

APPUNTI SU

RETI NEURALI E CONNESSIONISMO[†]

MARCELLO FRIXIONE

Versione di ottobre 2009

[†] *Questi appunti sono ricavati dall'introduzione a Paul Smolensky, Il connessionismo tra simboli e neuroni, Marietti, Genova, 1992, traduzione italiana di "On the proper treatment of connectionism", Behavioral and Brain Sciences, 11, 1988, e dal capitolo "Connessionismo", in S. Gensini e A. Rainone (a cura di), Filosofie della Mente. Tradizione e attualità, Carocci, Roma, 2008.*

Nell'ambito delle scienze cognitive classiche l'adozione di un'impostazione funzionalista si è spesso accompagnata a una deliberata (e ingiustificata) indifferenza nei confronti delle informazioni che le conoscenze sul cervello e sul sistema nervoso potevano offrire per la comprensione dei processi cognitivi. Tale atteggiamento venne esplicitamente messo in discussione dal *connessionismo*, una tendenza affermatasi nel corso degli anni ottanta del novecento (l'imporsi del connessionismo nella comunità scientifica si può far risalire alla pubblicazione di Rumelhart e McClelland 1986), ma che può essere considerata l'erede di ricerche precedenti, le quali erano state poste in ombra dal successivo affermarsi delle scienze cognitive classiche. Si tratta di ricerche che erano state sviluppate nell'ambito della cibernetica, a partire dagli studi pionieristici sui neuroni artificiali di McCulloch e Pitts, o in anni ancora precedenti (sugli antecedenti remoti del connessionismo si veda Cordeschi 2002; per una trattazione approfondita dei vari aspetti del connessionismo si veda Bechtel e Abrahamsen 2002; per situare questa tendenza nel contesto generale delle scienze cognitive e della loro storia si vedano ad esempio Marraffa 2003 e Abrahamsen *et al.* 2004).

I connessionisti condividono l'ipotesi computazionale alla base delle scienze cognitive, ma propongono l'impiego di architetture computazionali che si discostano da quelle utilizzate nelle scienze cognitive classiche. In particolare, viene messa in discussione l'adeguatezza delle architetture computazionali di tipo sequenziale sulla base del fatto che il sistema nervoso è un dispositivo di calcolo altamente parallelo. Negli esseri umani il numero dei neuroni è di un ordine di grandezza stimabile tra 10^{10} e 10^{11} ; ciascun neurone si comporta come una singola unità di calcolo, che opera in parallelo con gli altri. I neuroni sono connessi tra loro a formare una fitta rete: ciascun neurone ha decine di migliaia di sinapsi in entrata e in uscita, mediante le quali scambia i propri input e i propri output con i neuroni adiacenti. Per contro, ogni singolo neurone costituisce un dispositivo di calcolo relativamente semplice. La complessità dei meccanismi cognitivi è resa possibile dall'interazione di un grande numero di elementi fittamente interconnessi.

Inoltre, a differenza di quanto avviene ad esempio nei calcolatori di von Neumann, dove memoria e unità di calcolo (CPU) sono componenti rigidamente distinte, una distinzione netta tra memorizzazione delle informazioni e loro elaborazione non è presente nel sistema nervoso. Nel cervello non esiste alcun dispositivo centralizzato per il controllo dell'elaborazione.

1. I limiti dell'architettura di von Neumann

La struttura di un calcolatore di von Neumann è raffigurata, molto schematicamente, nella fig. 1. Opportuni dispositivi di *input* e di *output* permettono di accedere dall'esterno alla memoria del calcolatore, consentendo, rispettivamente, di inserirvi e di estrarne dei dati. Le informazioni contenute in memoria vengono elaborate da una singola unità di calcolo (detta CPU - *Central Processing Unit*), che opera sequenzialmente su di essi. La caratteristica più importante della macchina di von Neumann è costituita dal fatto che sia dati che programmi vengono trattati in modo sostanzialmente omogeneo, ed immagazzinati nella stessa unità di memoria. Così, quando un programma deve essere eseguito, l'unità di calcolo lo reperisce in memoria, e lo applica quindi ai dati, anch'essi conservati in memoria. Questo consente una grande flessibilità al sistema. Ad esempio, poiché dati e programmi sono oggetti di natura omogenea, è possibile costruire programmi che prendano in *input* altri programmi e li elaborino, e che producano programmi in *output*. Queste possibilità sono ampiamente sfruttate negli attuali calcolatori digitali, e da esse deriva gran parte della loro potenza e della loro facilità d'uso (ad esempio, un compilatore o un sistema operativo sono essenzialmente programmi che operano su altri programmi).

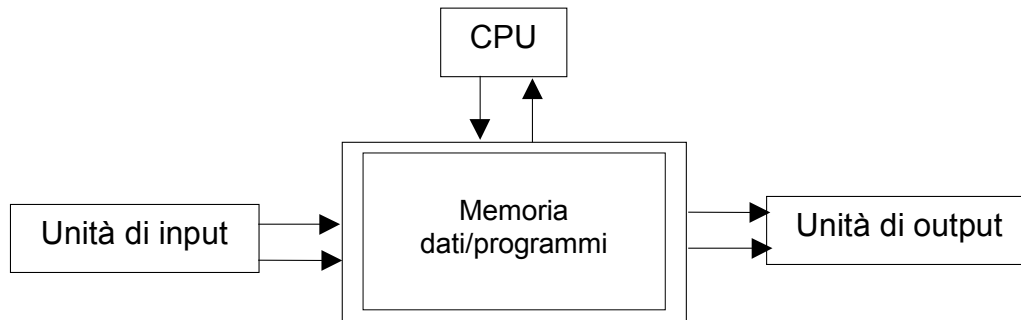


Fig. 1

Benché siano estremamente versatili, alle macchine con architettura di von Neumann sono stati imputati alcuni limiti dal punto di vista informatico. In particolare, è stata criticata la netta separazione tra immagazzinamento ed elaborazione dei dati che questo tipo di architettura comporta. In un calcolatore di von Neumann memoria e unità centrale di calcolo (CPU) sono due componenti rigidamente distinte. L'unità di calcolo attinge di volta in volta ai dati contenuti nella memoria, ma quest'ultima rimane sostanzialmente passiva durante la maggior parte della durata del calcolo. Si tratta del cosiddetto problema del "collo di bottiglia" della macchina di von Neumann: le informazioni vengono elaborate solo quando vengono richiamate dalla CPU del sistema. Ciò comporterebbe problemi di velocità di calcolo e di efficienza nello sfruttamento delle risorse computazionali.

Queste limitazioni hanno un corrispettivo dal punto di vista dello studio computazionale della mente. Una distinzione netta tra memorizzazione delle informazioni e loro elaborazione è difficilmente giustificabile sulla base delle conoscenze disponibili sul sistema nervoso. Nel cervello non esiste alcun dispositivo centralizzato per il controllo dell'elaborazione. Le operazioni computazionali nel sistema nervoso sembrano demandate ad un meccanismo di controllo altamente distribuito. Inoltre, non esiste una separazione netta tra dispositivi per la memorizzazione e per l'elaborazione delle informazioni. Ciò richiama altri problemi posti dai modelli computazionali dell'IA e della scienza cognitiva tradizionale, dovuti alla mancanza di plausibilità dal punto di vista anatomico e neurofisiologico del paradigma computazionale di von Neumann.

Il punto centrale è che il cervello è un dispositivo di calcolo *altamente parallelo*. Il numero dei neuroni è di un ordine stimabile tra 10^{10} e 10^{11} , e ciascuno di essi si comporta come una singola unità di calcolo, che lavora contemporaneamente a tutte le altre. I neuroni sono altamente interconnessi: ogni neurone ha molteplici sinapsi in entrata e in uscita, mediante le quali scambia i propri *input* e i propri *output* con gli altri neuroni. L'alto parallelismo della computazione neuronale è legato al fatto che ogni singolo neurone è un dispositivo di calcolo poco potente, in grado di calcolare soltanto funzioni estremamente semplici. La complessità dei meccanismi cognitivi deve quindi essere garantita dall'interazione di un grande numero di neuroni. Inoltre, i neuroni sono lenti rispetto ai calcolatori digitali: la durata di una singola "operazione" in un neurone viene stimata nell'ordine della decina di millisecondi. Perciò, tutti i fenomeni cognitivi di durata compresa in un secondo devono richiedere un calcolo la cui lunghezza sia dell'ordine di circa cento passi (Feldman e Ballard 1982). Questo non può essere certamente ottenuto se non mediante un calcolo di tipo parallelo. Per tali ragioni, un modello computazionale rigidamente sequenziale come quello di von Neumann non può essere adeguato a spiegare come viene elaborata l'informazione nel sistema nervoso.

Problemi di questo genere non erano ritenuti pertinenti da parte degli scienziati cognitivi di impostazione tradizionale, che condividevano in maniera più o meno esplicita l'assunzione funzionalista dell'indipendenza della mente dal suo supporto neurofisiologico. Tuttavia, vi sono altri problemi connessi ai modelli computazionali tradizionali che sono più "interni" alle problematiche della scienza cognitiva tradizionale, nel senso che sono rilevanti anche per il paradigma simbolico classico, pur essendo di difficile soluzione al suo interno.

