle scuole secondarie di primo grado fino al livello universitario il loro parere [21]. La quasi totalità dei docenti intervistati, nonostante sottolinei i rischi di un uso sbagliato di questi dispositivi, si sente contrario al divieto totale imposto dalla circolare firmata dal ministro Valditara, evidenziando come sarebbe stato meglio lasciare la possibilità decisionale ai docenti.

L'ultimo articolo è di Paola Ricciardi che intervista la psicologa e neuro-psicologa Tiziana Metitieri dell'ospedale pediatrico "Meyer" di Firenze [22]. Metitieri sottolinea come tuttora non ci siano evidenze scientifiche sull'impatto negativo dell'uso dei telefoni a livello di neurosviluppo e connessioni neurali, in particolare non è stata trovata una relazione causale tra uso di dispositivi e sviluppo cerebrale e benessere cognitivo e mentale. Rispetto ad un approccio autoritario dell'educazione che si basa sul vietare, Metitieri ritiene più efficace mostrare modelli positivi di didattica anche con l'ausilio di strumenti digitali, in linea con il "Decalogo sull'uso dei dispositivi mobili a scuola" redatto nel 2018 dall'allora ministra Valeria Fedeli [26]. All'interno di queste linee guida per il Piano Nazionale Scuola Digitale troviamo scritto che "I cambiamenti non vanno rifiutati, ma compresi e utilizzati per il raggiungimento dei propri scopi" e "Il digitale nella didattica è una scelta: sta ai docenti introdurla e condurla in classe", in apparente contrasto con la circolare firmata ad inizio Luglio 2024. Il timore di Metitieri, condiviso da molti insegnanti, è che questo divieto contribuirà ad aumentare il divario tra le attività scolastiche

e la vita di tutti i giorni, aumentando la retorica per cui la scuola è "sempre un po' più indietro" rispetto alla vita vera.

Capitolo 3

La mia esperienza al Play

3.1 Play Festival del Gioco 2024

Il 26 marzo 2024 la professoressa Silvia Benvenuti mi ha chiesto di partecipare con la professoressa Rita Fioresi al Play - Festival del Gioco di Modena in qualità di divulgatore per il progetto CaLISTA, proposta che ho accolto con entusiasmo. Il Play - Festival del Gioco è una convention italiana nata nel 2017 che si svolge nel corso di più giornate nel quartiere fieristico di Modena su un'area di circa 22000 mq; dall'edizione 2024 si è vista una crescita di EduPlay, la sezione della convention del gioco dedicata ad aspetti culturali ed educativi del gioco, con un'espansione dell'Area Scientifica a cui ho avuto il piacere di partecipare [27].

La mia presenza è stata possibile grazie a CaLISTA (Cartan geometry, Lie, Integrable Systems, quantum group Theories for Applications): una rete di ricerca all'interno del progetto europeo COST (European Cooperation in Science and Technology). COST è un'organizzazione di finanziamento capace di mettere in contatto fra loro ricercatori di tutta Europa e fornire fondi per la ricerca e lo sviluppo della matematica e la sua divulgazione. CaLISTA oltre a portare avanti ricerche nel campo delle simmetrie nei gruppi di Lie e nelle algebre di Lie, ha come obiettivo la *dissemination* e la divulgazione della matematica, in particolar modo della geometria. Interfacciandomi con la professoressa Rita Fioresi, action chair del progetto CaLISTA, ho preso la decisione di preparare delle attività relative al gioco del Go e delle AI. Con l'aiuto di Maurizio Parton, professore

associato all'Università degli Studi G. d'Annunzio Chieti e Pescara, ho preparato un'introduzione storica alle AI programmate per giocare a Go da poter esporre al Play; dopo la creazione di alcune slide illustrative per introdurre l'argomento, ho chiesto in prestito delle pietre e dei goban dal Bologna Go Club [28] per far provare il gioco del Go al pubblico del Play.

