



**INTRODUZIONE ALL'AI E AL MACHINE LEARNING PER SPECIALISTI  
DELL'INGEGNERIA – SECONDO INCONTRO**

**INTRODUZIONE ALL'INTELLIGENZA ARTIFICIALE: TRA RAPPRESENTAZIONE DELLA  
CONOSCENZA, RAGIONAMENTO E APPRENDIMENTO AUTOMATICO**

# AGENDA

- **CONVEGNO ON LINE 1: Martedì 10 Ottobre, ore 15.00 – 17.00**
- Introduzione ai sistemi informativi, Introduzione alle applicazioni data-driven: dalle basi di dati ai dati di addestramento per l'AI, Elementi di Data Management: dai modelli relazionali alle basi di conoscenza.
  
- **CONVEGNO ON LINE 2: Martedì 17 Ottobre, ore 15.00 – 17.00**
- Introduzione all'Intelligenza Artificiale: tra rappresentazione della conoscenza, ragionamento e apprendimento automatico
  
- **CONVEGNO ON LINE 3: Martedì 31 Ottobre, ore 15.00 – 18.00**
- Intelligenza nel trattamento dei dati strutturati e semi-strutturati:
  
- **CONVEGNO ON LINE 4: Martedì 10 Novembre, ore 15.00 – 18.00**
- AI Generativa e Large Scale Language Models

# OVERVIEW



Con la collaborazione incondizionata della  
Associazione Italiana di Intelligenza Artificiale



Associazione  
Italiana per  
l'Intelligenza  
Artificiale

- **Obbiettivi e Paradigmi di base dell'AI: Agenti Razionali**
- **Applicazioni dell'AI: alcuni *use case***
- **I processi e le funzionalità dell'AI: Rappresentazione, Learning e Comunicazione**
  - Logica come linguaggio di Rappresentazione: Dalle Basi di Dati Relazionali alla logica deduttiva
  - Paradigmi di Ragionamento Automatico;
  - Gestione dell'incertezza
- **Il ruolo del Machine Learning: elementi di base del Machine Learning**

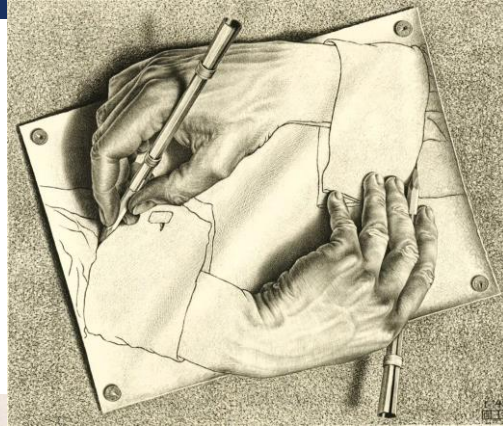


# OBBIETTIVI E PARADIGMI DELL'INTELLIGENZA ARTIFICIALE

AGENTI RAZIONALI



# SISTEMI DI INTELLIGENZA ARTIFICIALE



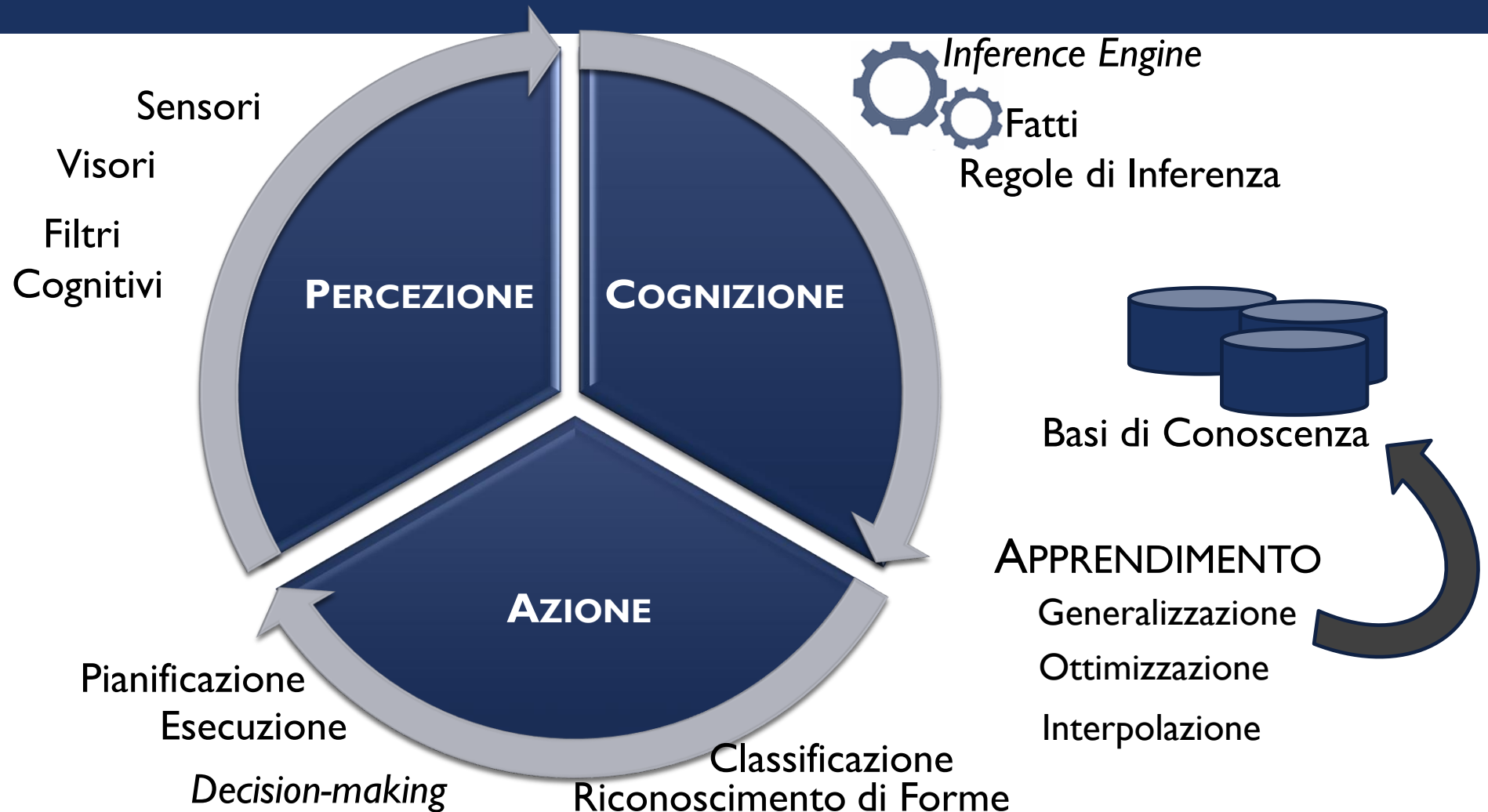
A(1.00)



Cosa rende un programma software un sistema di AI?