Innanzitutto, l'impostazione logico-linguistica della scienza cognitiva tradizionale pone inevitabilmente l'enfasi su certi tipi di fenomeni cognitivi "di alto livello" (come il linguaggio naturale e il ragionamento logico). Aspetti "di basso livello", come percezione e aspetti motori, sono difficili da trattare, e quindi emarginati. Inoltre, le attività cognitive di più alto livello vengono generalmente studiate isolatamente da quelle di livello più basso, ed è difficile prenderne in considerazione l'interazione (ad esempio tra linguaggio e ragionamento da un lato e percezione dall'altro, oppure tra pianificazione e controllo motorio). L'enfasi sul ruolo del linguaggio in ambito cognitivo rende problematico prendere in considerazione sistemi che non adoperano il linguaggio naturale, come gli animali e i bambini piccoli. I sistemi simbolici tollerano male gli errori, nel senso che se i loro *input* si discostano anche di poco da quelli previsti da chi li ha progettati, le loro prestazioni crollano drammaticamente. Questo contrasta con la capacità che hanno gli esseri umani e i sistemi intelligenti biologici di "arrangiarsi" anche in condizioni tutt'altro che ottimali. Inoltre sono poco resistenti ai danni: ogni danneggiamento locale di un sistema simbolico ha conseguenze globali disastrose. Ad esempio, l'eliminazione di alcuni assiomi da una base di conoscenza comporta che tutti i teoremi che dipendevano da quegli assiomi non possano più essere dedotti. Le prestazioni dei sistemi intelligenti reali sono certamente più flessibili al riguardo. Vi sono poi gli aspetti legati all'apprendimento e all'evoluzione di un sistema intelligente, e della sua interazione con l'ambiente, che restano per molti versi problematici nella scienza cognitiva simbolica. (Su questi temi si veda ad esempio Churchland e Sejnowski 1989; per una apologia "d'epoca" delle tesi connessioniste contrapposte alle tesi dell'IA classica si vedano Parisi 1989a e 1989b.)

(Si noti che, sul versante tecnologico, la fortuna del connessionismo è collegata anche all'ampia diffusione di calcolatori tradizionali - cioè sequenziali e basati sull'architetture di von Neumann - abbastanza potenti per simulare le reti neurali. Infatti la maggior parte della ricerca sperimentale sui modelli connessionisti viene effettuata simulando a livello *software* le reti neurali su calcolatori tradizionali. Gli anni ottanta hanno visto anche la progettazione e la commercializzazione di alcuni calcolatori ad alto parallelismo, come ad esempio la *Connection Machine* - Hillis 1986, un calcolatore ispirato ai principi della computazione neurale. Tuttavia, a distanza di più di due decenni, si può affermare che i calcolatori universali basati su un *hardware* di tipo connessionista non hanno avuto successo. Più diffusi sono invece gli esempi di *hardware dedicato* basato su architetture di tipo neurale, ossia processori neurali costruiti appositamente per affrontare compiti specifici.)

2. Unità, pesi e connessioni

Il connessionismo si propone come alternativa all'approccio tradizionale nell'intento di superare i problemi sopra menzionati. I sistemi computazionali che sono alla base dei modelli connessionisti, le cosiddette *reti neurali artificiali*, sono sistemi distribuiti ad alto parallelismo, ispirati, in senso lato, alle proprietà del sistema nervoso. Una *rete neurale* è un grafo orientato, cioè un insieme di nodi e di archi orientati che li connettono (il fatto che gli archi siano orientati significa che, nel caso generale, tali archi sono "freccie" ad una direzione). I nodi vengono detti *unità* della rete (e sono l'analogo dei neuroni), e gli archi (che costituiscono l'analogo delle sinapsi) vengono detti connessioni. La fig. 2 rappresenta un frammento di una rete neurale con sei unità e cinque connessioni. Ogni unità ha un certo numero di connessioni in ingresso e/o un certo numero di connessioni in uscita. Ciascuna unità costituisce un processore, un singolo dispositivo di calcolo che, ad ogni fase del calcolo, riceve i propri *input* attraverso le connessioni in ingresso, li elabora, e invia l'*output* alle altre unità connesse per mezzo delle sue connessioni in uscita. I dati che le unità elaborano e si scambiano sono valori di tipo numerico: di solito, si tratta di (approssimazioni di) numeri reali. In una rete tutte le unità operano in parallelo, e non esiste alcun processo di ordine "superiore", nessuna CPU che ne coordini l'attività. Il calcolo che ciascuna unità esegue è di norma

molto semplice; la potenza computazionale del sistema deriva dal grande numero delle unità e delle connessioni.

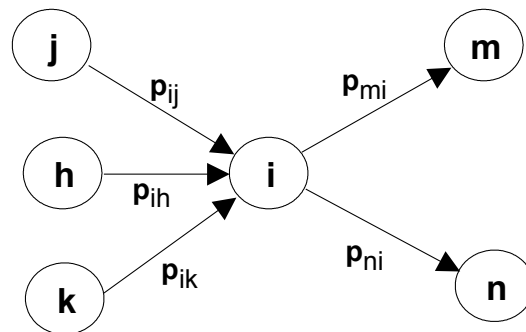


Fig. 2

Ad ogni unità di una rete connessionista sono associate una *funzione di attivazione* e una *funzione di output*. In ogni fase del calcolo, ad ogni unità è attribuito un certo *valore di attivazione*. Le fasi del calcolo normalmente evolvono a istanti di tempo discreti. In ogni istante, tutte le unità della rete calcolano il loro *output* sulla base del loro valore di attivazione e della funzione di *output* loro associata, e lo comunicano alle unità connesse. Queste, a loro volta, calcolano il loro nuovo valore di attivazione sulla base degli *input* ricevuti e della funzione di attivazione. Le funzioni di attivazione e di *output* sono spesso funzioni non lineari, ad esempio sigmoidi o funzioni di soglia. Le connessioni non sono canali di trasmissione neutri, ma modificano i dati che trasmettono moltiplicandoli per un certo valore numerico che è associato a ciascuna di esse. Tale valore è detto il *peso*, o la *forza*, della connessione (secondo l'analogia neuronale, il peso corrisponde alla forza di una sinapsi). Date due unità connesse i e j di una rete, indichiamo con p_{ij} il peso della connessione che va da i a j (fig.2).

In generale, il calcolo di una rete neurale avviene nel modo seguente. Si inizia con il fissare il valore di attivazione di alcune delle unità. Tali valori di attivazione imposti dall'esterno costituiscono una codifica dei dati del calcolo che la rete deve effettuare. Le unità destinate a ricevere questi valori iniziali vengono dette *unità di input*. Dopo di che, la rete viene lasciata evolvere spontaneamente, sino a che raggiunge uno stato di equilibrio. A questo punto, i valori di attivazione di alcune altre unità della rete, dette *unità di output*, costituiscono la codifica dei risultati del calcolo. Unità di *input* e unità di *output* vengono dette di solito *unità esterne* della rete. Tutte le altre unità, che non sono né unità di *input*, né unità di *output*, vengono dette *unità interne*, o *unità nascoste*.

Sin da ora sono evidenti le differenze con l'architettura computazionale di von Neumann: una rete neurale non è una macchina sequenziale ma parallela; inoltre, non esiste distinzione tra immagazzinamento ed elaborazione dei dati: ogni unità è contemporaneamente unità di calcolo e di memoria.

Di solito i pesi delle connessioni non sono prefissati, ma vengono fatti evolvere nel tempo sulla base di opportune regole, in modo che la rete "apprenda" a svolgere il compito che le è richiesto. Per lo più, all'inizio i pesi delle connessioni sono attribuiti in maniera casuale. Dopo di che le reti vengono "addestrate": i pesi vengono modificati sulla base delle regole di apprendimento, in maniera che i risultati approssimino via via quelli desiderati. La "conoscenza" di una rete è quindi immagazzinata nei pesi delle connessioni. I connessionisti enfatizzano questa capacità di apprendimento come una delle peculiarità delle reti neurali rispetto ai modelli computazionali tradizionali.

Tipi diversi di reti neurali si differenziano tra loro sulla base di vari fattori: la topologia della rete (cioè il modo in cui sono connesse tra loro le unità), le funzioni associate a ciascuna unità per calcolare il proprio valore di attivazione e il proprio output, e i diversi tipi di regole di

apprendimento. (Per una introduzione più approfondita agli aspetti tecnici delle varie classi di reti neurali e dei vari tipi di algoritmi di apprendimento si veda ad esempio Floreano e Mattiussi 2002).