All'evento insieme a me erano presenti molti ricercatori, divulgatori ed affiliati al progetto CaLISTA che hanno presentato escape room sulla storia della matematica, in particolare sul teorema di Pitagora ed i sistemi di numerazione babilonese ed egiziano, laboratori didattici sulla teoria dei nodi e sulle simmetrie spaziali.

Per far provare il gioco del Go ai partecipanti del Play ho deciso di utilizzare un goban 9×9 , indicato per giocatori alle prime armi viste le sue dimensioni ridotte che consentono partite più veloci ed accessibili a persone di ogni età. Dopo un'iniziale timidezza dei presenti nei confronti del goban, probabilmente a causa della sua similarità con il gioco degli scacchi, molti curiosi si sedevano al tavolo per farsi spiegare le regole e, visto il loro numero ridotto e la semplicità del gioco, rimanevano per una partita. La spiegazione delle regole veniva adattata a seconda dell'età della persona che avevo davanti, scegliendo un approccio più incentrato su similitudini per i giovani giocatori e più tecnico per i giocatori più adulti; il metodo risultato più fruttuoso è stato quello di tracciare un parallelo tra il goban ed un campo di battaglia, riprendendo le leggende sull'origine del Go, secondo cui il gioco è un'estensione delle abitudini generalesche di utilizzare delle pietre per pianificare gli attacchi del proprio plotone sulle cartine geografiche.

Il goban rappresenta il campo di battaglia, i giocatori sono i generali e le pietre i soldati del nostro esercito; i soldati vengono paracadutati nella posizione indicata dalla pietra e non possono essere spostati nel corso della partita. Ogni linea presente sul goban può essere vista come una strada, questo implica che i soldati possono essere posizionati solo negli incroci stradali; le strade che escono dall'incrocio vengono denominate libertà e ogni volta che ad un soldato vengono tagliate tutte le vie di ritirata dalle truppe nemiche (nel gioco del Go diremo che *non hanno più libertà*) queste lo catturano come prigioniero. Soldati dello stesso esercito posizionati su incroci con una strada in comune (che hanno delle libertà in comune) si fanno forza a vicenda, creando un plotone (un *gruppo*).

Le libertà del gruppo sono la somma delle libertà delle singole pietre, è quindi possibile

catturare un intero gruppo di pietre se si riesce a circondarlo completamente. Non si possono posizionare pietre in incroci già circondati dal nemico, sarebbe come mandare un soldato incontro a morte certa, questa viene chiamata *regola del suicidio*. La precedente regola non si applica nel caso in cui la presenza della nuova pietra nell'incrocio circondi un gruppo nemico, in questo caso la mossa è da ritenersi valida. La battaglia finisce quando non si possono più schierare unità o quando entrambi i generali si ritirano dal comando (passano il turno), quindi si calcola l'estensione del territorio conquistato per determinare il vincitore.

La "regola del *ko*" (giapponese, traducibile come "eone") veniva introdotta all'occorrenza, così da non appesantire eccessivamente la spiegazione iniziale, favorendo un approccio più diretto al gioco. La regola del *ko* serve per evitare ripetizioni infinite della stessa mossa, il caso tipico in cui questa regola viene introdotta è visibile nell'immagine qui sotto: Nero ha appena giocato la pietra col il numero 1 sopra, catturando una pietra bianca che stava nell'incrocio indicato dal cerchio rosso.