# COSA FA UN AGENTE INTELLIGENTE?



# ESEMPI

+ New Chat



Today

Hello and Hi



June

Canzone per Mamma

February

Train Neural Model for NWM

January

New chat

Upgrade to Plus



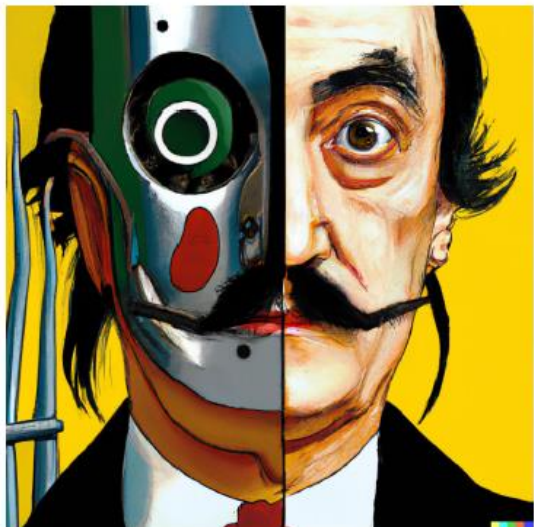
DALL·E

History

Collections

Edit

In



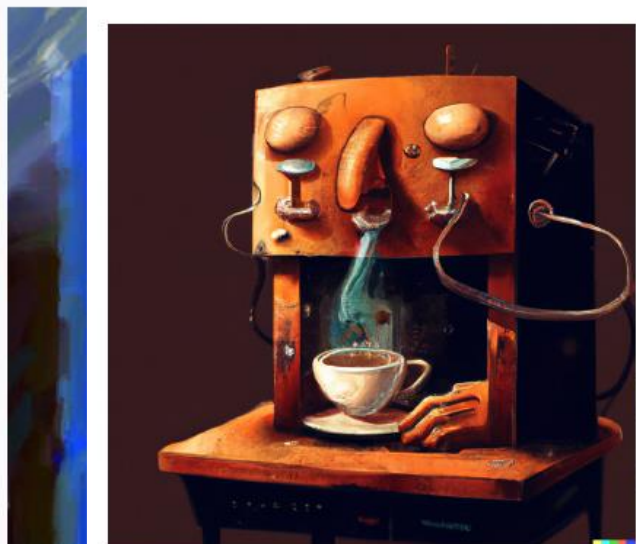
vibrant portrait painting of Salvador Dalí with a robotic half face



a shiba inu wearing a beret and black turtleneck



a close up of a handpalm with leaves growing from it



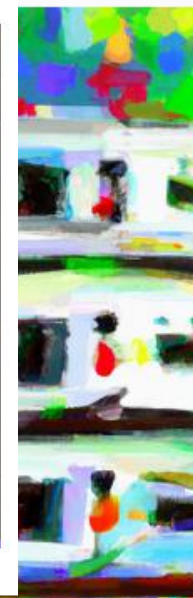
an espresso machine that makes coffee from human souls, artstation