3. Dai perceptroni alla *back propagation*

Storicamente, l'idea di fare apprendere una rete neurale artificiale per mezzo di una regola che modifichi i pesi delle connessioni risale a D.O. Hebb (1949). Il lavoro di Hebb sull'apprendimento delle reti neurali è della fine degli anni quaranta. Come si è detto, infatti, sebbene il connessionismo sia emerso prepotentemente come paradigma per lo studio della mente nel corso degli anni '80 del '900, le sue radici storiche sono più antiche, e vanno fatte risalire per lo meno alla cibernetica e alle ricerche sulle reti neurali artificiali degli anni '50 e dei primi anni '60. Significativi a questo proposito sono, tra gli altri, i lavori del già citato Hebb, di McCulloch e Pitts (1943), di Rosenblatt (1962) e dello stesso von Neumann, che, oltre ad avere elaborato il modello di calcolatore sequenziale a programma memorizzato che va sotto il suo nome, fu anche uno dei pionieri dello studio delle reti neuronali (per una panoramica su questo tipo di letteratura si veda l'antologia di Somenzi e Cordeschi 1994). In seguito, una serie di fattori di carattere tecnico e culturale fece sì che questa linea di ricerca venisse soppiantata quasi completamente dall'IA di tipo simbolico, per riemergere soltanto negli anni '80. Particolarmente significative a questo proposito furono le vicende del lavoro di Frank Rosenblatt. Rosenblatt propose un tipo di architettura neuronale estremamente semplice, da lui chiamata *perceptrone* (Rosenblatt 1962). Un perceptrone è costituito da due soli livelli di unità, un livello di unità di *input* ed un livello di unità di *output*. Nei perceptroni non esistono quindi unità nascoste. Da ciascuna delle unità di *input* parte una connessione verso ciascuna delle unità di *output*. Una volta attivate le unità di *input*, il *pattern* di attivazione sulle unità di *output* viene generato mediante un solo passo di calcolo. Un esempio di perceptrone è mostrato dalla fig. 3. In essa ogni unità di *output* rappresenta una classe di animali, mentre le unità di *input* rappresentano una serie di caratteristiche che ciascun animale può presentare o meno. Lo scopo della rete è quello di classificare gli animali sulla base delle loro caratteristiche: attivando in *input* un certo numero di unità, dovrebbe attivarsi in *output* l'unità (o eventualmente le unità) che rappresenta la classe che meglio corrisponde alle caratteristiche descritte dall'*input*. Ciò può essere ottenuto grazie ai pesi delle connessioni: ad esempio, i pesi che connettono l'unità di *input* *vola* con le unità di *output* *uccello* e *insetto* avranno un valore molto alto, mentre il peso della connessione tra *vola* e *mammifero* sarà bassissimo. In generale, il peso di un connessione sarà tanto più alto, quanto più le unità di *input* rappresentano caratteristiche tipiche della classe rappresentata dall'unità di *output*. Ovviamente, quanto più le caratteristiche date in *input* rappresentano un esemplare tipico di una classe, tanto più la risposta in *output* sarà precisa (ci sarà cioè una singola unità molto più attiva di tutte le altre). *Input* corrispondenti ad animali "atipici" (ad esempio struzzi, ornitorinchi, pipistrelli) daranno origine in *output* a *pattern* di attivazione in cui l'unità corretta è relativamente poco attiva rispetto alle altre.

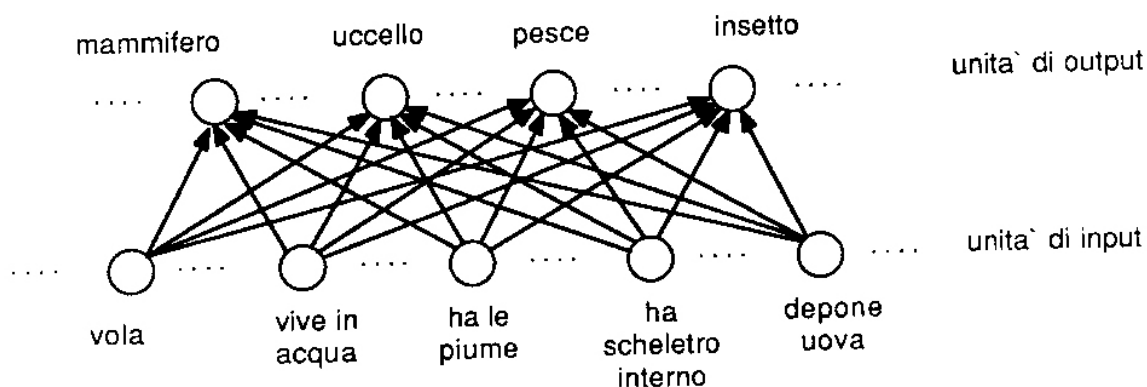


Fig. 3

In una rete di questo genere, sarebbe possibile programmare "a mano" i pesi delle connessioni, in maniera da ottenere le risposte volute. Tuttavia, in reti più complesse, tale possibilità diventa puramente teorica, e non attuabile di fatto. D'altra parte, si è detto che uno dei motivi di interesse dei modelli connessionisti consiste nel fatto che essi possono essere "addestrati" a fornire la risposta corretta mediante opportune regole di apprendimento. Per i perceptroni è disponibile una regola di addestramento particolarmente semplice detta *regola delta* (*delta rule*). Intuitivamente, il processo di apprendimento mediante la regola delta procede nel modo seguente. Si parte da una rete in cui i pesi delle connessioni sono attribuiti in maniera casuale. Dopo di che, si inizia a sottoporre in *input* alla rete una serie di esempi. All'inizio, le risposte ottenute (cioè le unità attive in *output*) saranno arbitrarie. Tuttavia, ogni volta che un *input* viene sottoposto alla rete durante la fase di addestramento, si provvede a confrontare l'*output* ottenuto con quello desiderato, e si incrementano di un'opportuna grandezza, stabilita in base alla regola delta, i pesi delle connessioni che vanno dalle unità attive dell'*input* alle unità dell'*output* che si vuole che vengano attivate. Viceversa, i pesi delle connessioni dalle unità attive dell'*input* alle altre unità di *output* vengono indeboliti¹.

Nell'esempio di fig. 3, se si presenta in *input* il *pattern* che corrisponde ad un pipistrello (attivando unità come è *coperto di pelo*, *allatta i piccoli*, *vola*), si provvederà a rafforzare i pesi delle connessioni tra queste unità e l'unità di *output* *mammifero*, e ad indebolire quelli delle connessioni con unità come *pesce* o *insetto*. Presentando altri esempi, i pesi di alcune connessioni (ad esempio quella tra è *coperto di pelo* e *mammifero*) verranno ulteriormente rafforzati, mentre altri verranno quasi sempre indeboliti (come quello tra *vola* e *mammifero*). Alla fine della fase di addestramento, i pesi delle connessioni rifletteranno le caratteristiche tipiche del campione di esempi utilizzato.

L'estrema semplicità dei perceptroni, sia in termini di struttura che di apprendimento, ha come controparte negativa la loro scarsa potenza computazionale. Minsky e Papert (1969) dimostrarono che esistono funzioni che sono banalmente computabili da una macchina di Turing (e

¹ La formula che stabilisce l'incremento dei pesi delle connessioni è del tipo seguente:

$$\Delta_{p_{ij}} = \eta \delta_j a_i$$

Per ogni unità di *input* i e per ogni unità di *output* j , $\Delta_{p_{ij}}$ è il valore di cui deve essere incrementato il peso della connessione tra i e j . δ_j è l'errore dell'unità j , cioè la differenza tra il valore di attivazione desiderato e quello ottenuto. a_i è il valore di attivazione dell'unità di *input* i . Infine, η è una costante che determina la velocità dell'apprendimento. Quanto più il valore di attivazione desiderato per j è alto rispetto al valore ottenuto, tanto più è grande l'incremento del peso. Se, viceversa, il valore desiderato è minore di quello ottenuto, l'incremento è negativo (in quanto $\delta_j < 0$). Infine, per quelle connessioni in uscita da quelle unità di *input* i che non sono state attivate (per cui cioè $a_i = 0$), l'incremento è nullo.

quindi da ogni calcolatore di von Neumann), ma che nessun perceptrone è in grado di calcolare. Ad esempio, non si può costruire un perceptrone che calcoli l'or esclusivo, cioè l'operatore verofunzionale *xor* la cui tavola di verità è mostrata dalla tabella 1a. Si vorrebbe un perceptrone con due unità di *input* e una unità di *output*, ciascuna delle quali possa assumere solo due valori di attivazione, 1 e 0 (che corrispondono rispettivamente al vero e al falso). Il comportamento della rete dovrebbe essere quello specificato dalla tabella 1b. Si può tuttavia dimostrare che un perceptrone che calcoli questa funzione non esiste.

<i>A</i>	<i>B</i>	<i>A xor B</i>
V	V	F
V	F	V
F	V	V
F	F	F

tab. 1a

<i>Pattern in input</i>	<i>Pattern in output</i>
11	0
10	1
01	1
00	0

tab. 1b

Tab. 1

Per aumentare la potenza computazionale dei perceptroni sarebbe sufficiente aggiungere uno o più livelli di unità nascoste tra lo strato di unità di *input* e lo strato di unità di *output*. Le reti di questo tipo vengono dette reti *multistrato*. La fig. 4 mostra una rete multistrato con un solo strato di unità nascoste. Tuttavia, per reti di questo genere la semplice versione della regola delta che abbiamo visto sopra non è più utilizzabile, in quanto essa non dice nulla su come aggiornare il peso di connessioni che non colleghino direttamente unità di *input* con unità di *output*. Quando uscì il libro di Minsky e Papert non si disponeva di alcuna via per superare questo tipo di difficoltà in maniera sufficientemente generale. Il lavoro di Minsky e Papert ebbe quindi un ruolo fondamentale nel far sì che le ricerche sulle reti neurali artificiali fossero abbandonate in favore del paradigma simbolico per lo studio computazionale della mente.