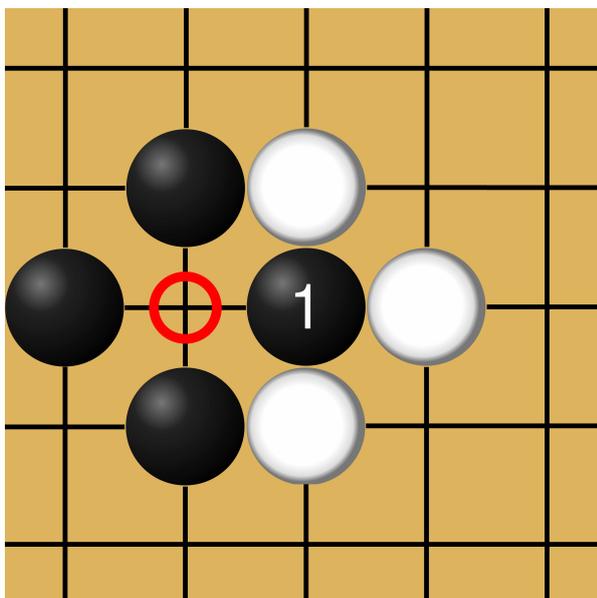


Figura 3.1: Classico esempio della regola del *ko*

Senza la regola del *ko*, Bianco potrebbe nuovamente posizionare una pietra nell'intersezione con il cerchio rosso catturando la pietra nera 1 appena posta, ricreando la configurazione precedente del tabellone, creando così la possibilità di una ripetizione

infinita di queste due mosse. Secondo la regola del *ko*, Bianco non può giocare nell'intersezione con il cerchio rosso in questo turno ed è costretto a giocare altrove, nel turno successivo Nero potrà chiudere il *ko* posizionando la propria pietra nel cerchio rosso oppure lasciare che Bianco catturi la pietra nera 1 nel turno successivo.

Durante le partite svolte fra giocatori di tutte le età, da bambini della scuola primaria a docenti di matematica in pensione, ho cercato di capire, anche chiedendo esplicitamente, quali fossero i ragionamenti dietro ogni mossa: spesso i giocatori evidenziavano dei ragionamenti basati su una o due mosse successive; altre volte, forse presi dal gioco, davano più importanza al salvataggio di una singola pietra rispetto ad un intero gruppo, decisione che spesso metteva in posizione di svantaggio i giocatori.

L'idea di lasciar giocare le persone senza consigli sulle mosse ha permesso loro di scoprire in autonomia alcune regole derivate da quelle iniziali, come ad esempio comprendere quando un gruppo è da considerarsi *vivo* o *morto*. Un gruppo di pietre isolato che non ha possibilità di collegarsi ad un altro gruppo vicino può essere *vivo* o *morto*: vivo quando non può essere catturato in nessun caso, nemmeno se il turno successivo è del giocatore avversario; viceversa è detto morto nel caso in cui il giocatore che possiede il gruppo non possa in alcun modo difenderlo, nemmeno se è il suo turno.

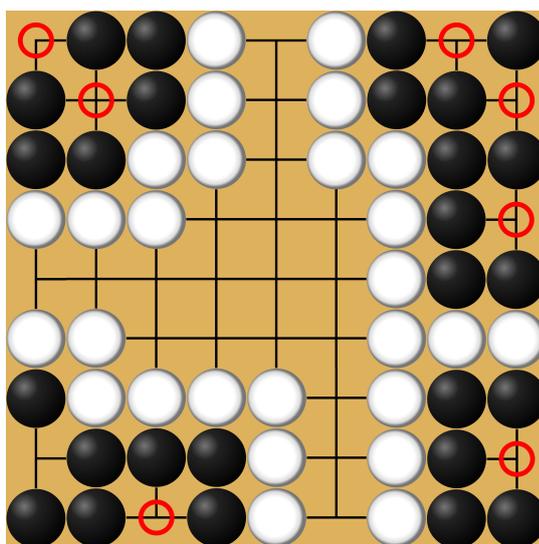


Figura 3.2: Esempio di gruppi vivi e gruppi morti

Nell'immagine a pagina precedente abbiamo evidenziato gli occhi con dei cerchi rossi, i gruppi di pietre nere nella parte alta della scacchiera sono vivi dato che hanno almeno due occhi, il gruppo in basso a destra è morto dato che ha un solo occhio. Il gruppo in basso a sinistra sembra avere due occhi, ma l'intersezione vuota senza cerchio può essere catturata da Bianco, rendendolo di fatto un falso occhio. Questa nuova regola, che spesso veniva compresa verso la fine, incoraggiava le persone a giocare una nuova partita, curiosi di scoprire se il risultato potesse avere esito differente dal precedente. La più grande differenza che ho notato fra giocatori giovani e giocatori anziani è stata sicuramente il tempo che trascorrevano tra una mossa ed un'altra: in media i giocatori giovani si riservavano meno tempo per pensare alla propria azione, forse perché miopi rispetto a successive tattiche dell'avversario o semplicemente perché più desiderosi di giocare e divertirsi.