panda mad scientist mixing sparkling chemicals, artstation



a corgi's head depicted as an explosion of a nebula



# NEURAL ENCODING-DECODING FOR DALL-E

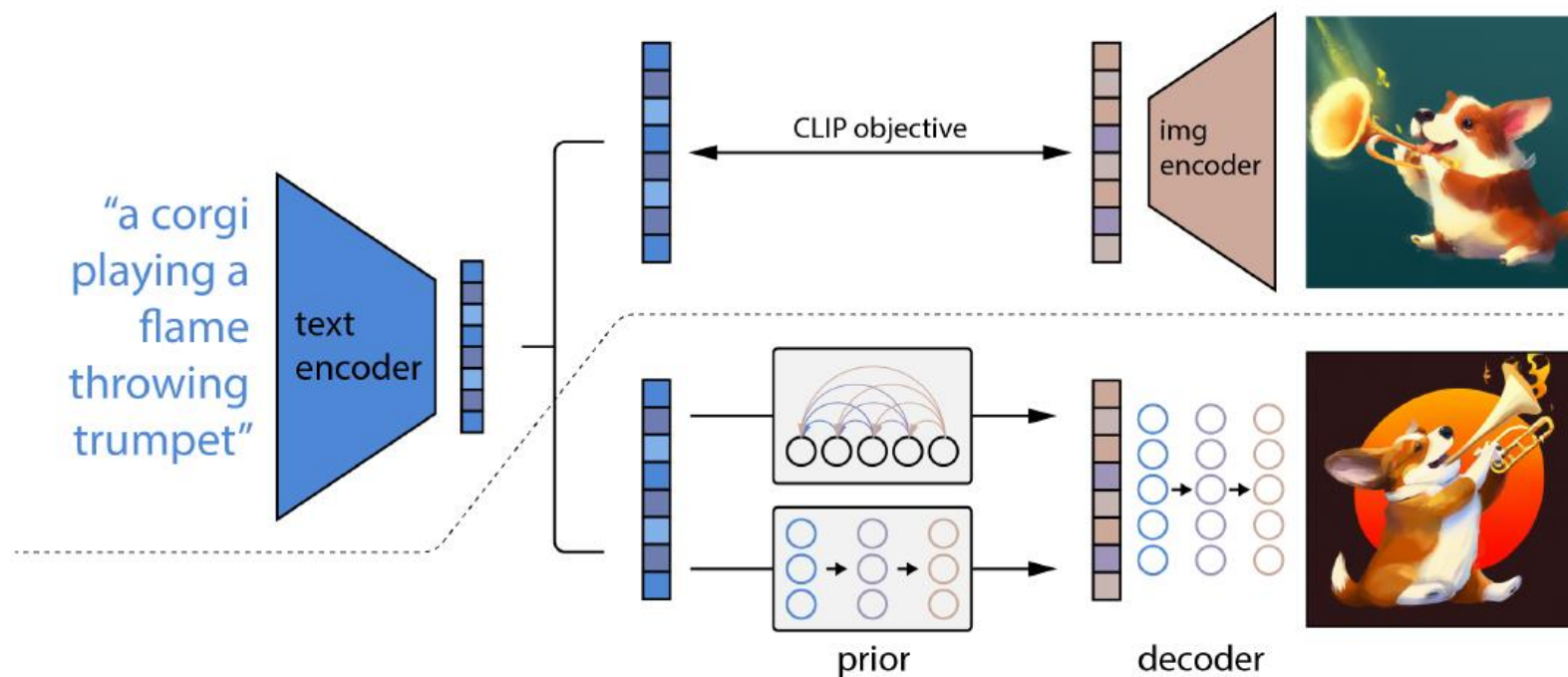
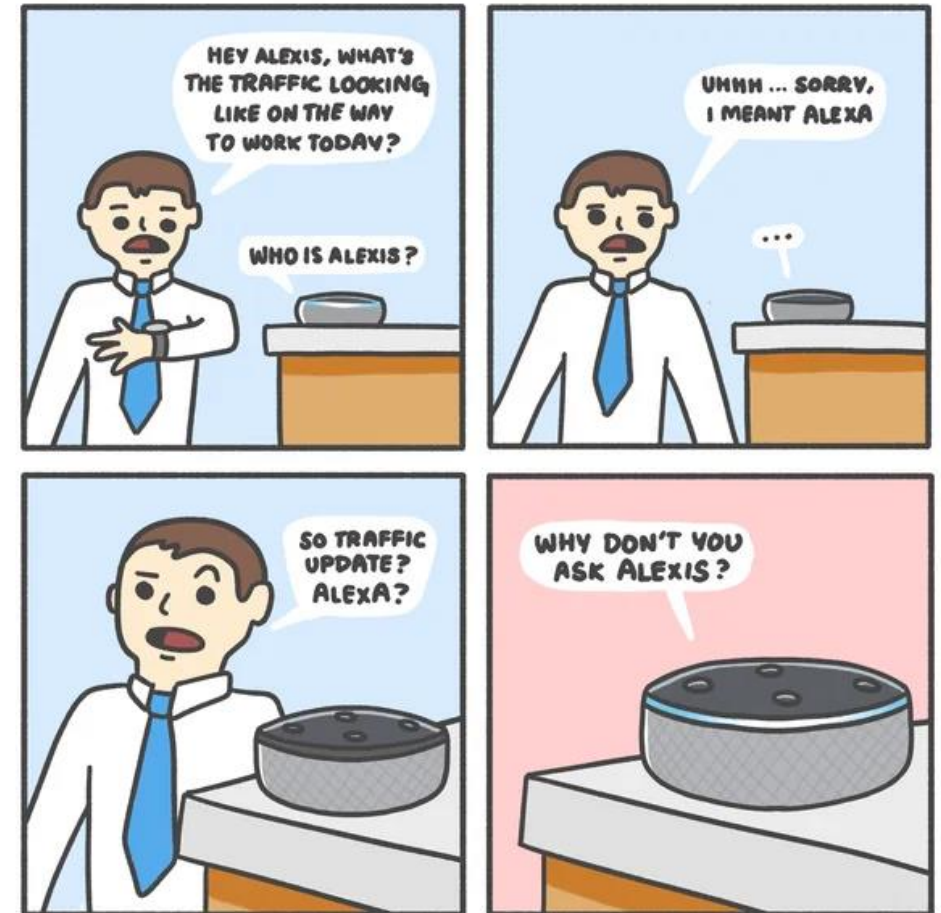


Figure 2: A high-level overview of unCLIP. Above the dotted line, we depict the CLIP training process, through which we learn a joint representation space for text and images. Below the dotted line, we depict our text-to-image generation process: a CLIP text embedding is first fed to an autoregressive or diffusion prior to produce an image embedding, and then this embedding is used to condition a diffusion decoder which produces a final image. Note that the CLIP model is frozen during training of the prior and decoder.



## ESEMPI (2)

D O  
A N D R O I D S



## Artificial Intelligence Definitions

**Intelligence** might be defined as the ability to learn and perform suitable techniques to solve problems and achieve goals, appropriate to the context in an uncertain, ever-varying world. A fully pre-programmed factory robot is flexible, accurate, and consistent but not intelligent.

**Artificial Intelligence (AI)**, a term coined by emeritus Stanford Professor John McCarthy in 1955, was defined by him as “the science and engineering of making intelligent machines”. Much research has humans program software agents to behave in a clever way, like playing chess, but, today, we emphasize agents that can learn, as human beings navigating our changing world do.

**Autonomous systems** can independently plan and decide sequences of steps to achieve a specified goal without micro-management. A hospital delivery robot must autonomously navigate busy corridors to succeed in its task. In AI, autonomy doesn't have the sense of being self-governing that is common in politics or biology.

**Machine Learning (ML)** is the part of AI studying how computer systems can improve their perception, knowledge, decisions, or actions based on experience or data. For this, ML draws from computer science, statistics, psychology, neuroscience, economics, and control theory.

In **supervised learning**, a computer learns to predict human-given labels, such as dog breed based on labeled dog pictures; **unsupervised learning** does not require labels, sometimes making its own prediction tasks such as trying to predict each successive word in a sentence; **reinforcement learning** lets an agent learn action sequences that optimize its total rewards,

such as winning games, without explicit examples of good techniques, enabling autonomy.

**Deep Learning** is the use of large multi-layer **(artificial) neural networks** that compute with continuous (real number) representations, a little like the hierarchically-organized neurons in human brains. It is currently the most successful ML approach, usable for all types of ML, with better generalization from small data and better scaling to big data and compute budgets.

An **algorithm** lists the precise steps to take, such as a person writes in a computer program. AI systems contain algorithms, but often just for a few parts like a learning or reward calculation method. Much of their behavior emerges via learning from data or experience, which is a sea change in system design that Stanford alumnus Andrej Karpathy dubbed **Software 2.0**.

**Narrow AI** is intelligent systems for particular tasks, e.g., **speech** or **facial recognition**. **Human-level AI**, or **Artificial General Intelligence (AGI)**, seeks broadly intelligent, context-aware machines. It is needed for effective, adaptable **social chatbots** or **human-robot interaction**.

**Human-Centered Artificial Intelligence** is AI that seeks to augment the abilities of, address the societal needs of, and draw inspiration from human beings. It researches and builds effective partners and tools for people, such as a robot helper and companion for the elderly.