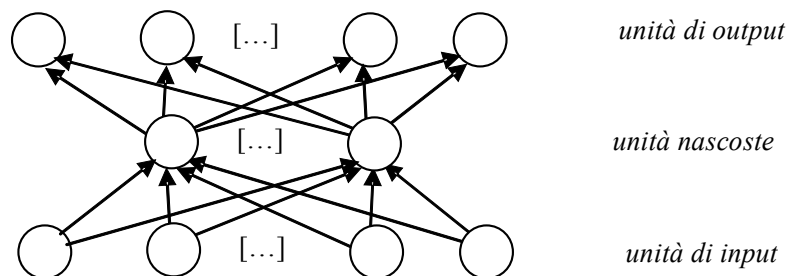


Fig. 4

Una soluzione soddisfacente a questi problemi fu trovata solo in anni più recenti. Le reti come quelle della fig. 4 sono dette reti di tipo *feed forward*: il flusso di attivazione procede sempre *in avanti* a partire dalle unità di *input* verso le unità di *output*. Per questo tipo di reti è stato elaborato un algoritmo di apprendimento che generalizza la regola delta, in maniera tale che, una volta confrontato l'*output* ottenuto con quello atteso, sia possibile la "propagazione all'indietro" (*back propagation*) dell'errore mediante la modifica dei pesi delle connessioni interne (Rumelhart, Hinton e Williams 1986). L'individuazione di questa "regola delta generalizzata" fu uno dei fattori determinanti per la rinascita dell'interesse per le reti neurali. Essa consentiva infatti di superare il problema teorico individuato da Minsky e Papert per i perceptroni, e metteva al tempo stesso a disposizione un modello di calcolo neuronale facilmente utilizzabile, sia a livello di ricerca che applicativo (le reti *feed forward* con *back propagation* sono infatti il tipo di rete neurale più diffuso a livello applicativo, e uno tra i più semplici da utilizzare).

I ricercatori che misero a punto la regola delta generalizzata facevano parte del cosiddetto gruppo PDP (PDP sta per *Parallel Distributed Processing*: elaborazione distribuita in parallelo), un gruppo di ricerca che ebbe un ruolo determinante, a livello di elaborazione teorica, nel processo di rivalutazione e di diffusione delle reti neurali. I due volumi di saggi che furono pubblicati a cura del gruppo PDP (Rumelhart, McClelland e il gruppo di ricerca PDP 1986; McClelland, Rumelhart e il gruppo di ricerca PDP 1986) possono essere considerati uno dei principali manifesti teorici del connessionismo, e uno dei fattori che più contribuirono a diffondere le idee connessioniste all'interno della comunità scientifica.

4. I Jet e gli Shark

Oltre alle reti *feedforward*, l'altra grande classe di reti neurali è costituita dalle *reti ricorrenti*, caratterizzate dal fatto che le connessioni possono formare dei cicli, come negli esempi di fig. 5.

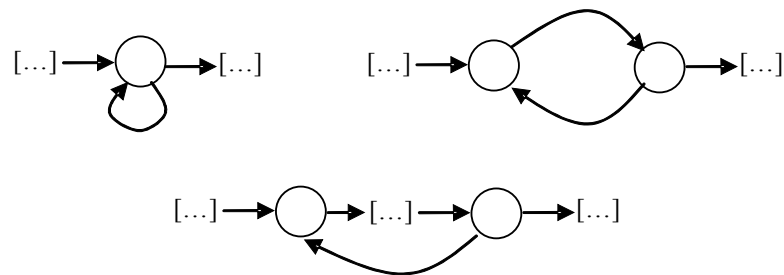


Fig. 5

In generale, in una rete ricorrente la presenza di cicli comporta che il processo di calcolo possa dare luogo a un'evoluzione dinamica più complessa rispetto alle reti *feedforward*, in cui l'informazione fluisce in un'unica direzione, dalle unità di input verso le unità di output. Un tipo particolare di reti ricorrenti sono le reti *simmetriche*, nelle quali per ogni connessione che va da una data unità u_1 a un'unità u_2 , c'è connessione simmetrica con lo stesso peso che va da u_2 a u_1 . Esempi di questa categoria sono le reti di Hopfield e le cosiddette macchine di Boltzmann. In questo tipo di reti il calcolo avviene assegnando i valori iniziali di un certo numero di unità esterne, e lasciando evolvere la rete. A seconda dei valori di partenza assegnati alle unità, la rete potrà raggiungere stati diversi di equilibrio che svolgono la funzione di attrattori.

Anche in questo caso si tratta di individuare regole generali di apprendimento per programmare i pesi delle connessioni. Diverse tecniche formali sono state proposte a questo scopo, in vari casi basate su analogie e concetti provenienti dalla termodinamica: si parte da una configurazione "caotica" casuale dei pesi, e l'apprendimento avviene facendo diminuire la "temperatura" della rete, facendo cioè in modo che la rete passi ad una configurazione più stabile ed ordinata. Si tratta di evitare che questo processo si blocchi in qualche *minimo locale*, cioè in una configurazione che corrisponde a uno stato di equilibrio non ottimale. Le *macchine di Boltzmann* (Hinton e Sejnowski 1986), basate sulla tecnica della *temperatura simulata* (*simulated annealing*), sono una classe di reti neurali elaborate per risolvere questo tipo di problemi. In questa tradizione si colloca anche la *teoria dell'armonia* (*harmony theory*) di Smolensky (1986). Per questi aspetti si veda ancora Floreano e Mattiussi 2002.

Vediamo un esempio molto semplice di rete neurale ricorrente sviluppato a fini espositivi da J. McClelland (1981), e riportato anche in (McClelland, Rumelhart e Hinton, 1986). Esso ci consentirà di comprendere alcune interessanti proprietà dei modelli connessionisti. La rete rappresenta un certo numero di informazioni relative agli appartenenti a due bande di malviventi, i Jets e gli Sharks (vedi tabella 2 - i nomi delle bande sono quelli di *West Side Story*). La fig. 6 mostra parte della rete neurale adoperata a questo scopo (in questo esempio non ci si pone il problema dell'apprendimento, ma si assume che alle connessioni siano già stati assegnati i pesi

appropriati). Le unità al centro sono le unità nascoste. Ciascuna di esse rappresenta uno dei personaggi della tab. 2. Gli altri gruppi di unità rappresentano le proprietà di cui questi personaggi godono (età, occupazione, banda di appartenenza, etc.). (Si noti che il nome viene rappresentato come una proprietà analoga alle altre.) Queste ultime sono le unità esterne della rete, che possono essere utilizzate indifferentemente sia come unità di *input* che come unità di *output*. Le unità interne sono collegate per mezzo di connessioni bidirezionali eccitatorie (cioè con peso positivo) alle unità che rappresentano le proprietà di cui gode l'individuo rappresentato; ad esempio, l'unità che rappresenta Lance è collegata al nome *Lance*, all'attività *ladro*, alla banda *Jets*, e così via (una connessione bidirezionale corrisponde a due connessioni unidirezionali di direzione opposta e con lo stesso peso). Inoltre, all'interno di ciascun gruppo, le unità sono collegate tra loro per mezzo di connessioni bidirezionali inibitorie (cioè con peso negativo). Tali connessioni, che non sono mostrate nella figura, fanno sì che in ogni gruppo tenda ad essere attiva una sola unità alla volta.

<i>Nome</i>	<i>Banda</i>	<i>Età</i>	<i>Tit. studio</i>	<i>Stato civile</i>	<i>Attività</i>
Art	Jets	40	LM	Scapolo	Spacciatore
Al	Jets	30	LM	Sposato	Ladro
Sam	Jets	20	U	Scapolo	Ricettatore
Clyde	Jets	40	LM	Scapolo	Ricettatore
Mike	Jets	30	LM	Scapolo	Ricettatore
Jim	Jets	20	LM	Divorziato	Ladro
Greg	Jets	20	D	Sposato	Spacciatore
John	Jets	20	LM	Sposato	Ladro
Doug	Jets	30	D	Scapolo	Ricettatore
Lance	Jets	20	LM	Sposato	Ladro
George	Jets	20	LM	Divorziato	Ladro
Pete	Jets	20	D	Scapolo	Ricettatore
Fred	Jets	20	D	Scapolo	Spacciatore
Gene	Jets	20	U	Scapolo	Spacciatore
Ralph	Jets	30	LM	Scapolo	Spacciatore
Phil	Sharks	30	U	Sposato	Spacciatore
Ike	Sharks	30	LM	Scapolo	Ricettatore
Nick	Sharks	30	D	Scapolo	Spacciatore
Don	Sharks	30	U	Sposato	Ladro
Ned	Sharks	30	U	Sposato	Ricettatore
Karl	Sharks	40	D	Sposato	Ricettatore
Ken	Sharks	20	D	Scapolo	Ladro
Earl	Sharks	40	D	Sposato	Ladro
Rick	Sharks	40	D	Divorziato	Ladro
Ol	Sharks	30	U	Sposato	Spacciatore
Neal	Sharks	30	D	Scapolo	Ricettatore
Dave	Sharks	30	D	Divorziato	Spacciatore

