

GIOVANNI GARBEROGLIO

## INTELLIGENZA ARTIFICIALE: UN NUOVO PARADIGMA DI CALCOLO AUTOMATICO

ABSTRACT - GARBEROGLIO G., 2017 - Artificial Intelligence: a new paradigm of automatic information processing.

Atti Acc. Rov. Agiati, a. 267, 2017, ser. IX, vol. VII, B: 55-69.

Recently, neural networks and their applications are becoming an ever increasing presence in everyday life. A lot of devices, from digital assistants to self-driving cars, are based on this new paradigm of information processing. In this paper, we will describe the principal characteristics of neural networks, we will discuss some of their applications and we will conclude with a bird-eye view of the discussions on philosophical and ethical problems raised by this technology.

KEY WORDS - Neural Networks; Artificial Intelligence; Self-driving cars.

RIASSUNTO - GARBEROGLIO G., 2017 - Intelligenza Artificiale: un nuovo paradigma di calcolo automatico.

Recentemente, le reti neurali e le loro applicazioni stanno trovando un sempre maggiore spazio nella vita quotidiana. Dagli ormai ubiquitari assistenti vocali alle macchine a guida automatica, tutta una serie di dispositivi moderni si basano su questo nuovo paradigma di calcolo. In questo contributo descriveremo le principali caratteristiche delle reti neurali, alcune loro applicazioni, finendo con alcuni accenni ai problemi filosofici ed etici che sono stati sollevati.

PAROLE CHIAVE - Reti Neurali; Intelligenza Artificiale; Automobili a guida autonoma.

### IL COMPUTER CHE VORREI

La fantascienza mi appassiona sin da giovane età. Fra tutti i *topoi* tipici di questo genere letterario, uno che sicuramente mi ha affascinato fin da subito sono stati i *computer parlanti*, specialmente quelli della serie *Star Trek*. Il mio primo computer è stato uno “Spectrum 48k” (sì, 48 kilobyte... oggi la SIM del mio telefono ha molta più memoria!) con cui, oltre che giocare, ho anche iniziato a programmare. Per quanto il BASIC fosse facile da imparare

e (relativamente) facile da usare per scrivere alcuni programmi elementari, quello che potevo farci era sicuramente molto più limitato (e assai meno soddisfacente) di quanto non potesse fare il Capitano Kirk dell'*Enterprise*: invece di comunicare col suo computer di bordo in un linguaggio strano (ma simile all'inglese) per mezzo di una tastierina di plastica, lui poteva addirittura parlarci!

Ah, se avessi avuto io a disposizione una simile tecnologia! Chissà quante cose noiose avrei potuto evitarmi (ad esempio, le traduzioni dal latino) e chissà quante cose interessanti avrei potuto imparare, usando il computer come un vero assistente che capisse al volo quello che volevo e mi aiutasse a soddisfare tutte le mie curiosità.

E invece per far fare qualcosa al mio "Spectrum" dovevo pensare, e non poco. Anziché parlargli nel mio linguaggio, ero costretto a parlargli nel suo; per ottenere una qualsiasi cosa dovevo aver analizzato nei dettagli tutti i passaggi necessari allo scopo e poi c'era la questione non banale della traduzione della sequenza delle operazioni da fare nel linguaggio a disposizione.

A onor del vero programmare un computer è una maniera utilissima per verificare se si è capito qualcosa, e questo modo di procedere è ormai diventato parte integrante della mia professione di fisico computazionale. Dopo ormai 30 anni, però, sono ancora al punto che devo dare al computer istruzioni dettagliatissime sulle cose da fare prima di aspettarmi di ottenere un qualsivoglia risultato. Ma perché alcune tecnologie mostrate in Star Trek (serie che risale agli anni '60 del secolo scorso) sono ormai di uso comune – come ad esempio i "comunicatori personali" (telefoni cellulari), gli auricolari senza fili o i tablet – mentre invece i computer parlanti ancora no?

O, meglio, non *ancora*?

Uno dei motivi principali si può ritrovare senz'altro nei limiti dell'approccio tradizionale alla programmazione, la cosiddetta *programmazione imperativa* secondo cui occorre elencare in un qualche linguaggio da computer le operazioni da fare e/o le decisioni da prendere ovvero, come si dice in gergo matematico, implementare l'*algoritmo* che risolve il problema che si è posti. Praticamente tutti i computer con cui si ha a che fare quotidianamente, compreso il proprio telefono, sono programmati in questo modo. Supponiamo di voler fare una chiamata col nostro telefono, solitamente aspetta un segnale dallo schermo sensibile al tocco; quando viene messo un dito sull'icona che rappresenta una cornetta viene eseguito il programma telefono, il quale si occupa di chiedere un numero, collegarsi alla rete telefonica, inoltrare la chiamata, trasmettere la voce e così via. Tutto questo viene fatto seguendo in maniera pedissequa le istruzioni inserite da un qualche programmatore. Ogni telefono di ogni persona fa esattamente la stessa cosa.