Un episodio che probabilmente mi porterò dietro da questa esperienza riguarda un bambino che frequentava il terzo anno di una scuola primaria a Modena, che per semplicità chiameremo Giuseppe. Giuseppe si è avvicinato con curiosità al goban con sopra ancora le pietre di una precedente partita ed ha iniziato a spostarle da un incrocio all'altro, incuriosito dal caratteristico suono che queste producono quando vengono posizionate sul legno della scacchiera. Dopo aver spiegato le regole ed aver proposto di giocare una partita, Giuseppe ha accettato la sfida ed ha posizionato la sua prima pietra; la prima fase del gioco è passata studiando l'avversario e ricordando alcune regole, con alcuni tentativi di cattura da entrambi i lati della battaglia. Le mosse di Giuseppe erano calcolate: spesso si fermava a riflettere su quale sarebbe potuta essere la mia mossa in risposta ad una sua pietra giocata, lungimiranza che gli permetteva di evitare diverse perdite.

A metà partita la madre di Giuseppe ha iniziato a guardare con interesse le mosse del figlio, consigliando alle volte le strategie ritenute migliori. Finito il primo gioco, terminato con la mia premeditata sconfitta, Giuseppe era entusiasta di provare alcune nuove tattiche; in questo momento un suo compagno di classe si è avvicinato al tavolo con il goban. Giuseppe, esaltato dal nuovo gioco, si è subito proposto di spiegare le regole all'amico e, dopo alcune precisazioni da parte mia, ha lasciato che fosse il suo sfidante a posizionare la prima pietra. Proseguendo nella partita però, la madre di Giuseppe correggeva sempre più spesso le mosse del figlio, a tal punto che il bambino chiedeva il

parere della madre prima di posizionare la propria pedina.

Come sostiene l'astrofisico Neil deGrasse Tyson in diverse interviste ed interventi che ha tenuto: "Kids are born scientists" (i bambini nascono scienziati); spesso viene citata un'intervista in cui Tyson racconta di quando, camminando per Central Park dopo un forte temporale, gli capitò di osservare una madre passeggiare con il proprio figlio, vestito con impermeabile e galosce. A causa della pioggia il terreno era fangoso e pieno di pozzanghere e dai movimenti del bambino si poteva intendere come questo provasse il desiderio di saltare nell'acqua; la madre avendo intuito le intenzioni del figlio cambiò strada evitando che il fango potesse sporcare i vestiti di entrambi [23].

Agli occhi di Tyson l'avvenimento equivale ad un esperimento incompiuto, data l'innata curiosità dei bambini il saltare in una pozzanghera può essere fonte di scoperta, ad esempio: azione-reazione di una forza e fluidodinamica. Non posso fare a meno di rivedere Giuseppe nel bambino con l'impermeabile di Central Park, desideroso di sperimentare diverse tattiche in questo nuovo gioco ma timoroso di sbagliare, forse a causa dalla presenza indiscreta della madre.

3.2 Approccio ludico al problem solving

Far giocare i partecipanti del Play di Modena a Go aveva come obiettivo quello di far scontrare persone di tutte le età con il tema del *problem solving*, tema molto caro ai matematici per ovvi motivi. Il *problem solving*, come ci suggerisce la parola stessa, è l'abilità di un individuo di risolvere problemi usando tecniche e modi generici, per parlarne è necessario quindi specificare cosa intendiamo con il termine *problema*. La definizione che riportiamo è quella della Gestalt: una corrente psicologica nata agli inizi del XX secolo in Germania, incentrata sull'idea che la percezione si sviluppi come una totalità strutturata di elementi che si influenzano a vicenda e non di elementi isolati.