LM = licenza media; D = diploma; U = Università

Tab. 2

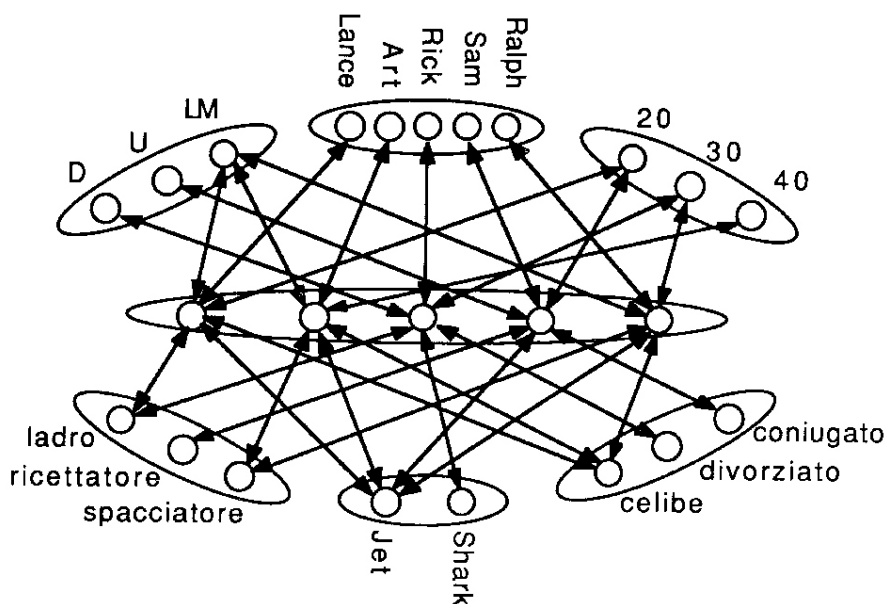


Fig. 6

A questo punto, se si vogliono informazioni su qualcuno dei personaggi rappresentati, ad esempio su Lance, basta attivare in *input* l'unità che corrisponde al suo nome. Questo fa sì che venga attivata l'unità nascosta corrispondente, che di conseguenza causa l'attivazione in *output* di tutte le unità che rappresentano le sue proprietà. Analogamente, se voglio interrogare la rete per sapere il nome di qualcuno che sia uno spacciatore quarantenne, attivo in *input* le due unità corrispondenti. Questo fa sì che vengano parzialmente attivate diverse unità nascoste (tutte quelle connesse alle unità attivate in *input*). Tuttavia, quella che rappresenta Art sopravvarrà sulle altre. Egli infatti è l'unico a godere di entrambe le proprietà, e l'unità che lo rappresenta riceve quindi *input* da due diverse connessioni. Di conseguenza, il nome di Art è l'unità che riceverà maggiore attivazione in *output*.

Ovviamente, risultati di questo genere sarebbero stati possibili in qualunque base di dati tradizionale. La rete neurale però offre anche altre interessanti prestazioni. Innanzi tutto, dà la possibilità di trattare informazione incompleta o parzialmente erronea. Supponiamo di interrogare la rete sull'esistenza di uno spacciatore scapolo sulla quarantina che sia uno Shark con la licenza media. Nella base di conoscenza non esiste nessuno che soddisfi contemporaneamente tutte queste caratteristiche. Tuttavia, la rete fornisce comunque una risposta, che corrisponde all'individuo che meglio si adegua alla descrizione: nell'esempio specifico l'unità che riceve un grado di attivazione maggiore è quella che rappresenta Art, che corrisponde alla descrizione eccetto per il fatto di essere un Jet anziché uno Shark. Ovviamente, il grado di attivazione di Art sarebbe stato maggiore se tutte le proprietà fossero state corrette. In generale, il grado di attivazione di un'unità in *output* è tanto maggiore, quanto più il sistema è "sicuro" della risposta. Tuttavia, il fatto che alcuni dati siano errati non impedisce alla rete di "avanzare un'ipotesi" o di "fornire un suggerimento" sulla base dell'evidenza di cui dispone.

Inoltre, la rete è in grado di completare le proprie risposte con valori assegnati "per *default*" sulla base di analogie induttive. Supponiamo ad esempio di non disporre dell'informazione che Lance è un ladro, e che manchi quindi la corrispondente connessione nella rete. Tuttavia, nella tabella, ci sono parecchi individui che condividono molte proprietà con Lance, e che, in quanto a occupazione, sono tutti dei ladri. Quando le proprietà di Lance sono attive, le unità nascoste che li rappresentano ne vengono a loro volta attivate in modo parziale, e contribuiscono ad attivare la proprietà *ladro* (seppure con un grado di attivazione minore che se esistesse una connessione diretta con l'unità che rappresenta Lance). Anche qui si tratta di "suggerimento" che la rete fornisce sulla

base dell'informazione disponibile. Questo grazie a una sorta di euristica implicita nel modello, per cui si induce una proprietà ignota di un individuo sulla base delle proprietà di altri individui simili.

Infine, il modello è in grado, generalizzando i dati di cui dispone, di formare descrizioni di *prototipi* che corrispondono alle proprietà rappresentate. Supponiamo ad esempio di voler sapere quali sono le proprietà del tipico appartenente alla banda dei Jet. Per fare ciò si può attivare in *input* l'unità *Jet* della rete. Essa causerà l'attivazione delle unità nascoste che rappresentano i vari membri della banda, le quali, a loro volta, attiveranno parzialmente le unità che rappresentano le proprietà di ciascuno di essi. La maggioranza dei Jet sono scapoli, sono sulla ventina e hanno la licenza media (sebbene nessuno di essi goda contemporaneamente di tutte queste proprietà - cfr. tab. 2). Quindi le unità che rappresentano queste caratteristiche saranno le più attive dei rispettivi gruppi. In questo modo la rete ha individuato una sorta di "prototipo" di membro dei Jet, anche se nessun individuo specifico vi corrisponde esattamente. Si noti che tutte queste prestazioni non sono state ottenute inserendo meccanismi espliciti (come ad esempio regole per il trattamento degli errori), che è quanto sarebbe stato necessario in una rappresentazione tipo simbolico. Esse derivano piuttosto dalle proprietà stesse del modello, e dal modo in cui l'informazione è rappresentata in esso. Ad esempio, la rappresentazione di un prototipo, che nei sistemi tradizionali di IA è ricondotta a una struttura simbolica esplicita (ad esempio i *frame* di Marvin Minsky), in un sistema connessionista di questo tipo emerge dalle proprietà computazionali globali della rete.

Si noti che nella rete neurale di questo esempio non esiste una distinzione netta tra unità di *input* e unità di *output*, ma tutte le unità esterne possono essere utilizzate indifferentemente per fornire dati alla rete e per estrarne i risultati. Il calcolo avviene presentando in *input* un *pattern* di attivazione sulle unità esterne, e l'*output* della rete consiste in un completamento di tale *pattern*.

5. Connessionismo e neuroscienze

Merita una precisazione il rapporto che sussiste tra connessionismo e neuroscienze. Non sempre le reti neurali sono state impiegate per elaborare modelli dettagliati della struttura anatomica e funzionale del sistema nervoso. Vi è stato chi, come Churchland e Sejnowski (1992), ha preso sul serio i rapporti tra connessionismo e neuroscienze. Più spesso tuttavia i modelli connessionisti fanno riferimento al sistema nervoso solo come a una fonte di ispirazione, nell'ipotesi che un'architettura computazionale ad alto parallelismo risulti più adeguata a spiegare importanti aspetti dei fenomeni cognitivi, senza che tuttavia si vada alla ricerca di riscontri empirici puntuali nei dati relativi al sistema nervoso. Le reti neurali artificiali sarebbero più adatte a catturare la flessibilità che caratterizza le prestazioni dei sistemi cognitivi reali, ad esempio rispetto alle capacità di apprendimento di cui si è detto sopra. Un altro esempio riguarda compiti che dipendono dal soddisfacimento in parallelo di vincoli multipli: in fenomeni come il controllo motorio un agente deve tenere conto contemporaneamente di molteplici fattori, quali la postura del proprio corpo, gli obiettivi da raggiungere, gli ostacoli da evitare, e così via. Oppure si considerino compiti in cui è essenziale tenere conto del contesto: nella fig. 7 ogni lettera ammette due letture diverse (la prima potrebbe essere una P o una R, eccetera); eppure vi è una interpretazione (quella che corrisponde alla parola RED) che emerge in maniera spontanea e immediata. Ciò dipende dal contesto formato dalla collocazione reciproca delle lettere: delle otto interpretazioni possibili, RED è l'unica che corrisponde a una parola dell'inglese. In casi del genere l'elaborazione distribuita in parallelo dei modelli connessionisti risulta particolarmente efficace (si veda al proposito il primo capitolo di Rumelhart e McClelland 1986, dove sono descritti alcuni semplici modelli per risolvere problemi di questo tipo).



Fig. 7

Uno slogan spesso ribadito dai teorici del connessionismo è che le reti neurali, al contrario dei modelli delle scienze cognitive classiche, sarebbero in grado di cogliere la *microstruttura* dei processi cognitivi: i modelli connessionisti starebbero ai modelli classici come le spiegazioni microfisiche della meccanica quantistica stanno alle descrizioni macroscopiche della fisica classica. Come la fisica classica, le scienze cognitive simboliche sarebbero in grado di fornire al più spiegazioni approssimate, che valgono solo come casi limite, e che non colgono la natura profonda dei fenomeni studiati. Per ottenere una spiegazione precisa e dettagliata dei fenomeni cognitivi si deve passare al livello di descrizione proprio dei modelli connessionisti. In questo senso, i modelli connessionisti sono detti talvolta *subsimbolici* (Smolensky 1988), in contrapposizione ai modelli simbolici delle scienze cognitive classiche.

6. Le rappresentazioni connessioniste: il problema della composizionalità

L'adozione da parte dei connessionisti di architetture più vicine alla struttura del sistema nervoso è associata all'impiego di rappresentazioni diverse rispetto a quelle delle scienze cognitive classiche. Queste adottano formalismi di tipo logico-linguistico, in cui le computazioni vengono assimilate a processi inferenziali. Nelle reti neurali invece le informazioni sono rappresentate tramite i valori numerici associati alle connessioni e alle unità (rispettivamente i pesi e i valori di attivazione), e le computazioni hanno natura statistica, in quando consistono nell'effetto concomitante di un gran numero di semplici operazioni numeriche che avvengono in parallelo. Ciò viene considerato un punto di forza dai fautori del connessionismo, ma ha fornito anche lo spunto a una delle critiche più influenti ai modelli connessionisti. Tale critica è incentrata sulla nozione di *composizionalità* delle rappresentazioni.