I programmatori si sono ben presto resi conto, però, che questo modo di procedere, era molto limitante, nel senso che sebbene si riescano a velocizzare in maniera incredibile operazioni generalmente ritenute noiose (ma precisamente definibili, come ad esempio le operazioni matematiche) si incontrano enormi difficoltà a far eseguire ad un computer compiti “relativamente” semplici, come ad esempio riconoscere le parole di un discorso oppure elencare i vari elementi che compongono un’immagine.

Se, per esempio, si volesse creare un programma per computer in grado di scrivere sotto dettatura, l’approccio tradizionale si troverebbe in difficoltà. Certo, ci si può rivolgere alla linguistica, la quale ci informa che le parole e le lettere corrispondono a fonemi – ad esempio la vocale “a” – e quindi il problema viene ricondotto a quello di far conoscere al computer i vari fonemi. Cosa ci può essere di più facile? Un vero e proprio “gioco da ragazzi” (anzi, forse addirittura un “gioco da bambini”). Ormai i computer sono così potenti da vincere a scacchi i migliori giocatori umani: potranno ben imparare qualcosa di così semplice, no?

La soluzione di questo problema, però, non è così diretta come potrebbe sembrare a prima vista: sebbene ciascuno di noi riesca a capire “al volo” il parlato di un’altra persona, questo avviene in maniera istintiva, e sicuramente non c’è dentro il nostro cervello un “programma” che viene eseguito e ci permette di capire le parole in un dettato. Anzi, come questo processo avvenga nel cervello è ancora un problema aperto di neurofisiologia. Ciò non toglie che comunque si possa tentare di trovare un algoritmo efficiente che trasformi le parole in lettere, ed effettivamente ci si è provato.

Ma purtroppo con scarsi risultati, dovuti principalmente al fatto che le varie persone parlano con velocità diverse, intonazioni diverse e accenti diversi. Se è “facile” per un essere umano interpretare il parlato “al netto” di tutte queste variabili, molto più difficile lo è per un computer che esegue un programma specifico, seguendo una procedura che debba valere per tutti i possibili interlocutori, che possono avere un’enorme varietà di intonazioni, inflessioni e toni di voce.

Eppure negli ultimi anni si è assistito ad una progressiva introduzione di “assistenti vocali” nei telefoni cellulari e nei computer. Chiunque abbia provato ad interagirci ha notato che ormai la comprensione del parlato è di livello molto buono: quasi tutti questi assistenti ormai trascrivono le istruzioni ricevute a voce, e questa trascrizione è decisamente fedele ed abbastanza indipendente dalle caratteristiche di chi parla (per la gioia di mia madre, che ha scoperto come videochiamarmi, semplicemente chiedendolo – per favore! – alla “vocina” dentro il tablet).

## ISPIRATI DALLA NATURA

Alla base di questi recenti successi, o anche di altri che ancora non hanno trovato la loro strada dal “laboratorio informatico” al grande pubblico, c’è un nuovo approccio alla programmazione, che trae ispirazione dalla natura invece che dalla matematica. Nessuno ha mai imparato a riconoscere il parlato, a classificare e riconoscere gli oggetti oppure a leggere perché gli è stato insegnato un algoritmo opportuno. Di solito invece si impara tramite esempi, esercitandosi, ed infine riuscendo a generalizzare quello che si è imparato. Sicuramente da piccolo mi hanno insegnato cosa fosse un cucchiaino, sia mostrandomelo che mostrandomene l’uso. Mi sono ovviamente esercitato a lungo coi cucchiaini che avevamo in casa e ora, in un ambiente dove non sono mai stato (come ad esempio al ristorante) riesco senza indugio a riconoscere un cucchiaino anche se non è esattamente uguale a quelli con cui ho avuto a che fare prima.

Nessuno, e men che meno io stesso, ha la più pallida idea di come avvenga questo processo, ma avviene e funziona molto bene. Molto, ma molto meglio di quanto non si riesca ad “insegnare” alle macchine, con gli algoritmi più raffinati e completi che si possano immaginare.

È chiaro però che all’origine di questa incredibile e molto raffinata abilità c’è il cervello e la sua struttura. Struttura che è molto diversa da quella dei microprocessori del computer. Questi ultimi si basano su “porte logiche”, ovvero piccolissimi circuiti (di dimensioni di qualche decina di miliardesimo di metro!) capaci di fare solo alcune semplicissime operazioni, come ad esempio applicare la tensione in un filo (di uscita) se c’è tensione su altri due fili (di ingresso) a cui sono collegati (porta AND), oppure applicare tensione su un filo se c’è tensione su almeno uno dei due fili (porta OR) e così via. Questi circuiti sono poi usati per costruirne di più grandi, ad esempio circuiti capaci di fare operazioni matematiche su numeri interi da 0 a  $2^{64}$ , numeri “a virgola mobile” e così via. Nei microprocessori moderni, come ad esempio quello del computer con cui sto scrivendo questo articolo, il numero di queste circuiti elementari è dell’ordine del miliardo.