Gaetano Kanizsa, psicologo della Gestalt, scriverà in merito al *problem solving*: "*un problema sorge quando un essere vivente, motivato a raggiungere una meta, non può farlo in forma automatica o meccanica, cioè mediante un'attività istintiva o attraverso un comportamento appreso*" [24]. In questa definizione gioca un ruolo chiave anche la *motivazione*: motore che permette al soggetto di proseguire nella risoluzione, una

necessità dettata da un bisogno fisiologico senza la quale non esisterebbe il problema. Le prime idee della Gestalt, per quanto innovative rispetto al contesto storico in cui sono emerse, fanno ancora grande affidamento sulle *euristiche*: un insieme di metodi che i risolutori di problemi applicano di fronte ad un ostacolo. In particolare, nel 1945 viene pubblicato *How to solve it*, un libro dello psicologo George Polya in cui si divide un problema in quattro fasi: *comprensione*, *compilazione di un piano*, *sviluppo del piano* e *verifica*. La fase centrale per la risoluzione di un problema è la compilazione del piano di risoluzione, in cui si ricorre a delle euristiche per raggiungere la soluzione, queste euristiche possono essere viste come domande la cui risposta può far intravedere la soluzione. Gli assunti principali dei ragionamenti di Polya per facilitare il problem solving sono:

- Per essere un buon risolutore di problemi è sufficiente avere un buon bagaglio di conoscenze ed euristiche
- L'efficacia dell'insegnamento per euristiche

Come osservarono molti psicologi successivi a Polya, quali ad esempio Schoenfeld e Brown, i due assunti fondamentali di Polya non sono sufficienti a risolvere un generico problema, da qui si iniziò a tener conto di altre variabili nel problem solving, come ad esempio il ruolo delle emozioni e delle convinzioni relative ai problemi.

Il problem solving riveste un ruolo essenziale nella nostra società moderna: Paul Halmos identifica il risolvere problemi come la vera essenza della matematica, il Parlamento Europeo individua nella competenza matematica una delle otto competenze chiave per l'apprendimento permanente di un cittadino europeo, definendo la competenza matematica come la *capacità di sviluppare e applicare il pensiero e la comprensione matematici per risolvere una serie di problemi in situazioni quotidiane* [25].

All'interno delle Indicazioni Nazionali il problem solving è citato in ogni ciclo d'istruzione, ad esempio già nel documento per il curriculum della scuola dell'infanzia e del primo ciclo d'istruzione possiamo leggere: *Caratteristica della pratica matematica è la risoluzione di problemi, che devono essere intesi come questioni autentiche e significative, legate alla vita quotidiana, e non solo esercizi a carattere ripetitivo o quesiti ai quali si risponde semplicemente ricordando una definizione o una regola* [29].

Possiamo chiederci per quale motivo sia importante promuovere il problem solving a scuola, la risposta possiamo trovarla in un articolo Brigaglia e Perez [30]. Nel testo della ricerca si sottolinea come la capacità di risolvere problemi sia inclusa nelle cinque finalità educative all'interno del Piano Nazionale di Informatica per il triennio, oltre che in diversi punti nei dodici obiettivi di apprendimento.

Un modo per recuperare la motivazione allo studio della matematica è quello di rendere lo studente protagonista di un processo mentale: lo scopritore della soluzione, anziché soggetto passivo nell'applicazione di un algoritmo di risoluzione, questo processo è facilitato quando ci troviamo all'interno di una *situazione a-didattica*: situazione in cui lo studente diventa responsabile del processo e del metodo di risoluzione. Un'attività di carattere a-didattico che possiamo proporre agli studenti è quella del *laboratorio didattico*. Quando sentiamo parlare di laboratorio a scuola probabilmente tornano alla mente lezioni di fisica o di chimica dove venivano svolti degli esperimenti sotto la guida dell'insegnante, per poi scrivere una relazione sulla propria esperienza.