In generale, in un sistema di rappresentazioni di tipo composizionale si distingue tra un insieme di *simboli primitivi*, o *atomici*, e un insieme di *simboli complessi*. Questi ultimi vengono generati a partire dai simboli primitivi attraverso l'applicazione di opportune regole sintattiche di composizione (di norma, a partire da un insieme finito di simboli primitivi si può generare un insieme potenzialmente illimitato di simboli complessi). Le lingue naturali sono l'esempio paradigmatico di sistemi composizionali: i simboli primitivi corrispondono agli elementi del lessico (o, per essere più precisi, ai morfemi), e i simboli complessi includono l'insieme (potenzialmente infinito) di tutti gli enunciati.

Nei sistemi composizionali il significato di un simbolo complesso è funzione della struttura sintattica del simbolo stesso e del significato dei simboli primitivi che vi compaiono. Vale a dire, il significato di un simbolo complesso può essere determinato a partire dal significato dei suoi componenti per mezzo di regole semantiche che "operano in parallelo" alle regole di composizione sintattica. In ciò consiste il *principio di composizionalità del significato*, che era stato identificato da Frege come una delle caratteristiche principali delle lingue umane.

Un semplice esempio di composizionalità è dato dal comportamento semantico dei connettivi proposizionali. Si consideri ad esempio la congiunzione. Il valore di verità di un enunciato complesso del tipo $a \wedge b$ dipende composizionalmente dai valori di verità degli enunciati componenti a e b : $a \wedge b$ è vera se e soltanto se sia a che b sono entrambe vere. Da questa definizione composizionale della semantica di $a \wedge b$ consegue anche la validità delle regole logiche di inferenza che riguardano la congiunzione, come ad esempio:

- (i) *regola di eliminazione della congiunzione*: da una premessa con la forma $a \wedge b$ seguono logicamente sia a , sia b ;
- (ii) *regola di introduzione della congiunzione*: da due premesse a e b segue logicamente $a \wedge b$.

Nelle scienze cognitive classiche si assume che le rappresentazioni mentali abbiano natura compositiva. Una delle formulazioni più esplicite di questo assunto è dovuta a Jerry Fodor e Zenon Pylyshyn (1988), secondo i quali la composizionalità costituirebbe un requisito irrinunciabile che le rappresentazioni mentali devono soddisfare al fine di spiegare alcuni fenomeni cognitivi fondamentali, in primo luogo il carattere *generativo* della cognizione: la mente umana, comunque la si intenda, è presumibilmente finita, eppure ciascuno di noi è in grado di formulare e comprendere pensieri sempre nuovi, con i quali non è mai venuto in contatto prima. Ad esempio, siamo in grado di rappresentarci mentalmente il contenuto di *un gatto rosa cantò appassionatamente all'assemblea di condominio* anche se incontriamo questo enunciato per la prima volta. Alla composizionalità è legata anche la *sistematicità* delle prestazioni cognitive: sembra evidente che la capacità di concepire certi contenuti sia collegata in modo sistematico alla capacità di concepirne altri. Ad esempio, se un soggetto comprende *il gatto insegue un topo*, allora egli è in grado di comprendere anche *il topo insegue un gatto*, e ciò in virtù del fatto che i due enunciati hanno una forma sintattica simile. Se ne può concludere che la capacità di comprendere contenuti proposizionali dipende dalla struttura compositiva dei contenuti stessi. Tutto ciò può essere spiegato senza difficoltà se si assume che le rappresentazioni mentali abbiano una forma simile a un linguaggio compositivo.

Fodor e Pylyshyn ritengono che tutto questo andrebbe irrimediabilmente perduto nei modelli connessionisti, in quanto le rappresentazioni connessioniste non sono dotate di composizionalità. Consideriamo la congiunzione. E' facile costruire una rete neurale come quella di fig. 8, dove due unità corrispondono ai due congiunti a e b , e una terza unità viene utilizzata per rappresentare la loro congiunzione $a \wedge b$. Supponiamo che le unità possano assumere solo due valori di attivazione, 1 (ossia attivo, che facciamo corrispondere al vero) e 0 (ossia non attivo, che facciamo corrispondere al falso). I pesi delle connessioni si possono stabilire in maniera tale che l'unità $a \wedge b$ diventi attiva quando sono attive contemporaneamente sia a , sia b ; e che inoltre, se $a \wedge b$ è attiva, allora vengano attivate sia a , sia b .

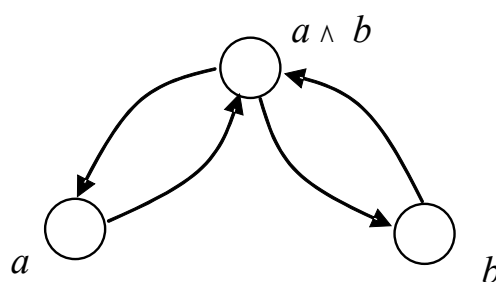


Fig. 8

A prima vista questa rete sembrerebbe esibire lo stesso comportamento delle summenzionate regole di inferenza (i) e (ii). Tuttavia esistono importanti differenze. Nei modelli simbolici la struttura sintattica dei simboli ha un ruolo importante nel determinare il comportamento del sistema. Ad esempio, le regole logiche operano sulle formule in base alla struttura sintattica di queste ultime. Nulla di tutto ciò accade nella rete della fig. 8. Il nome " $a \wedge b$ " assegnato all'unità in alto è una semplice etichetta, che potrebbe essere sostituita da qualsiasi altra sequenza di simboli senza modificare il comportamento della rete. La rappresentazione di $a \wedge b$ non è dunque ottenuta per composizione a partire dalle rappresentazioni di a e di b ; tutte le rappresentazioni sono, in un certo senso, atomiche. La differenza tra la rete di fig. 8 e le regole logiche (i) e (ii) è ulteriormente evidente se si considera che queste ultime si applicano a formule qualsiasi, sia primitive, sia

complesse. Ad esempio, la stessa regola (ii) che consente di derivare $a \wedge b$ a partire dalle premesse a e b consentirebbe anche di derivare $(p \vee q) \wedge \neg p$ a partire dalle premesse $(p \vee q)$ e $\neg p$. Invece nella fig. 8 a e b rappresentano enunciati specifici, e, per prendere in considerazione altri casi, è necessario costruire nuove porzioni di rete introducendo nuove unità e nuove connessioni, del tutto indipendenti dalle precedenti. In sintesi dunque, secondo Fodor e Pylyshyn i modelli neurali non consentirebbe di rendere conto della generatività e della sistematicità delle abilità cognitive umane in quanto non sarebbero sensibili alla struttura compositiva delle rappresentazioni.

I problemi dovuti alla mancanza di composizionalità emergono ogni qual volta si intendano realizzare forme di inferenza linguistico-simbolica con tecniche connessioniste. Particolarmente complesso risulta il trattamento delle variabili, ad esempio nella realizzazione di sistemi a regole. Adottando una tecnica in parte simile a quella della rete di fig. 8, si possono costruire reti connessioniste che realizzano sistemi di regole proposizionali del tipo *se p allora q*: l'antecedente e il conseguente di ciascuna regola vengono rappresentati come unità della rete, e i nessi di implicazione del tipo "se ... allora ..." vengono rappresentati mediante connessioni i cui pesi sono definiti in maniera tale che se l'antecedente di una regola è attivo, allora viene attivato anche il conseguente.

Tuttavia, questa tecnica non può essere estesa a regole della logica dei predicati. Essa è infatti completamente inadeguata per rappresentare variabili quantificate, e per realizzare il meccanismo dell'istanziamento (*binding*) delle variabili, che è invece garantito dalla struttura compositiva della logica del primo ordine. Un'inferenza come:

$$\frac{\forall x (\exists y R(x,y) \rightarrow P(x)) \quad R(a, b)}{P(a)}$$

che è banalmente implementabile in un sistema simbolico (ad esempio in Prolog²) risulta fonte di grandi difficoltà per i sistemi connessionisti.

Una proposta di parte connessionista per rispondere alle critiche di Fodor e Pylyshyn è stata avanzata da Paul Smolensky (1988). Secondo Smolensky, la mancanza di composizionalità affliggerebbe esclusivamente i modelli connessionisti *locali*, nei quali cioè le rappresentazioni vengono fatte corrispondere a singole unità della rete, come appunto nella fig. 8 (sono locali anche le rappresentazioni degli esempi dei parr. 3 e 4). Diverso sarebbe il caso dei modelli connessionisti *distribuiti*, in cui le rappresentazioni emergono come dei *pattern* di attivazione che risultano appunto distribuiti su di un intero gruppo di unità. Anche questa caratteristica sarebbe ispirata alle proprietà del sistema nervoso, nell'ipotesi che la conoscenza nel cervello sia rappresentata in maniera distribuita su *pattern* di attivazione neurali³. Un esempio di rappresentazione distribuita è costituito dalla rappresentazione connessionista di vari tipi di stanza descritta in (Rumelhart, Smolensky *et al.* 1986). In questa rete neurale ogni nodo rappresenta una proprietà che una stanza può soddisfare o meno (ad esempio, il fatto che ci sia il telefono, che ci sia una scrivania oppure una vasca da bagno, e così via). I diversi tipi di stanza (cucina, studio, stanza da bagno, eccetera) non sono rappresentati localmente da singole unità, ma corrispondono a *pattern* di attivazione in cui sono attive le unità che rappresentano le proprietà di cui ciascun tipo di stanza gode. La stanza da bagno, ad esempio, sarà rappresentata da un *pattern* in cui è attivo il nodo *vasca da bagno*, mentre non è attivo il nodo *scrivania*, nel *pattern* che rappresenta lo studio sarà molto attivo il nodo *scrivania*, mentre il nodo *letto* avrà un grado di attivazione molto basso, e così via.