Il cervello però non funziona così. Non ci sono porte logiche o transistor al silicio, ma un sacco di cellule collegate tra di loro con dei filamenti. Alcuni di questi filamenti prendono segnali biochimici dalle cellule vicine (ingressi) e in base a questo mandano a loro volta segnali ad altre cellule (uscite). Il tutto avviene continuamente, modificandosi ed adattandosi alle condizioni dell’ambiente in cui si sta vivendo e all’attività che si sta svolgendo. I neuroni sono sicuramente diversi come funzionamento dalle porte logiche, ma parecchie attività (ad esempio eseguire somme e moltiplicazioni) possono essere fatti da entrambi. Da un punto di vista

matematico, si può dimostrare che usando le porte logiche elementari si può calcolare qualsiasi funzione, ovvero, traducendo in italiano, svolgere qualsiasi compito.

E coi neuroni?

Le prime idee sull'utilizzo di architetture informatiche "simili ai neuroni" per effettuare calcoli e classificazioni risalgono agli anni '50 del secolo scorso. Nessuno sapeva (sa?) esattamente come funzionano i neuroni, quindi i primi modelli di circuiti elettronici simili ai neuroni, dovevano effettivamente basarsi su approssimazioni. Un modello che si è poi rivelato vincente era il cosiddetto *perceptrone*, una specie di porta logica con però più ingressi i quali potevano assumere più valori, invece che gli 0 e 1 (spento e acceso) delle porte logiche. Il perceptrone elabora i valori dei suoi ingressi combinandoli con dei suoi pesi nella maniera più semplice possibile (lineare, per chi mastica la matematica) ed in base al risultato ottenuto da questa operazione setta un valore per il canale in uscita. Uno schema di un perceptrone è riportato in Fig. 1.

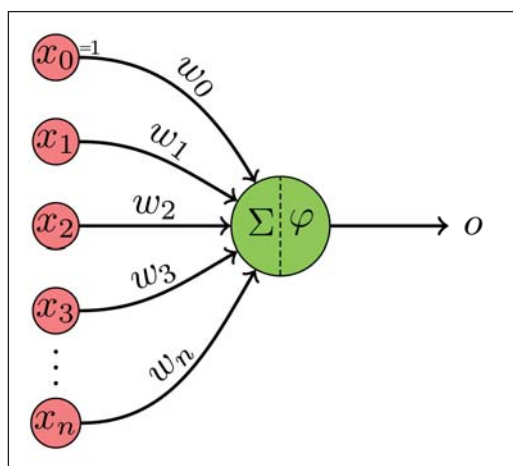


Fig. 1 + Schema di un perceptrone con input  $x_1, \dots, x_n$ ; pesi  $w_1, \dots, w_n$ ; e funzione di trasferimento  $\phi()$ .  
[da Wikipedia]

Da un punto di vista matematico, si può dimostrare che, analogamente alle porte logiche, un insieme opportunamente grande di perceptroni è in grado di calcolare qualsiasi funzione. Tradotto in italiano: può arrivare a qualsiasi conclusione o prendere qualsiasi decisione sulla base degli ingressi; in altre parole un insieme di perceptroni è garantito avere una grande flessibilità, analoga alla plasticità osservata nel cervello umano. Questo significa che *in linea di principio* è possibile organizzare una serie di perceptroni in modo che, a seconda dei pesi con cui "elaborano" i loro ingressi, riescano a fare qualsiasi attività: ad esempio trasformare il parlato in scritto, riconoscere

la presenza di oggetti specifici in immagini, tradurre automaticamente da una lingua all'altra, e così via.

Ma tra il dire ed il fare c'è di mezzo il mare, e sebbene *sulla carta* le potenzialità dei perceptron fossero incredibili, in pratica non c'era nessun criterio per stabilire quali fossero i collegamenti ottimali, oltre che i pesi corretti, per riuscire a svolgere le attività anche più semplici. Oltre a questo c'è da dire che negli anni '50 e '60 non era chiaro come implementare in pratica questi costrutti matematici. Per le porte logiche c'erano le valvole (ed i transistor), ma per i perceptron? Nessuno aveva la più pallida idea di come costruire dei circuiti elettrici che facessero tutte queste cose. Il "piano B" poteva consistere nel far eseguire le operazioni dei vari perceptron ad un computer, ma a quell'epoca i computer erano molto lenti e poi rimaneva la questione di come decidere quali fossero i pesi da usare. Furono fatte alcune proposte, ma gran parte delle idee vennero lasciate sul tavolo. Per circa una ventina d'anni.

Verso la fine degli anni '80, e grazie anche ai progressi dell'informatica "tradizionale", queste idee vennero ripescate e si inventò un algoritmo molto efficiente in grado di trovare rapidamente dei pesi ragionevolmente corretti per un determinato scopo usando una procedura molto simile a quella che si usa per insegnare ai bambini: si mostrano degli esempi e si calcola la risposta del sistema, in base alla quale vengono modificati i pesi stessi in modo da ottenere risposte via via sempre più precise. In questo modo il sistema di perceptron o, come viene chiamato attualmente, la *rete neurale*, è in grado di "imparare" in base all'esperienza.