Un laboratorio di matematica non necessita di un'aula specifica e spesso richiede meno materiale rispetto ad un laboratorio di fisica o chimica; l'obiettivo del laboratorio è quello di rendere gli studenti protagonisti del processo logico-deduttivo, portandoli alla soluzione di problemi di carattere matematico e non solo, utilizzando conoscenze, matematiche e non, apprese nel percorso scolastico. Nonostante non esista una definizione formale di "laboratorio di matematica" è convenzione parlare di laboratorio quando gli studenti hanno un ruolo attivo, confrontandosi con i propri compagni di classe, lavorando in gruppi, al fine di raggiungere un obiettivo comune, proponendo in libertà le proprie idee e le proprie competenze. Per gli insegnanti il laboratorio è l'ambiente ideale per stimolare la creatività degli studenti, in cui rivestono il ruolo di guide esperte, ponendo domande finalizzate ad un ragionamento anche attraverso materiali manipolabili o software didattici.

Un'altra regola non scritta per parlare di laboratorio, è quella di ridefinire il ruolo dell'errore. "*Gli errori sono necessari, utili come il pane e spesso anche belli: per esempio, la torre di Pisa*" è scritto nella prima pagina de Il libro degli errori, raccolta di filastrocche e racconti brevi di Gianni Rodari (1964) [31]. La penna di Gianni Rodari mette in evidenza un aspetto caro ai matematici: la capacità di vedere l'errore in modo propo-

sitivo, come fonte di ispirazione e miglioramento. In un laboratorio didattico, l'errore assume un ruolo diverso da quello che assume durante una verifica: l'errore infatti non viene represso o cancellato, bensì discusso con il gruppo classe per sfruttare al massimo le intuizioni che si celano dietro ad esso.

Spesso gli studenti si sentono soffocati dall'idea di sbagliare, per paura del giudizio dei compagni e soprattutto dell'insegnante, per un buon laboratorio è necessario creare un ambiente in cui sbagliare sia permesso. Tornando all'articolo di Brigaglia e Perez, possiamo osservare come un metodo per proporre problemi matematici in contesti a-didattici sia l'elemento ludico: attribuire cioè un carattere ludico al problema da risolvere celando, solo momentaneamente, la matematica che si nasconde all'interno.

Nonostante l'idea del gioco matematico possa sembrare moderna, moltissimi problemi matematici nel corso della storia sono nati come sfide fra studiosi, come ad esempio il problema della divisione della posta o le *disfide matematiche* che si tenevano nel XVI secolo sotto i portici della Chiesa di Santa Maria dei Servi a Bologna.

Come evidenzia Peter Vankúš nel suo articolo, i giochi matematici sono stati utilizzati come strumento didattico fin dall'antica Grecia. La ricerca inoltre evidenzia come l'approccio ludico alla matematica "aumenti le conoscenze e le attitudini matematiche degli alunni" [32]. Questa ed altre evidenze scientifiche sul tema della didattica, in particolare della matematica, sembrano indirizzare i futuri docenti ad un approccio meno frontale alla didattica, favorendo sempre più situazioni a-didattiche incentrate sul lavoro di gruppo degli studenti e sul reciproco supporto.

Conclusioni

Lo scopo di questo elaborato era quello di analizzare, senza pretesa di completezza, il mondo delle intelligenze artificiali e realizzare in seguito un laboratorio didattico attinente al tema. Nella prima parte ci siamo concentrati sull'aspetto storico delle intelligenze artificiali, soffermandoci sui supercomputer che nei decenni sono stati capaci di superare e battere i campioni del mondo di scacchi e di Go.

Abbiamo poi spiegato il funzionamento di una rete neurale partendo dalla sua componente fondamentale, il perceptrone, per poi aumentare la complessità della rete parlando del modello del trasformatore che sta alla base dei chatbot più moderni, come ad esempio ChatGPT.

L'utilizzo delle intelligenze artificiali a livello scolastico sta diventando con gli anni argomento sempre più pressante, che richiede una discussione informata su più fronti: dal lato degli insegnanti e degli studenti, abbiamo quindi affrontato la discussione basandoci su paper che prendono in considerazione l'opinione degli studenti e degli insegnanti ed articoli che evidenziano i pro ed i contro di una didattica più direzionata verso le AI, parlando anche delle decisioni legislative sul tema che sono state prese per l'inizio di questo anno scolastico 2024/2025.