A mio avviso, uno degli esempi più chiari di cosa si intenda per rappresentazione distribuita è offerto dalle unità nascoste di una rete *feed forward* multistrato: la conoscenza della rete è "rappresentata" in esse, ma, di norma, nessuna informazione particolare è localizzata in un'unità

² Il Prolog è un linguaggio di programmazione basato sulla logica dei predicati del primo ordine.

³ Anche se è noto che vi sono molti tipi di neuroni (biologici) che costituiscono delle rappresentazioni locali. Ne sono esempio vari tipi di neuroni presenti nelle diverse aree della corteccia visiva, oppure i celebri neuroni specchio.

specifica. Si consideri un'ipotetica rete multistrato per il riconoscimento di caratteri dell'alfabeto (fig. 9). Le lettere sono presentate in *input* sotto forma di matrici di punti: ogni unità di *input* della rete è associata ad un punto della matrice. Le unità di *output* rappresentano localmente le varie lettere. Una volta che la rete sia stata addestrata a riconoscere le lettere che le vengono presentate, la conoscenza sulla struttura grafica delle lettere è rappresentata nelle unità nascoste e nelle relative connessioni. Si tratta però di una rappresentazione distribuita: nessuna informazione specifica è rappresentata localmente nelle singole unità: di norma non vi sarà un'unità che si attiva quando compare un determinato tratto grafico. Inoltre, "come è fatta" la A è rappresentato nelle stesse unità e connessioni che rappresentano "come sono fatte" la B e tutte le altre lettere. Solo la rete nel suo complesso è in grado di esibire la conoscenza acquisita.

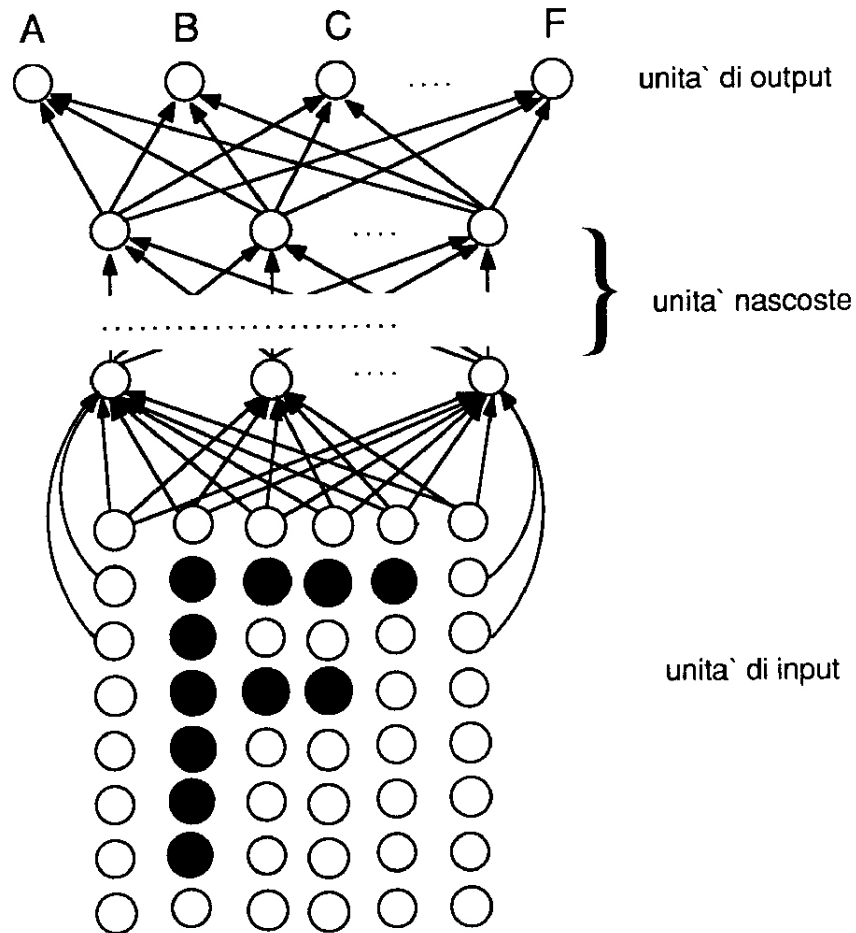


Fig. 9

Un ulteriore esempio è stato proposto da Smolensky in relazione al tema della composizionalità delle rappresentazioni connessioniste. Si supponga di voler rappresentare il concetto *tazza di caffè*. In una rete locale tale concetto corrisponderebbe a un'unità specifica. In una rappresentazione distribuita invece le unità di una rete verrebbero utilizzate per rappresentare un insieme di *microcaratteristiche* (*microfeature*), ossia tratti elementari che, nel nostro esempio, potrebbero essere *contenitore*, *liquido caldo*, *vetro a contatto con legno*, e così via (fig. 10). Il concetto *tazza di caffè* non corrisponde ad alcuna unità della rete, ma è rappresentato come una distribuzione dei valori di attivazione di queste unità. Ad esempio, nel *pattern* che rappresenta *tazza di caffè* sarà attivo il tratto *liquido caldo*, mentre non sarà attivo *vetro a contatto con legno*. Nella fig. 10 la rappresentazione di *tazza di caffè* corrisponde alla colonna *a*; le unità attive sono visualizzate come pallini neri. Nei modelli connessionisti distribuiti la struttura composizionale (che è assente al livello delle singole unità) emergerebbe al livello delle rappresentazioni distribuite

su *pattern* di attivazione. L'esempio di fig. 10 presenterebbe già una certa struttura compositiva: la rappresentazione di *tazza di caffè* potrebbe essere vista come il risultato della composizione di due *pattern*, che rappresentano rispettivamente i concetti *tazza vuota* e *caffè* (colonne *b* e *c*). Si tratta ovviamente di un esempio elementare; la compositività è un fenomeno ben più complesso e di portata ben più vasta. Tuttavia, secondo Smolensky, questa strada avrebbe potuto portare a una risposta generale al problema.

Unità			Microcaratteristiche
<i>a</i>	<i>b</i>	<i>c</i>	
●	●	○	<i>Contenitore</i>
●	○	●	<i>liquido caldo</i>
○	○	○	<i>vetro a contatto con legno</i>
●	●	○	<i>superficie curva di porcellana</i>
●	○	●	<i>aroma intenso</i>
●	○	●	<i>liquido marrone a contatto con porcellana</i>
○	○	○	<i>oggetto metallico oblungo</i>
●	●	○	<i>manico di dimensione adatta a un dito</i>
●	○	●	<i>liquido con superfici curve</i>

Fig. 10

Sono stati sviluppati vari tentativi di realizzare sistemi compositivi con tecniche connessioniste distribuite, più o meno lungo la traccia della proposta di Smolensky. Tuttavia, a distanza di due decenni, si può affermare che i risultati non sono stati all'altezza delle aspettative: i sistemi ottenuti sono molto complicati, e risultano comunque limitati rispetto ai corrispondenti sistemi simbolici.

Un'altra linea di ricerca tesa a conciliare rappresentazioni compositivi e reti neurali si è orientata allo sviluppo di *sistemi ibridi*, che combinassero componenti connessioniste con componenti basate sui formalismi dell'intelligenza artificiale classica. Anche in questo caso sono state proposte varie soluzioni *ad hoc* per problemi specifici, senza che si riuscissero a individuare strumenti teorici e spiegazioni di portata generale. In sintesi, è lecito affermare che il problema di far convivere compositività e modelli connessionisti non ha a tutt'oggi trovato una risposta soddisfacente.

7. Regole esplicite, innatismo, apprendimento

Un altro vivace dibattito tra sostenitori e detrattori del connessionismo ha avuto come oggetto il linguaggio naturale. Nell'ambito delle scienze cognitive classiche la posizione sul linguaggio più autorevole è quella chomskiana. Tra gli assunti alla base della linguistica di Chomsky vi è l'ipotesi che la conoscenza del linguaggio sia rappresentata nella mente sotto la forma di regole esplicite. Inoltre le posizioni chomskiane sono fortemente innatiste: la facoltà mentale del linguaggio (la grammatica universale) è innata e uguale per tutti gli esseri umani; l'apprendimento di una lingua specifica si limita alla fissazione di alcuni parametri della grammatica universale.

Ponendo una forte enfasi sull'apprendimento, le posizioni dei connessionisti sono per lo più lontane dall'innatismo. Inoltre i connessionisti preferiscono spiegare le regolarità nel comportamento cognitivo nei termini dell'effetto emergente di un grande numero di interazioni subsimboliche piuttosto che nei termini di rappresentazioni mentali macroscopiche ed esplicite (ad

esempio, regole). Il connessionismo è quindi difficilmente conciliabile con le tesi chomskiane sul linguaggio.