Questo modo di procedere sostituisce l'approccio tradizionale alla programmazione con una metodologia molto diversa, in cui la rete neurale viene messa in grado di trovare da sola tutti i parametri necessari per poter svolgere il suo compito senza che il programmatore debba sforzarsi di analizzare e descrivere a priori tutte le possibilità. La difficoltà viene spostata dallo sviluppo dell'algoritmo (ad esempio, quello per trascrivere un messaggio vocale) alla progettazione e l'allenamento di una rete neurale sufficientemente complessa da poter effettuare il compito con efficacia, ma non troppo complessa da avere un numero ingestibile di parametri.

Infatti ci si rese ben presto conto che per quanto questa procedura possa funzionare *in principio*, in pratica sono necessarie reti abbastanza complicate, con un numero molto grande di parametri. A titolo di esempio, una delle reti neurali più usate per riconoscere i numeri di 0 a 9 scritti a mano contiene diverse decine di migliaia di pesi, ma dopo essere stata addestrata con 60.000 esempi, è in grado di riconoscere le cifre con un'accuratezza di oltre il 99%. Per compiti molto più complessi, come ad esempio la classificazione degli oggetti presenti in un'immagine, il numero di parametri che

descrivono la rete può essere molto grande, dell'ordine di diverse *decine di milioni*. Come si può ben immaginare, riuscire a trovare la combinazione giusta di tutti questi parametri che permettano alla rete neurale di svolgere bene un determinato compito è un processo abbastanza impegnativo.

Gli studiosi di questo campo hanno trovato un alleato inaspettato, già bene presente sotto gli occhi di tutti, che ha permesso – insieme alla costante miglioria dei metodi di “insegnamento” – di poter arrivare a creare reti neurali abbastanza complicate da poter competere con gli umani in compiti ritenuti impossibili da risolvere con un procedimento puramente algoritmico. L'alleato in questione consiste nelle *schede grafiche* presenti in ogni computer. Negli ultimi 20 anni, grazie alla pressante richiesta di videogiochi dotati di visualizzazione sempre più realistica, le schede grafiche hanno subito uno sviluppo impressionante. I processori presenti in questo tipo di hardware sono stati progettati per effettuare sempre più velocemente tutta l'impressionante mole di calcoli necessaria per produrre rappresentazioni tridimensionali accurate di ambienti, oggetti e personaggi. Le operazioni matematiche per questo tipo di visualizzazioni sono essenzialmente operazioni di *algebra lineare*, proprio le stesse necessarie nella fase di addestramento e di modellizzazione di una rete neurale. Le schede grafiche più moderne sono in grado eseguire le operazioni necessarie per addestrare una rete neurale quasi 100 volte più velocemente dei comuni processori “generalisti”, ed il loro utilizzo negli ultimi anni è risultato in un enorme sviluppo di reti neurali “profonde”, in cui i “neuroni” (perceptroni) sono partizionati in diverse sottounità capaci ciascuna di un proprio compito, analogamente a quello che le recenti scoperte di neurobiologia dicono essere la struttura del cervello.

#### COMPUTER ALLA “CONQUISTA” DEL MONDO

E i risultati non si sono fatti attendere. Oltre ad affrontare il problema classico della classificazione (di oggetti in immagini, o di fonemi nel parlato) si è provato ad affrontare con le reti neurali alcuni problemi ritenuti “impossibili” da risolvere con computer tradizionali.

Per quanto mi riguarda, tra i recenti successi dell'approccio neurale (o *deep learning*, per chi mastica l'inglese) ho trovato particolarmente interessante lo sviluppo di un sistema in grado di giocare efficacemente a *Go*. Il *Go* è un gioco da tavolo molto diffuso in Asia, si gioca su una “scacchiera” tipicamente di 19x19 elementi su cui i contendenti dispongono delle pedine (bianche per uno, nere per l'altro), le quali possono solo venire rimosse quando sono circondate da pedine di un altro colore. Dal punto

di vista matematico, il gioco del *Go* ha moltissime più mosse possibili che non il gioco degli scacchi, e questo rende un'analisi esaustiva praticamente impossibile anche coi più potenti computer in circolazione. Per questo motivo fino a pochi anni fa una persona anche solo minimamente esperta del gioco riusciva senza problemi a battere qualsiasi computer.

Fino a che, come potete immaginare, non sono intervenute le reti neurali, ed in particolare la rete chiamata AlphaGo e sviluppata nientepodimeno che da Google. Dopo aver "diligentemente" imparato a giocare prendendo spunto da 230.000 partite giocate dai migliori campioni di tutti i tempi, AlphaGo è riuscito a sconfiggere i migliori campioni asiatici, anche se non senza un gran dispiegamento di forze: il programma che implementava la rete neurale girava su quasi 180 schede grafiche in parallelo, con una potenza di calcolo pari a quella di circa 15000 normali processori di computer. Certo, un'impressionante "potenza di fuoco", ma praticamente nulla se consideriamo che l'impresa di battere il campione mondiale di *Go* era ritenuta fuori discussione anche con l'utilizzo dei più potenti supercomputer del mondo (che arrivano tranquillamente a mettere in campo potenze di calcolo pari a *decine di milioni* di comuni processori).