Nell'ultimo capitolo ho raccontato la mia esperienza al Play di Modena dove ho portato il laboratorio didattico pensato per l'occasione, analizzando le strategie messe in atto per spiegare le regole del gioco del Go e per stimolare il ragionamento degli avversari.

Abbiamo spiegato il significato di termini quali laboratorio didattico, situazione a-didattica e problem solving, portando alcune ricerche per avvalorare l'ipotesi che un approccio ludico alla matematica possa favorire l'apprendimento e l'interesse degli studenti nei confronti della materia.

Portare il gioco del Go al Play mi ha permesso di fare divulgazione matematica e mi ha dato la possibilità di interfacciarmi con un pubblico che difficilmente è presente ai convegni fra matematici: tutta quella fetta del pubblico denominata di "non esperti".

Lo scopo del mio laboratorio didattico era, oltre a dare un'infarinatura sul funzionamento delle reti neurali e sulla storia dell'intelligenza artificiale, di rafforzare il rapporto che alcuni studenti possono avere con la matematica, o addirittura di far riavvicinare alla matematica adulti che le hanno voltato le spalle.

La disseminazione della matematica serve per far uscire la matematica dalle università ed i libri di testo, perché la matematica fa parte della vita di tutti i giorni e non è solo una materia relegata dietro i banchi di scuola.

Ritengo che il laboratorio didattico sul Go possa essere riutilizzato in un contesto scolastico, sia come approccio ludico alla matematica, data la sua similitudine con il gioco degli scacchi, sia per introdurre alcuni argomenti strettamente matematici, come ad esempio gli assi cartesiani: essendo il goban una griglia, possiamo utilizzare le intersezioni delle sue linee come le coordinate di un punto (a patto di fissare un punto di origine) per rendere tangibile e manipolabile il concetto delle coordinate di un punto agli studenti.

Bibliografia

- [1] Ciancarini P. (1992), I giocatori artificiali, Mursia, Italia
- [2] Seiradakis, J.H., Edmunds, M.G. Our current knowledge of the Antikythera Mechanism. *Nat Astron* 2, 35-42 (2018)
- [3] Poe E.A. (2012), Il giocatore di scacchi di Maelzel, Mursia, Italia
- [4] Murray Campbell, A. Joseph Hoane, Feng-hsiung Hsu, Deep Blue, *Artificial Intelligence*, Volume 134, Issues 1-2, 2002, Pages 57-83
- [5] Sylvain Gelly, David Silver, Monte-Carlo tree search and rapid action value estimation in computer Go, *Artificial Intelligence*, Volume 175, Issue 11, 2011, Pages 1856-1875
- [6] <https://arxiv.org/abs/2211.00241>
- [7] Wang T. (2023), Adversarial Policies Beat Superhuman Go AIs, 40th edition of the International Conference on Machine Learning, Luglio 25, reperibile a <https://slideslive.com/39006680/adversarial-policies-beat-superhuman-go-ais>
- [8] McCarthy J. (1955) A proposal for the Dartmouth summer research project on artificial intelligence
- [9] Rosenblatt F. (1957) The perceptron: A perceiving and recognizing automaton (Project PARA)
- [10] McCulloch, W.S., Pitts, W. A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics* 5, 115-133 (1943)