Questa contrapposizione è sfociata in un vivace dibattito sull'apprendimento del passato remoto (*past tense*) dei verbi inglesi. In sintesi, l'apprendimento del *past tense* nei bambini segue delle regolarità precise, ed è distinto in tre fasi (segue cioè la cosiddetta "curva a U"): nella prima fase i bambini imparano alcuni esempi sparsi, regolari e irregolari, di passato remoto, trattandoli sostanzialmente come casi distinti; nella seconda fase imparano che il passato remoto si ottiene per mezzo del suffisso *-ed*, e trattano i verbi irregolari come se fossero regolari (dicendo ad esempio *goed* invece di *went*); in questa fase dimenticano le forme irregolari che nella prima fase usavano in modo corretto; nella terza fase infine imparano a trattare in maniera corretta sia i verbi regolari che quelli irregolari. Per la linguistica chomskiana si tratta di un esempio paradigmatico che può essere spiegato facendo appello a regole mentali esplicite: nella seconda fase i bambini apprenderebbero una regola per la formazione del passato, e questo causerebbe la regolarizzazione di tutti i verbi; la terza fase corrisponderebbe alla memorizzazione di un elenco di eccezioni, che verrebbero trattate a parte.

David Rumelhart e James McClelland (1986, cap. 18 – assente nella trad. it.) hanno proposto un modello connessionista dell'apprendimento del *past tense* in cui le varie fasi sopra descritte emergono dal funzionamento di una rete neurale che non impiega alcuna rappresentazione esplicita di regole. Inoltre il comportamento desiderato viene appreso attraverso un addestramento basato sulla somministrazione di esempi, senza che sia necessario ipotizzare forme esplicite di conoscenza "innata". Steven Pinker e Alan Prince, in un articolo apparso sullo stesso numero della rivista *Cognition* che ospitava l'articolo di Fodor e Pylyshyn sulla composizionalità (Pinker e Prince 1988), criticarono il modello di Rumelhart e McClelland sulla base di una serie di osservazioni sia di carattere strettamente linguistico, sia relative al comportamento dei bambini nel corso dell'apprendimento. A ciò fece seguito un ricco dibattito, accompagnato da diverse revisioni del modello connessionista finalizzate a rispondere alle varie critiche (per una sintesi dettagliata si veda il cap. 5 di Bechtel e Abrahamsen 2002).

A prescindere da una valutazione specifica su questo particolare problema, le tecniche connessioniste di apprendimento sono molto lontane dallo spiegare in tutta la sua complessità un fenomeno come il linguaggio e il suo apprendimento, e sembra comunque improbabile che una impresa del genere possa mai avere successo.

In generale, le contrapposizioni nette e intransigenti degli anni passati sembrano in parte superate, e sostituite da un atteggiamento più "ecumenico" e tollerante. Certamente si ha che fare con visioni diverse della mente, e su molti problemi le posizioni sono lontane. Ad esempio, i connessionisti sono più propensi a porre l'enfasi sul ruolo dell'apprendimento piuttosto che sull'esistenza di conoscenze innate. Analogamente, rispetto al problema della struttura modulare della mente, le preferenze dei connessionisti vanno piuttosto verso l'ipotesi di architetture cognitive generali che non di moduli computazionali specializzati. Tuttavia, non si tratta di contrapposizioni assolute, né si ha a che fare con schieramenti monolitici. Ad esempio il connessionismo è del tutto compatibile con un tipo di atteggiamento innatista secondo cui le informazioni innate sono, per così dire, "cablate" nell'architettura delle reti. Inoltre gli ultimi due decenni hanno visto emergere nuove impostazioni e nuovi schieramenti (cognizione situata, sistemi dinamici, sistemi reattivi, eccetera – al proposito si veda ad esempio Marruffa 2003) che hanno reso ulteriormente complesso e articolato il panorama delle scienze cognitive, per cui oggi la contrapposizione connessionismo/scienze cognitive classiche non monopolizza più il dibattito. Data questa situazione, il problema generale è caso mai quello di far convivere e interagire in modo proficuo spiegazioni di tipo diverso, che spesso sembrano avere successo per tipi diversi di fenomeni cognitivi.

Riferimenti bibliografici

- Abrahamsen, A., Bechtel, W. e Graham, G. (2004). *Menti, cervelli e calcolatori. Storia della scienza cognitiva*, Laterza, Roma-Bari; tr. it. della prima parte di W. Bechtel. e G. Graham (a cura di), *A Companion to Cognitive Science*, Basil Blackwell, Oxford, 1998.
- Bechtel, W. e Abrahamsen, A. (2002). *Connectionism and the Mind. Parallel Processing, Dynamics, and evolution in Networks*. Blackwell, Oxford (seconda edizione).
- Churchland, P. e Sejnowski, T. (1989). Rappresentazione neurale e computazione neurale. *Sistemi Intelligenti*, 1(2):177-212.
- Churchland, P.S., e Sejnowski, T.J. (1992). *The Computational Brain*. MIT Press, Cambridge, MA; tr. it. *Il cervello computazionale*, Il Mulino, Bologna, 1995
- Cordeschi, R. (2002). *The Discovery of the Artificial. Behavior, Mind and Machines Before and Beyond Cybernetics*, Kluwer Academic Publishers, Dordrecht.
- Feldman, J.A. (1989). Connectionist representation of concepts. In R. Pfeifer, Z. Schreter, F. Fogelman-Soulié e L. Steels (a cura di), *Connectionism in Perspective*, North Holland, Amsterdam.
- Feldman, J.A. e Ballard, F.H. (1982). Connectionist models and their properties. *Cognitive Science*, 6:205-54.
- Floreato, D. e Mattiussi, C. (2002). *Manuale sulle reti neurali*. Il Mulino, Bologna (seconda edizione).
- Fodor, J. e Pylyshyn, Z. (1988). "Connectionism and cognitive architecture: A critical analysis". *Cognition*, 28, pp. 3-71. 73-193
- Hebb, D.O. (1949). *The Organization of Behavior*. John Wiley, New York.
- Hillis, D. (1986). *The Connection Machine*. MIT Press, Cambridge, Mass.
- Hinton, G.E. , McClelland, J.L. e Rumelhart, D.E. (1986). Distributed representations. In Rumelhart et al. (1986).
- Hinton, G.E. e Sejnowski T.J. (1986). Learning and relearning in Boltzmann machines. In Rumelhart et al. (1986).
- Marruffa, M. (2003). *Filosofia della psicologia*, Laterza, Roma-Bari.
- McClelland, J.L. (1981). Retrieving general and specific information from stored knowledge of specifics. *Proceedings of the Third Annual Meeting of the Cognitive Science Society*, 170-2.
- McClelland, J.L., Rumelhart, D.E e il gruppo di ricerca PDP (1986). *Parallel Distributed Processes. Explorations in the Microstructure of Cognition. Vol. 2: Psychological and Biological Models*. MIT Press, Cambridge, Mass. Trad. it. parziale: *La microstruttura dei processi cognitivi. Elaborazione distribuita in parallelo*, 1991, Il Mulino, Bologna.
- McClelland, J.L., Rumelhart, D.E. e Hinton, G.E. (1986). The appeal of parallel distributed processing. In Rumelhart et al. (1986).
- McCulloch, W.S. e Pitts, W.H. (1943). A logical calculus of ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5.
- Minsky, M. e Papert, S. (1969). *Perceptrons*. MIT Press, Cambridge, Mass. (seconda edizione ampliata: 1987).
- Parisi, D. (1989a). *Intervista sulle reti neurali*. Il Mulino, Bologna.
- Parisi, D. (1989b). La mente come cervello. *Sistemi Intelligenti*, 1(2): 213-242.
- Pinker, S. e Prince, A. (1988). "On language and connectionism: Analysis of a parallel distributed processing model of language acquisition", *Cognition*, 28, pp. 73-193.
- Rosenblatt, F. (1962). *Principles of Neurodynamics*. Spartan, New York.
- Rumelhart, D.E., Hinton, G.E. e Williams, R.J. (1986). Learning internal representations by error propagation. In Rumelhart et al. 1986.
- Rumelhart, D.E. e McClelland, J.L. (a cura di) (1986). *Parallel Distributed Processing. Explorations in the Microstructure of Cognition*. MIT Press, Cambridge, MA; tr. it. parziale *PDP microstruttura dei processi cognitivi*, Il Mulino, Bologna, 1991.

- Rumelhart, D.E., Smolensky, P., McClelland, J.L. e Hinton, G.E. (1986). Schemata and sequential thought processes in PDP models. In McClelland et al. (1986).
- Smolensky, P. (1986). Information processing in dynamical systems: foundations of harmony theory. In Rumelhart et al. (1986).
- Smolensky, P. (1988). "On the proper treatment of connectionism", *Behavioral and Brain Sciences*, 11, pp. 1-74; tr. it. *Il connessionismo tra simboli e neuroni*, Marietti, Genova, 1992.
- Smolensky, P. (1989). Connectionism and constituent structure. In R. Pfeifer, Z. Schreter, F. Fogelman-Soulié e L. Steels (a cura di), *Connectionism in perspective*, North Holland, Amsterdam.
- Smolensky, P. (1990). Tensor product variable binding and the representation of symbolic structures in connectionist systems. *Artificial Intelligence*, 46:159-216.
- Somenzi, V. e Cordeschi, R. (a cura di) (1994). *La filosofia degli automi*, 2a edizione Bollati Boringhieri, Torino.