Non paghi dei successi ottenuti, i programmatori di Google si sono messi in testa di fare ancora meglio, ed hanno sviluppato una rete neurale che ha *imparato da sola* il *Go*, sfidando ripetutamente un'altra versione di se stessa ed avendo come conoscenza iniziale solamente le regole del gioco (mosse possibili e criterio di vittoria). Dopo 21 giorni ininterrotti di "autoallenamento" questa nuova rete neurale, chiamata AlphaGo Zero, si è dimostrata in grado di sconfiggere sonoramente tutte le "vecchie versioni" di Alpha Go, oltre che a vincere 60 incontri *consecutivi* con i più forti campioni mondiali.

Per quanto questo esempio possa sembrare "accademico", in realtà le reti neurali stanno silenziosamente entrando nella vita di tutti i giorni, con un effetto che probabilmente sarà dirompente. A parte gli ubiquitari assistenti vocali di cui ho già accennato più sopra, una delle "rivoluzioni prossime venture" riguarda sicuramente le automobili a guida automatica. La mia impressione è che qui nella cara e vecchia Europa si sia rimasti un po' indietro, ma negli Stati Uniti ci sono diversi progetti in stato avanzato di sviluppo. Intendo proprio con le macchine che circolano già per le strade. Mi è capitato di vederne alcune durante una mia visita a Pittsburgh più di un anno fa (ad uso dei miei futuri biografi, trattasi dei mesi di Settembre ed Ottobre del 2016), quando era appena iniziata una sperimentazione sul campo condotta da Uber (l'azienda di trasporti automobilistici privati) in collaborazione con l'Università Carnegie Mellon, una delle più importanti e famose università mondiali per la *computer science*.



In seguito all'accordo tra gli enti di sviluppo/ricerca e la città di Pittsburgh, queste macchine dovevano avere un pilota umano per i casi di emergenza. Potete immaginare la sensazione “strana” che mi ha colto quando ho visto queste macchine passare per la prima volta: nel sedile posteriore passeggeri intenti a fare le loro attività, col posto del guidatore occupato da una persona *a braccia conserte*. Dopo l'iniziale stupore è però subentrata una certa curiosità: ma come “guidano” queste macchine-robot? Pur premesso che il traffico in una metropoli americana non ha niente a che vedere col traffico di una qualsiasi città italiana (i motorini e le moto sono pochissimi e l'americano medio alla guida è rispettoso del codice della strada e decisamente gentile), mi sono presto trovato ad apprezzare la facilità e la scioltezza con cui queste macchine si muovevano nel traffico... e dopo un po' anche i loro limiti.



Fig. 2 - Una macchina della collaborazione Uber/Carnegie Mellon per le strade di Pittsburgh (fonte: newsroom.uber.com).

Ricordo di un episodio che mi ha particolarmente colpito: stavo andando all'Università che mi ospitava, passeggiando alla volta di un incrocio. Davanti a me una di queste macchine occupa la fila di “svolta a sinistra”, con freccia d'ordinanza e fermandosi prima della linea di stop con una precisione millimetrica. Aveva il semaforo rosso. “Chissà come agisce ora?”, mi sono chiesto. Non appena scattato il verde, le macchine provenienti in direzione contraria sono partite, e la macchina-robot si è tranquillamente disposta ad aspettare di avere il via libera, dando giustamente la precedenza a chi ne aveva il diritto. Un qualsiasi pilota un po' esperto in questi casi sa cosa fare: crearsi un'opportunità di ingresso. Nessuno lo ha però insegnato

alla macchina, che fa l'unica cosa che "le regole" le concedono: aspettare di avere la via libera. Così il semaforo diventa rosso, ritorna il verde, e ridiventa rosso, senza che la macchina si sia mossa di un millimetro. A questo punto noto uno scambio verbale tra il cliente e l'autista, il quale prende in mano le redini della situazione: occupa l'incrocio in maniera più decisa e, scattato il verde, svolta velocemente prima che arrivino le altre macchine.

Questo episodio mi ha dato da pensare, e mi ha portato ad informarmi meglio. In effetti tra i vari requisiti con cui le macchine-robot vengono sviluppate c'è quello di essere perfettamente ligie al codice della strada, ma chiunque abbia mai guidato sa bene che in certi casi (come appunto quello della svolta a sinistra) il rispetto delle regole si scontra con la necessità dell'azione, il che porta a prendere decisioni come quella messa in opera dal pilota di cui sopra. Il giorno in cui le macchine-robot fossero rilasciate per strada senza supervisione e si trovassero a condividerla con piloti umani, sicuramente qualche incomprensione potrebbe succedere. Anzi, è già successa, e la cronaca riporta i casi di alcuni incidenti che hanno visto coinvolte queste macchine-robot, con commenti che vanno dal "colpevolizzante" (l'errore è sempre umano) al "catastrofico" (ecco, vedete, orde di macchine-robot-killer che non sanno quello che fanno stanno per invadere le nostre strade), passando per il "si comportano in un modo strano" (il che, ad un'analisi più accurata, si è rivelato essere equivalente a "sono anche troppo ligie al codice della strada").