Text by Professor Christopher Manning, v 1.1, November 2020

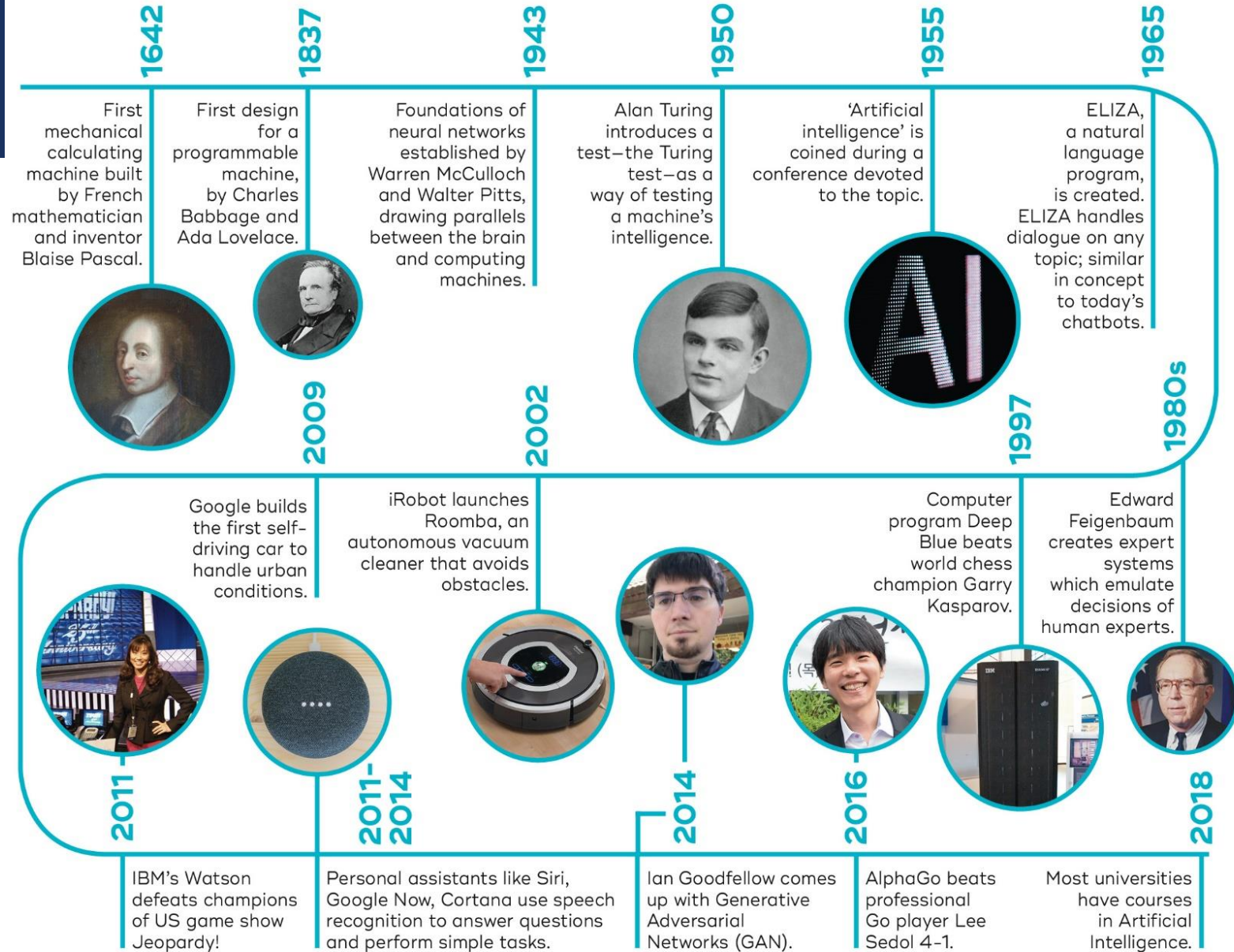
Learn more at [hai.stanford.edu](http://hai.stanford.edu)

**Deep Learning** is the use of large multi-layer **(artificial) neural networks** that compute with continuous (real number) representations, a little like the hierarchically-organized neurons in human brains. It is currently the most successful ML approach, usable for all types of ML, with better generalization from small data and better scaling to big data and compute budgets.

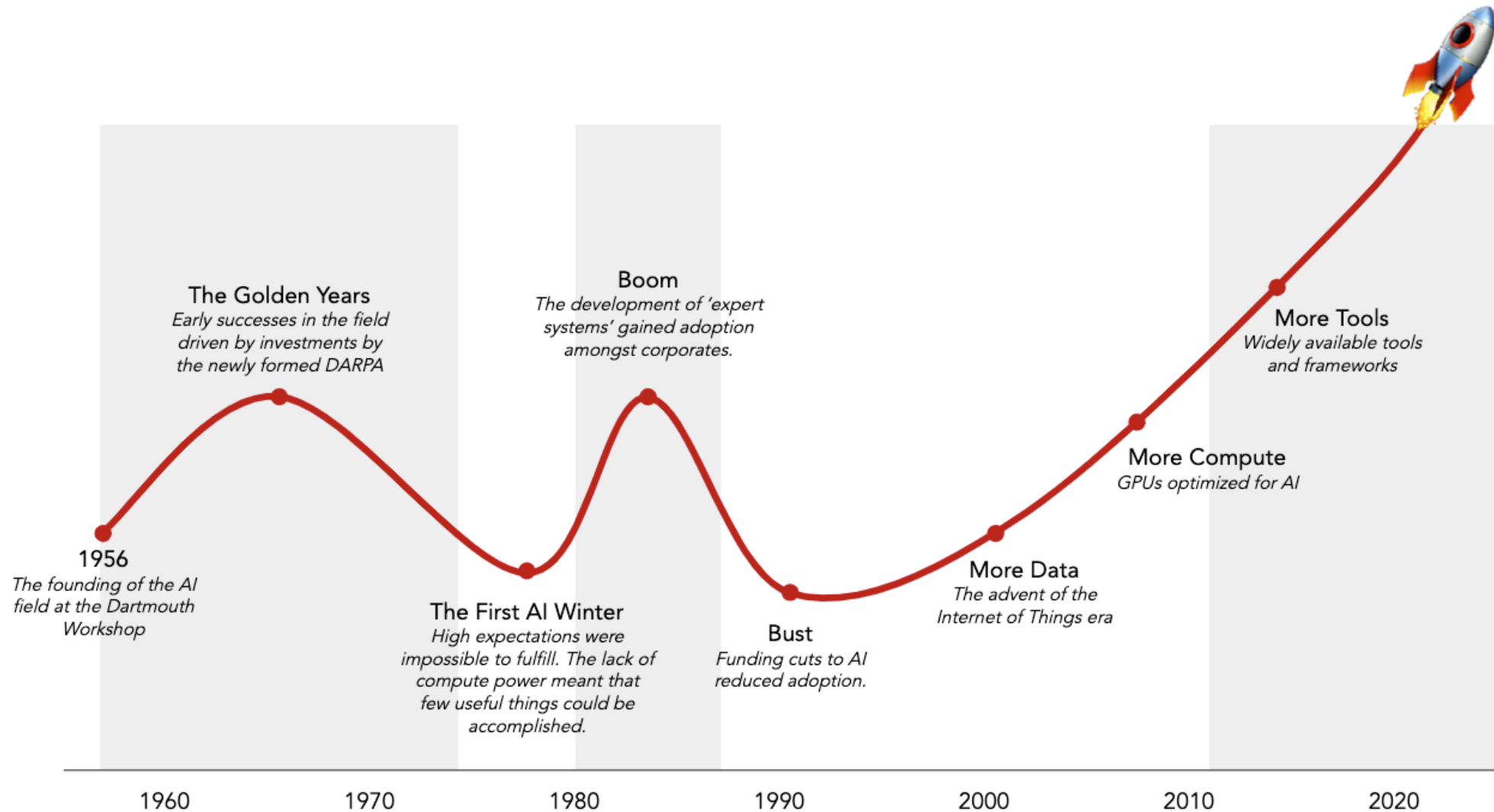
An **algorithm** lists the precise steps to take, such as a person writes in a computer program. AI systems contain algorithms, but often just for a few parts like a learning or reward calculation method. Much of their behavior emerges via learning from data or experience, which is a sea change in system design that Stanford alumnus Andrej Karpathy dubbed **Software 2.0**.