- [11] <https://arxiv.org/abs/1609.04747>
- [12] C. Subakan, M. Ravanelli, S. Cornell, M. Bronzi and J. Zhong, "Attention Is All You Need In Speech Separation," ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Toronto, ON, Canada, 2021, pp. 21-25
- [13] <https://arxiv.org/abs/2201.06657>
- [14] Eken, Süleyman, Ethic wars: student and educator attitudes in the context of ChatGPT (February 21, 2023)
- [15] Faycal Farhi, Riadh Jeljeli, Ibtehal Aburezeq, Fawzi Fayez Dweikat, Samer Ali Alshami, Radouane Slamene, Analyzing the students' views, concerns, and perceived ethics about chat GPT usage, *Computers and Education: Artificial Intelligence*, Volume 5, 2023, 100180
- [16] Zhu Chenjia, Sun Meng, Luo Jiutong, Li Tianyi, Wang Minhong, How to Harness the Potential of ChatGPT in Education?, *Knowledge Management & E-Learning*, v15 n2 pp. 133-152 Jun 2023
- [17] Rivoltella P.C., Rossi P.G. (2019), *Tecnologie per l'educazione*, Pearson
- [18] Wong R, Ming L, Raja Ali R The Intersection of ChatGPT, Clinical Medicine, and Medical Education *JMIR Med Educ* 2023;9:e47274
- [19] Jain, Varsha and Rai, Himanshu and ., Parvathy and Mogaji, Emmanuel, The Prospects and Challenges of ChatGPT on Marketing Research and Practices (March 23, 2023)
- [20] Guerraggio A. (2024), Il matematico dei due mondi, *Prisma* n.66 pp.36-39 Settembre 2024
- [21] Guerraggio A. (2024), Tutta colpa del cellulare. O no? *Prisma* n.66 pp.40-43 Settembre 2024

-
- [22] Ricciardi P. (2024), "Non Ã" vietando che tuteliamo i ragazzi", Prisma n.66 pp.44-45 Settembre 2024
- [23] Ministero dell'Istruzione dell'Università e della Ricerca (2018), Dieci punti per l'uso dei dispositivi mobili a scuola
- [24] <https://www.play-modena.it>
- [25] <https://bolognagoclub.altervista.org/>
- [26] Intervista a Neil deGrasse Tyson, astrofisico, del 19 novembre 2019, reperibile a <https://www.youtube.com/watch?v=Tv0kQbOIrjY>
- [27] G. Kanizsa, "Il "problem solving" nella psicologia della Gestalt", in G. Mosconi, V. D'Urso, (a cura di), "La soluzione dei problemi", Giunti-Barbera, Firenze, 1973, p.35
- [28] Gazzetta ufficiale dell'Unione europea, Raccomandazione del consiglio del 22 maggio 2018 relativa alle competenze chiave per l'apprendimento permanente
- [29] Ministero dell'Istruzione dell'Università e della Ricerca (2012), Indicazioni nazionali per il curriculum della scuola dell'infanzia e del primo ciclo d'istruzione
- [30] Brigaglia A., Perez G. E. (2001), Il problem solving e il gioco nell'insegnamento della matematica: le costruzioni geometriche
- [31] Rodari G. (2011), Il libro degli errori, Einaudi Ragazzi, Italia
- [32] Vankúš Peter (2005), History and Present of Didactical Games as a Method of Mathematics' teaching'

Ringraziamenti

Attenzione: I seguenti ringraziamenti non sono stati scritti con alcun ordine, a dire il vero non sono nemmeno sicuro che siano completi.

Vorrei ringraziare Silvia Benvenuti, la mia professoressa e la mia relatrice, per aver accolto con entusiasmo l'argomento che ho scelto per la tesi nonostante non fosse la sua "usual cup of coffee".

Ringrazio anche Maurizio Parton e Carlo Metta per avermi aiutato con la storia delle AI ed il laboratorio didattico che ho portato al Play di Modena, per cui ringrazio Rita Fioresi per avermi permesso di parteciparvi come divulgatore.

Ringrazio i miei genitori ed i miei familiari, per il costante supporto che mi hanno dato in questi anni, per i consigli e le opinioni.

Voglio ringraziare i miei colleghi del corso di didattica di matematica: i Pinguini Didattici Nucleari, con cui ho avuto modo di passare questi due anni in assoluta serenità.

Ringrazio i miei amici, bolognesi e non, per i litri di alcolici bevuti insieme, le ore di sonno non dormite ed i decibel di risate condivise.

Ringrazio la Siam Boxing, per avermi accolto nella loro palestra donandomi il calore che solo compagni di sport possono dare.

Ringrazio la musica per avermi accompagnato in questo viaggio.