#### DI CHI È LA COLPA?

Oltre ai giornalisti, anche i filosofi si sono cimentati sulla questione delle macchine a guida autonoma, che presenta secondo l'opinione di chi scrive alcuni spunti interessanti. Il primo che vorrei segnalare riguarda la morale: che sia con metodi tradizionali o con reti neurali, le macchine robot andranno in qualche modo programmate (o istruite) e si troveranno a prendere decisioni, che in alcuni casi potrebbero riguardare la vita o la morte del conducente o di altre persone presenti. Che cosa si fa in questo caso? Si decide che la priorità è quella della vita del conducente? Oppure posti di fronte al rischio di mettere a repentaglio la vita di (molte) altre persone, sarebbe il caso di assicurarsi che la programmazione della macchina-robot preveda l'autosacrificio, di sé e delle persone a bordo?

Detta così la faccenda si presta evidentemente ad interminabili discussioni e solleva problemi etici di non trascurabile rilevanza. In ambiente anglosassone questa questione è nota come *il problema del carrello*: si immagina di avere un carrello su dei binari che si dirigono verso un bivio. In un lato

c'è un gruppo di persone, nell'altro un bambino. Avendo la possibilità di scegliere, cosa si sceglierebbe?

Non sono sicuro di avere una risposta che sarei anche in grado poi di mettere in pratica, noto solamente che quando ho preso la mia patente di guida non c'era nessun capitolo sui "Dilemmi morali e loro soluzione" nel libro da studiare, e l'unico criterio che è stato usato per conferirmi il diritto di guidare un'automobile è stato quello di sincerarsi che sapessi riconoscere i cartelli stradali e che fossi in grado di scorrazzare per la città il mio istruttore e l'ingegnere per una mezz'ora.

Non vedo perché lo stesso criterio non possa essere usato anche per le macchine robot. Anzi, si potrebbe pure argomentare che il numero di incidenti in cui siano mai state coinvolte le macchine-robot è molto, ma molto minore del numero di incidenti che ci si aspetterebbe su basi statistiche per un insieme di auto di paragonabile entità guidate da umani per lo stesso periodo. Non credo di aver mai sentito parlare di incidenti in cui le persone coinvolte abbiano portato traumi di una certa rilevanza. E le prime statistiche mostrano che nella stragrande maggioranza dei casi la colpa dell'incidente veniva da errate scelte da parte dei guidatori umani coinvolti. Forse più che considerare i casi estremi, si potrebbe iniziare a ragionare sui benefici *generali* che l'aver in circolazione una schiera di piloti molto, ma molto, ligi alle regole della strada (limiti di velocità inclusi) potrebbe portare alla popolazione.

Invece gran parte della discussione verte su cosa poi debba accadere una volta stabilita che la colpa di un certo incidente debba ricadere sulla macchina-robot. In questo caso, chi ne è il responsabile? Il passeggero (magari anche proprietario della macchina), oppure la ditta costruttrice, a cui in ultima analisi è imputabile l'origine del comportamento dei propri prodotti? A questo proposito mi si permettano un paio di considerazioni. In primis, noto che la legge mi impone di pagare ogni anno un'assicurazione sulla responsabilità civile per l'automezzo di mia proprietà. Nel caso di macchine-robot, direi che quest'onere debba ricadere su chi guida, e qui potrebbero verificarsi varie possibilità: o la macchina è *completamente* automatica e quindi la responsabilità civile è del produttore, oppure la macchina prevede la possibilità che il pilota prenda il comando, ed a quel punto l'assicurazione andrebbe pagata da entrambi. Non ho molti dubbi sul fatto che le case automobilistiche, qualora fosse dimostrata la generale tendenza delle macchine-robot a non essere causa di incidenti, riuscirebbero ad ottenere dalle compagnie di assicurazione delle tariffe estremamente agevolate.

Concordo con tutti i pessimisti che stiano leggendo queste mie parole: il vero problema è la responsabilità *penale*. Cosa succede se negli incidenti

ci scappa il morto? E come si risolve il problema del carrello? Anche qui, lascio a chi di dovere discutere la questione, ma metterei sul campo un paio di considerazioni: cosa succede oggi, con gli umani alla guida delle automobili in strada, in questi casi? Beh, chi decidesse di risolvere una questione sacrificandosi verrebbe considerato un eroe, mentre chi si salvasse a scapito di innocenti spettatori verrebbe probabilmente processato per omicidio. Una possibile soluzione sarebbe quindi quella di prevedere che le macchine possano agire in entrambi i modi, e quindi decidere di tutelare i passeggeri oppure (provare a) minimizzare le vittime, arrivando fino alla soluzione estrema dell'autosacrificio, lasciando al proprietario la scelta della configurazione del veicolo e quindi la responsabilità delle conseguenze.