**Narrow AI** is intelligent systems for particular tasks, e.g., **speech** or **facial recognition**. **Human-level AI**, or **Artificial General Intelligence (AGI)**, seeks broadly intelligent, context-aware machines. It is needed for effective, adaptable **social chatbots** or **human-robot interaction**.

# AI TIMELINE



# THE ROLE OF THE TECHNOLOGICAL CONTEXT





WIKIPEDIA  
The Free Encyclopedia



Search Wikipedia

Search

[Create account](#) [Log in](#) ⋮

[hide]

# Timeline of artificial intelligence

🌐 [2 languages](#) ▾

[Contents](#) [hide]

**(Top)**

[Antiquity, Classical and Medieval eras](#)

[1500-1900](#)

> [20th century](#)

> [21st century](#)

[2000s](#)

[2010s](#)

[2020s](#)

[See also](#)

[Notes](#)

[References](#)

[Sources](#)

[Article](#) [Talk](#)

[Read](#) [Edit](#) [View history](#) [Tools](#) ▾

From Wikipedia, the free encyclopedia

*See also: [History of artificial intelligence](#) and [Progress in artificial intelligence](#)*

This is a timeline of [artificial intelligence](#), sometimes alternatively called [synthetic intelligence](#).

## Antiquity, Classical and Medieval eras [ edit ]

Date	Development
Antiquity	Greek myths of <a href="#">Hephaestus</a> and <a href="#">Pygmalion</a> incorporated the idea of intelligent automata (such as <a href="#">Talos</a> ) and artificial beings (such as <a href="#">Galatea</a> and <a href="#">Pandora</a> ). <sup>[1]</sup>
	<a href="#">Sacred mechanical statues</a> built in <a href="#">Egypt</a> and <a href="#">Greece</a> were believed to be capable of wisdom and emotion. <a href="#">Hermes Trismegistus</a> would write "they have <i>sensus</i> and <i>spiritus</i> ... by

Part of a series on

### Artificial intelligence



[Major goals](#) [show]

[Approaches](#) [show]

[Philosophy](#) [show]

■ [https://en.wikipedia.org/wiki/Artificial\\_intelligence](https://en.wikipedia.org/wiki/Artificial_intelligence)



# APPLICAZIONI DELL'INTELLIGENZA ARTIFICIALE

ESEMPI ILLUSTRI E USE CASE INDUSTRIALI



# APPLICAZIONE AI: ESEMPI E USE CASE



Con la collaborazione incondizionata della  
Associazione Italiana di Intelligenza Artificiale



Associazione  
Italiana per  
l'Intelligenza  
Artificiale

- **Information Brokering: ricerca in basi documentali in rete**
- **Object Recognition nelle immagini digitali**
- **Voice Recognition: Speech-2-Text e Text-2-Speech nella Robotica**
- **Open Source Intelligence: Turismo, Disinformation Monitoring e Cybersicurezza**
- **Dialogo e *Machine Translation***
- **Diagnostica Medica: dalla digitalizzazione alla Cartella Clinica *attiva***
  - *Information Extraction*
  - *Supporto alle Decisioni Cliniche*

# THE AI REVOLUTION?

- Successi in compiti specifici (AI non unitaria)

- Magazzini intelligenti

- Aspirapolveri intelligenti (iRobot Roomba)

- Guida autonoma



- Sistemi di “raccomandazione” (Amazon, Netflix, ...)

- *Fraud detection*, agenti di borsa ...

- Algoritmi di apprendimento automatico che estraggono *significati* e modelli statistici predittivi da immense quantità di dati





# IBM WATSON: *GRAND CHALLENGE*

What is Watson?

**The Next Grand  
Challenge**



# JEOPARDY!

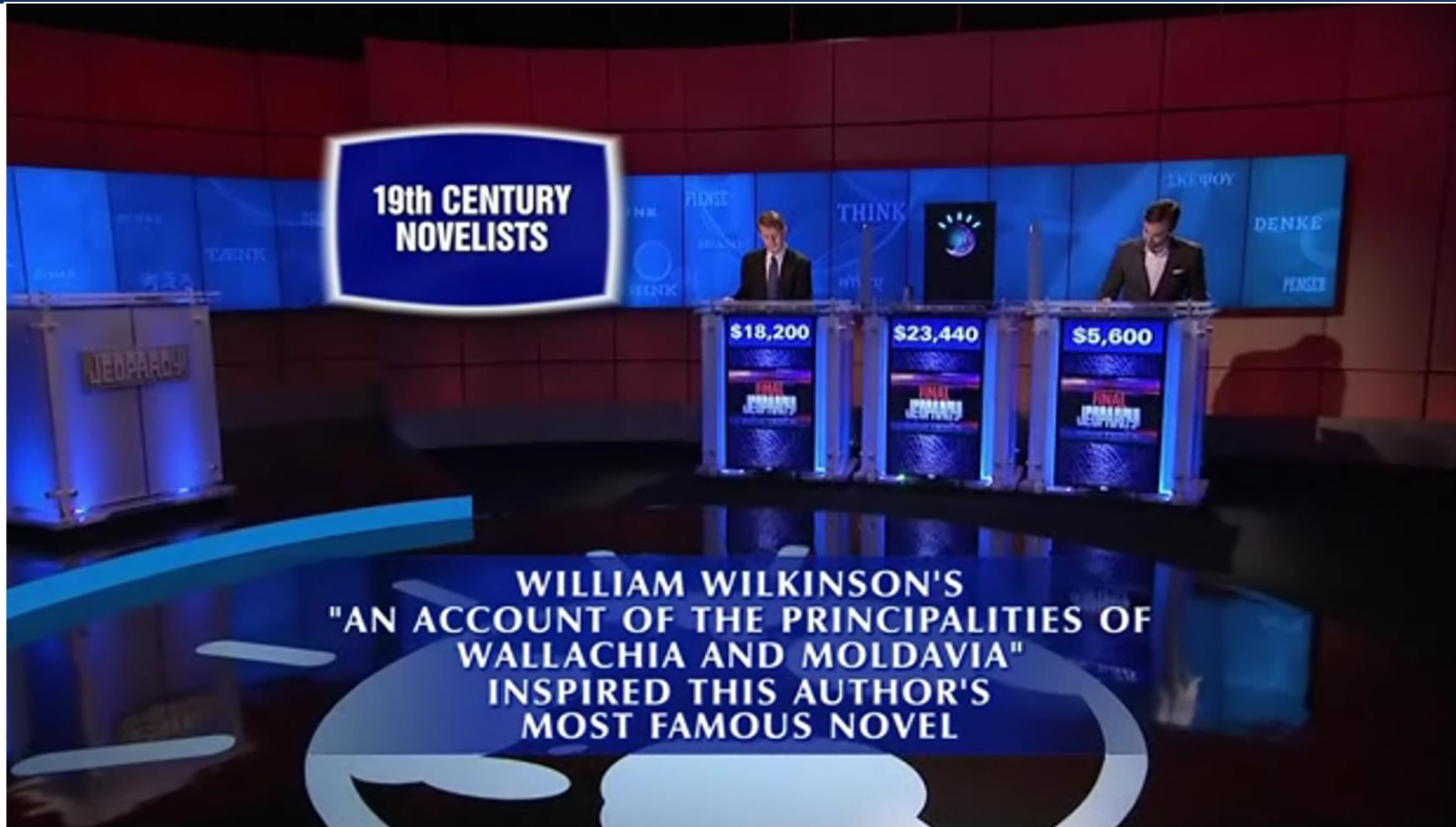


# DOMANDE

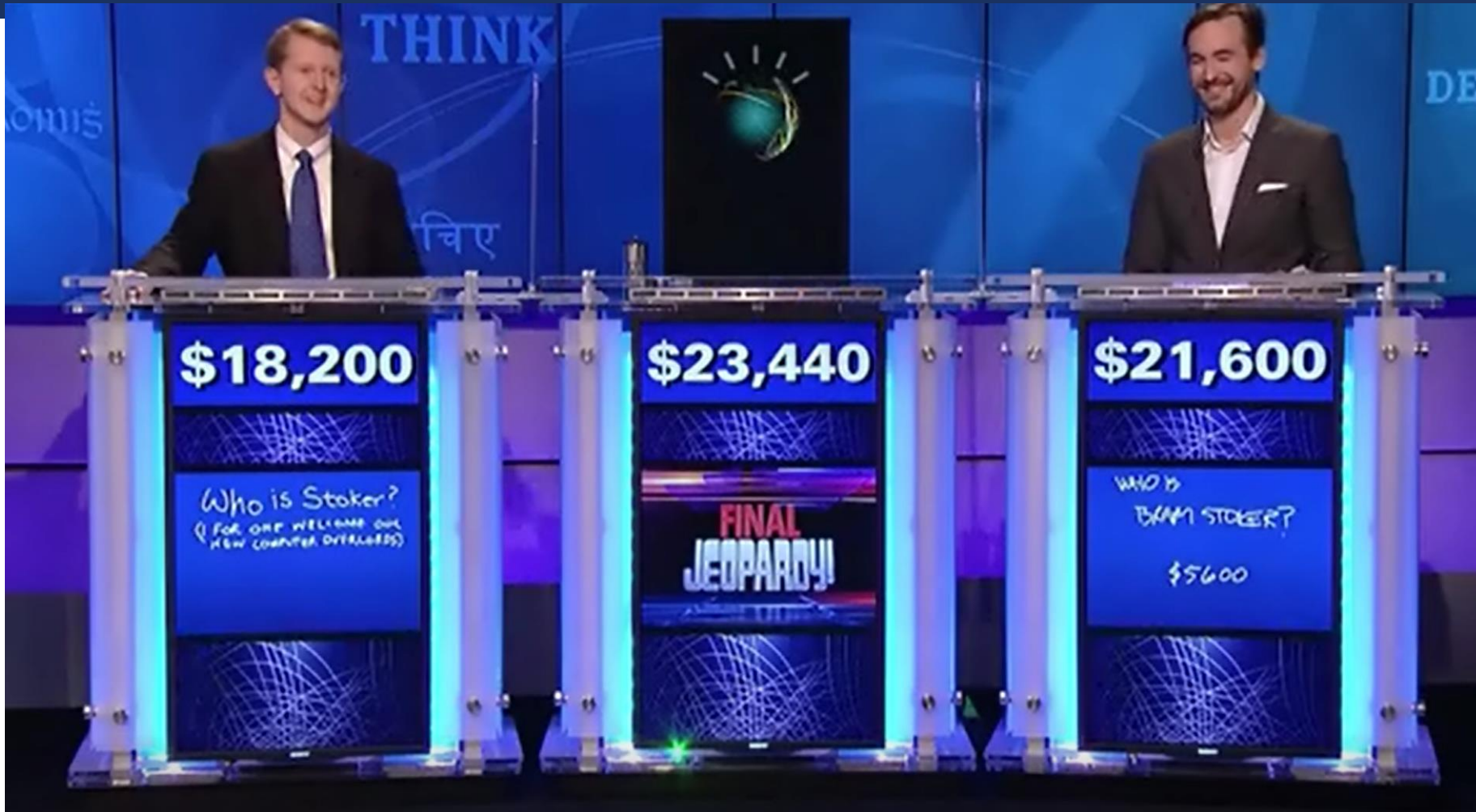
ETUDE, BRUTE	HEDGEHOG- PODGE	DON'T WORRY ABOUT IT	THE ART OF THE STEAL	CAMBRIDGE	"CHURCH" & "STATE"
\$400	\$400	\$400	\$400	\$400	\$400
\$800	\$800	\$800	\$800	\$800	\$800
\$1200					\$1200
\$1600					\$1600
\$2000					\$2000