Detto così sembra semplice, ma come spesso accade nella scienza (e nella morale), ogni soluzione di problemi (vera o presunta, come in questo caso) apre molte più domande di quante non se ne risolvano. E la questione che subito emerge è estremamente tecnica, ma estremamente importante. Per vedere da dove nasce, basta considerare la *nonchalance* con cui ho espresso la proposta di una macchina-robot configurabile con due tipi di comportamento. In un'ottica di programmazione "tradizionale" il comportamento di un software è perfettamente descritto dalla sequenza delle istruzioni che lo determinano, e le varie scelte sono espresse in maniera non ambigua come parte di queste istruzioni.

Nel creare un programma tradizionale si parte quindi da una descrizione astratta di quello che deve succedere (ad esempio, vogliamo costruire un programma che giochi a scacchi) che diventa via via più concreta man mano che i moduli e le loro interazioni vengono espresse in un qualche linguaggio di programmazione (analizzare quante più possibili sequenze di mosse future, guardare quale di esse darebbero alla fine un vantaggio migliore e fare la mossa che corrisponde al primo passo della sequenza). Nonostante la teoria sia corretta si commettono molto spesso – ahimé – degli errori di programmazione, ed è per questo che è buona norma *verificare* che alla fine il programma che si è creato soddisfi effettivamente i requisiti iniziali. Qualora questo non succeda (e non succede quasi mai), si tratta di andare a vedere quale istruzione del programma sia stata data nel modo sbagliato, correggerla e riverificare. Questa parte dello sviluppo di software, chiamata *debugging* (= togliere gli insetti, perché in uno dei primi calcolatori successe che una serie di comportamenti inaspettati fossero proprio dovuti alla presenza di un insetto tra i circuiti) è fondamentale per avere alla fine un prodotto che sia conforme alle aspettative con un altissimo grado di fiducia.

Le reti neurali, però, non seguono alcun programma predefinito: la loro risposta a determinate situazioni risulta come effetto di un calcolo complicato in cui intervengono migliaia (o milioni) di unità semplici come quella

riportata in Fig. 1, i cui pesi sono determinati in base al risultato voluto e agli esempi mostrati. La situazione è più simile a quella di un cervello umano che non a quella di un listato di un programma e analogamente ad un cervello umano risulta spesso difficile capire *perché* una rete neurale abbia avuto una specifica risposta, qualora questa non corrisponda alle specifiche. Il debugging delle reti neurali non è così “semplice” come con i programmi tradizionali, e può arrivare a sembrare qualcosa di più vicino alla neurofisiologia che non all’informatica, dal momento che di solito si cerca di capire se ci siano specifiche zone della rete neurale che processano certe caratteristiche dell’input, quali di queste aree siano responsabili del risultato “sbagliato” e cosa si debba fare, nell’eventualità di cambiare la struttura della rete, per minimizzare l’errore osservato.

A differenza delle scienze della mente (umana) però qui si ha accesso alla descrizione completa della rete neurale, e si può quindi sperare di riuscire a farsi un’idea del perché e del percome una certa rete abbia fornito un certo risultato, ovvero abbia preso una certa decisione. Pare addirittura che l’Unione Europea stia per passare una legge che sancisce il *diritto ad avere una spiegazione*, qualora sia stata presa una decisione riguardo ai propri cittadini. Se questa spiegazione può essere (relativamente) facile da produrre nel caso in cui un essere umano l’abbia presa, diventa molto più difficile produrla quando proviene da elaborazioni fatte per mezzo di reti neurali. Anche senza scomodare le macchine-robot, parecchie decisioni vengono ormai prese sulla base di sofisticati (?) modelli neurali addestrati con una mole impressionante di dati. Ad esempio è possibile che dietro alla decisione di una banca di concedere o meno un mutuo ci sia (anche) la stima effettuata da parte di una rete neurale allenata per riconoscere i clienti insolventi da quelli ammodino. Purtroppo però non si è ancora in grado di stabilire in generale come e perché una rete neurale abbia prodotto un determinato output. La ricerca, però, prosegue imperterrita.

#### LA FANTASCIENZA PROSSIMA VENTURA

Come dovrebbe essere chiaro dai discorsi affrontati finora, quello dell’*intelligenza artificiale* è un campo che sta avendo uno sviluppo impressionante, e che già adesso pone questioni su temi importanti, come ad esempio la morale.

Tra i miei libri preferiti c’è di sicuro *Turbare l’universo* di Freeman Dyson, un fisico molto famoso, ma soprattutto un visionario di prim’ordine. In parecchie delle sue opere non specialistiche, Dyson eccelle nell’immaginare scenari plausibili e non-banali di possibili futuri, descrivendoli con una

prosa accattivante ed una notevole attenzione ai dettagli. Non oso certo anche solo provare a competere, ma vorrei cogliere l'occasione per porre sul tavolo una questione che probabilmente prima o poi si porrà. Anzi è già stata posta.