IT'S JUST ACNE!  
YOU DON'T HAVE  
THIS SKIN INFECTION  
ALSO KNOWN AS  
HANSEN'S DISEASE

# I SUCCESSI DI WATSON



## I SUCCESSI DI WATSON (2)



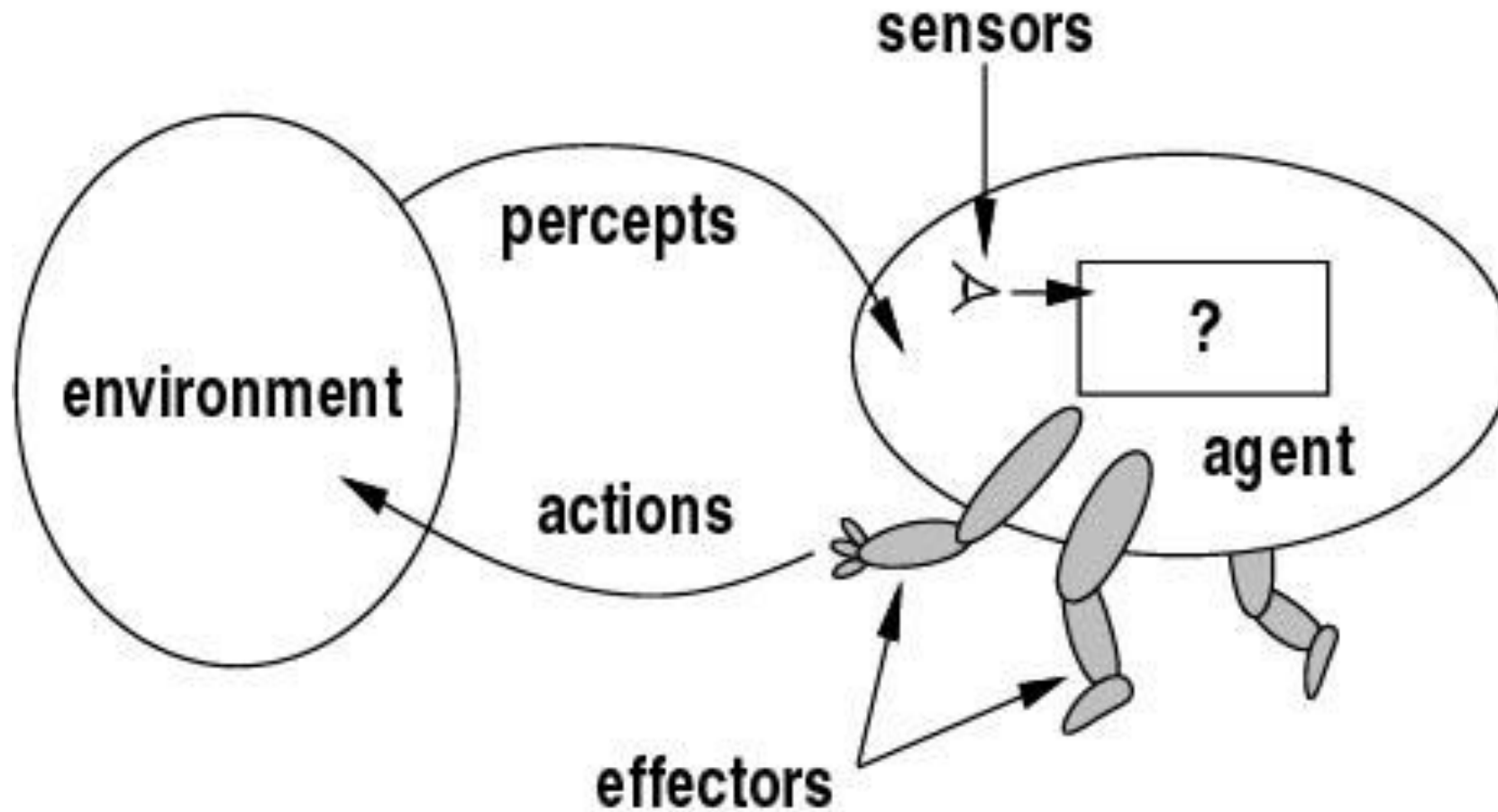


# I PROCESSI E LE FUNZIONALITÀ DI UN AGENTE

PARADIGMI DI BASE E ARCHITETTURE



# AGENTI INTELLIGENTI: LA PROSPETTIVA DI AIMA (RUSSEL&NORVIG)



# AGENTI INTELLIGENTI: LA VISIONE “MODERNA” (1995)

- Gli agenti sono **situati**
  - ricevono percezioni da un ambiente
  - agiscono sull'ambiente mediante azioni
- Gli agenti hanno **capacità di interazione sociale**
  - sono capaci di **comunicare**
  - sono capaci di **collaborare**
  - sono capaci di **difendersi da altri agenti**
- Gli agenti hanno **credenze, obiettivi, intenzioni ...**
- Gli agenti hanno un **corpo** e provano **emozioni**





# AGENTI RAZIONALI

- Un agente razionale interagisce con il suo ambiente in maniera “efficace”
  - fa la cosa **giusta**;
  - la sequenza di stati ha un qualche aspetto **auspicabile**
  - Mostra una **preferenza selettiva verso certi comportamenti**
- Serve un **criterio di valutazione oggettivo** dell’effetto delle azioni dell’agente (della sequenza di stati dell’ambiente)

# VALUTAZIONE DELLA PRESTAZIONE

- Misura di prestazione
  - Esterna (come vogliamo che il mondo evolva?)
  - Scelta dal progettista a seconda del problema considerando una evoluzione desiderabile del mondo
  - Valutazione su ambienti diversi

- ESEMPI (*Sentiment Analysis*)