Può una macchina pensare?

In generale, mi sembra che sia molto diffusa l'opinione che il pensiero è una delle attività tipiche e caratterizzanti degli esseri umani, quello che forse di più ci distingue da tutti gli altri animali. Nonostante questo, però, sembra che il pensiero (nel senso di intelligenza, o addirittura di autocoscienza) non sia una cosa facilmente spiegabile o comprensibile. Cosa vuol dire pensare? Cosa vuol dire essere autocoscienti? Penso che ognuno di noi, se non interrogato, abbia almeno un'idea di quale possa essere la risposta. Ma quando si tratta di metterla nero-su-bianco, ci si accorge che la situazione non è facilmente esprimibile. D'altra parte i filosofi ci campano da centinaia di anni.

Agli albori dell'informatica alcuni visionari si posero il problema di cosa volesse dire essere intelligenti. Una delle risposte più famose a questa domanda venne data da Alan Turing, il quale semplicemente affermò che una macchina può essere considerata intelligente come un umano quando un essere umano non è in grado di distinguerla da un altro umano facendole delle domande. Questa procedura è nota come *test di Turing*, e di sicuro l'assistente vocale del computer con cui sto scrivendo questo contributo non la supera nemmeno per sbaglio.

I progressi in questo campo però sono, come appunto dicevo, impressionanti e alcuni ricercatori sono arrivati addirittura a stimare che entro il 2050 si riuscirà a creare una macchina dotata di un'intelligenza uguale o superiore a quella umana. Non chiedetemi di valutare la verosimiglianza di questa previsione, ma trovo molto interessante il solo fatto che sia stata messa sul tavolo. In effetti, cosa vuol dire essere intelligenti? Cosa vuol dire pensare? Finora abbiamo avuto a che fare con un solo esempio di intelligenza (quella umana, vera o presunta che sia) e quindi più o meno ogni teoria a riguardo è stata basata su questo unico dato. Che l'intelligenza umana sia rappresentativa dell'intelligenza *in generale* è tutto da verificare. Richard Feynman, uno dei più grandi fisici del XX secolo, è famoso per aver detto: «quello che non riesco a creare, non lo capisco». Sono convinto che lo sforzo di creare una macchina intelligente porterà sicuramente nuova linfa agli studi sull'intelligenza, sulla coscienza, e – probabilmente – avrà anche la conseguenza di rendere un po' più umile il nostro approccio alla questione. Alcuni arrivano addirittura a sostenere che la mera creazione di una macchina intelligente sarà il primo passo per un accrescimento *esponenziale* dell'intelligenza: così come noi intelligenti esseri umani ab-

biamo progettato una macchina più intelligente di noi, così la macchina riuscirà a concepire e progettare un'altra macchina più intelligente e così via.

Come si può ben immaginare, questi scenari si coniugano facilmente con previsioni catastrofiche e non posso che rimandare il lettore interessato a ricercare informazioni sulla (presunta) *Singolarità Tecnologica*. Trovo che siano letture degne della miglior fantascienza. Nel mio piccolo, penso che qualora si riuscisse a creare una macchina intelligente quanto o più degli esseri umani, questa si comporterà più o meno come descritto nel film *Lei*, invece che diventare una minaccia per la specie umana. Quest'ultimo è il leit-motiv di una serie fantascientifica molto famosa – e molto ben fatta, secondo l'umile parere dello scrivente – *Battlestar Galactica*. Personalmente trovo molto intrigante (e plausibile) l'idea ivi espressa che se dei “figli dell'umanità” arriveranno mai ad esplorare lo spazio, saranno fatti di silicio invece che di carbonio.

Non sono sicuro di aver dato un notevole contributo allo sviluppo della scienza; spero almeno di aver dato qualche buon consiglio su come rilassarsi (in modo intelligente?) davanti alla televisione nel caso di domeniche piovose.

#### BIBLIOGRAFIA, SITOGRAFIA, FILMOGRAFIA

AA.VV., 2004 - *Battlestar Galactica (reimagined)*.

EDDMONDS D., 2017 - *Uccideresti l'uomo grasso?* Raffaello Cortina Editore.

FREEMAN D., 2010 - *Turbare l'universo*. Bollati Boringhieri.

GARBEROGLIO G., 2014 - *Zio, ma cosa fai di Lavoro?* Accademia Roveretana Agiati, ser. IX, vol. IV, B.

GARBEROGLIO G., 2017 - *Mobility Story*. FBK Magazine. <https://magazine.fbk.eu/it/news/mobility-story-giovanni-garberoglio-1-prepariamoci-partire/>.

HARDESTY L., 2017 - *How neural networks think*. MIT news. <http://news.mit.edu/2017/how-neural-networks-think-0908>.

JONZE S., 2013 - *Lei (Her)*.

WIKIPEDIA. *Singolarità Tecnologica*. [https://it.wikipedia.org/wiki/Singolarità\\_tecnologica](https://it.wikipedia.org/wiki/Singolarità_tecnologica).