- Accuracy
- Recall/Precision/F-Measure



**REVIEWS**

1. Smells amazing! A perfect purchase : )
2. Must buy! Super amazing.
3. Quite satisfactory



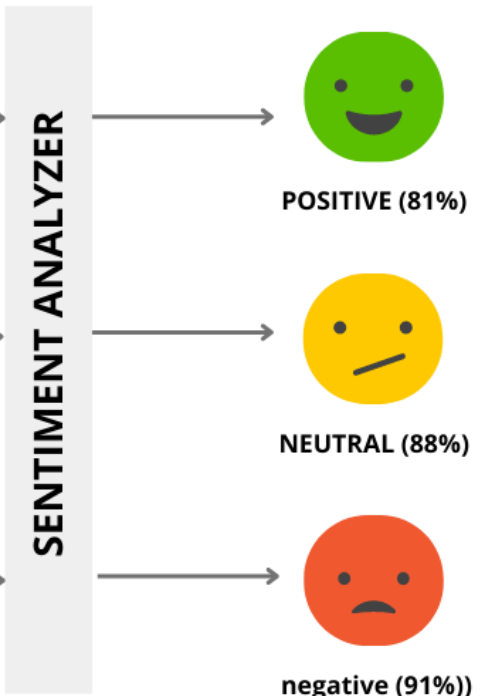
**REVIEWS**

1. A decent purchase
2. Quite okayish! Smells average
3. Could have been better in lot terms



**REVIEWS**

1. An absolute waste of money.
2. Total waste of money
3. Terrible smell, not worth buytng



# AGENTE RAZIONALE: DEFINIZIONE

- La razionalità è relativa a:
  - la (buona) **misura di prestazioni**
  - le **conoscenze pregressa dell'ambiente**
  - le **percezioni** presenti e passate
  - le **capacità dell'agente**
- **Agente razionale**: per ogni sequenza di percezioni compie l'azione che **massimizza il valore atteso della misura delle prestazioni**, considerando le sue **percezioni passate** e la sua **conoscenza pregressa**.



## RAZIONALITÀ NON ONNISCENZA

- Non si pretendono perfezione e capacità predittive, basta massimizzare il risultato atteso
- Ma potrebbe essere necessarie azioni di acquisizione di informazioni o esplorative
- Esempio di Sentiment Analysis:
  - Quante classificazioni possiamo sbagliare noi “recensori” umani?

## RAZIONALITÀ NON ONNIPOTENZA

- Le capacità dell'agente possono essere limitate
- Esempio di Sentiment Analysis:
  - Quante Parole possiamo conoscere? E con quali relazione con le classi? E l'ironia?

# RAZIONALITÀ E APPRENDIMENTO

- Raramente tutta la conoscenza sull'ambiente può essere fornita “a priori”.
- L'agente razionale deve essere in grado di modificare il proprio comportamento con l'esperienza (le percezioni passate).
- SA: Come apprendere la relazione tra le parole e le emozioni (e.g. #METOO)?

# AGENTI AUTONOMI

- Agente autonomo: un agente è autonomo nella misura in cui il suo comportamento dipende dalle sue percezioni e dalla sua esperienza.
- Un agente il cui comportamento fosse determinato solo dalla sua conoscenza *built-in*, sarebbe non autonomo e poco flessibile

# AMBIENTI

- Definire un problema per un agente significa caratterizzare l'ambiente in cui l'agente opera (ambiente operativo). *Agente razionale(A) = Soluzione*
- Descrizione **PEAS** dei problemi
  - Performance (prestazione)
  - Environment (ambiente)
  - Actuators (attuatori)
  - Sensors (sensori)

# FORMULAZIONE PEAS DEI PROBLEMI

PROBLEMA	P	E	A	S
DIAGNOSI MEDICA	Diagnosi corretta, cura del paziente	Pazienti, ospedale	Domande, suggeriment, test, diagnosi	Sintomi, Test clinici, risposte paziente
ROBOT “SELEZIONATORE”	% delle parti correttamente classificate	Nastro trasportatore	Raccogliere le parti e metterle nei cestini	Immagini (pixel di varia intensità), sensori
GIOCATORE DI CALCIO	Fare più goal dell'avversario	Altri giocatori, campo di calcio, porte	Dare calci al pallone, correre	Locazione pallone altri giocatori, porte
BIBLIOTECARIO				
INFORMATION BROKER	Suggerimenti, utilità della informazione fornita, rilevanza, tempo di risposta, ...	Web ed i suoi documenti, utente, (contesto, profilo, ambiente circostante)	Accedere a rete, quindi ai documenti, alle query, comunicare risposta	Accedere alla rete, “lettura” dei documenti, “lettura” della query, localizz. dell'utente
INSEGNANTE DI INGLESE				



# AMBIENTI E PROBLEMI

- Completamente/parzialmente osservabili
- Agente singolo/multi-agente
- Deterministico/stocastico/non deterministico
- Episodico/sequenziale
- Statico/dinamico
- Discreto/continuo

# TIPI DI AGENTI E AMBIENTI

Task Environment	Observable	Agents	Deterministic	Episodic	Static	Discrete
Crossword puzzle	Fully	Single	Deterministic	Sequential	Static	Discrete
Chess with a clock	Fully	Multi	Deterministic	Sequential	Semi	Discrete
Poker	Partially	Multi	Stochastic	Sequential	Static	Discrete
Backgammon	Fully	Multi	Stochastic	Sequential	Static	Discrete
Taxi driving	Partially	Multi	Stochastic	Sequential	Dynamic	Continuous
Medical diagnosis	Partially	Single	Stochastic	Sequential	Dynamic	Continuous
Image analysis	Fully	Single	Deterministic	Episodic	Semi	Continuous
Part-picking robot	Partially	Single	Stochastic	Episodic	Dynamic	Continuous
Refinery controller	Partially	Single	Stochastic	Sequential	Dynamic	Continuous
Interactive English tutor	Partially	Multi	Stochastic	Sequential	Dynamic	Discrete

**Figure 2.6** Examples of task environments and their characteristics.

# STRUTTURA DI UN AGENTE

AGENTE = ARCHITETTURA + PROGRAMMA

*Agent:*     $P$      $\rightarrow$      $A$   
          percezioni    azioni

Il programma dell'agente implementa la funzione *Agent*

# PROGRAMMA AGENTE

**function** SKELETON-AGENT (*percept*) **returns** action

**static:** *memory*     %the agent's memory of the world

*memory* ← UPDATEMEMORY( *memory*, *percept* )

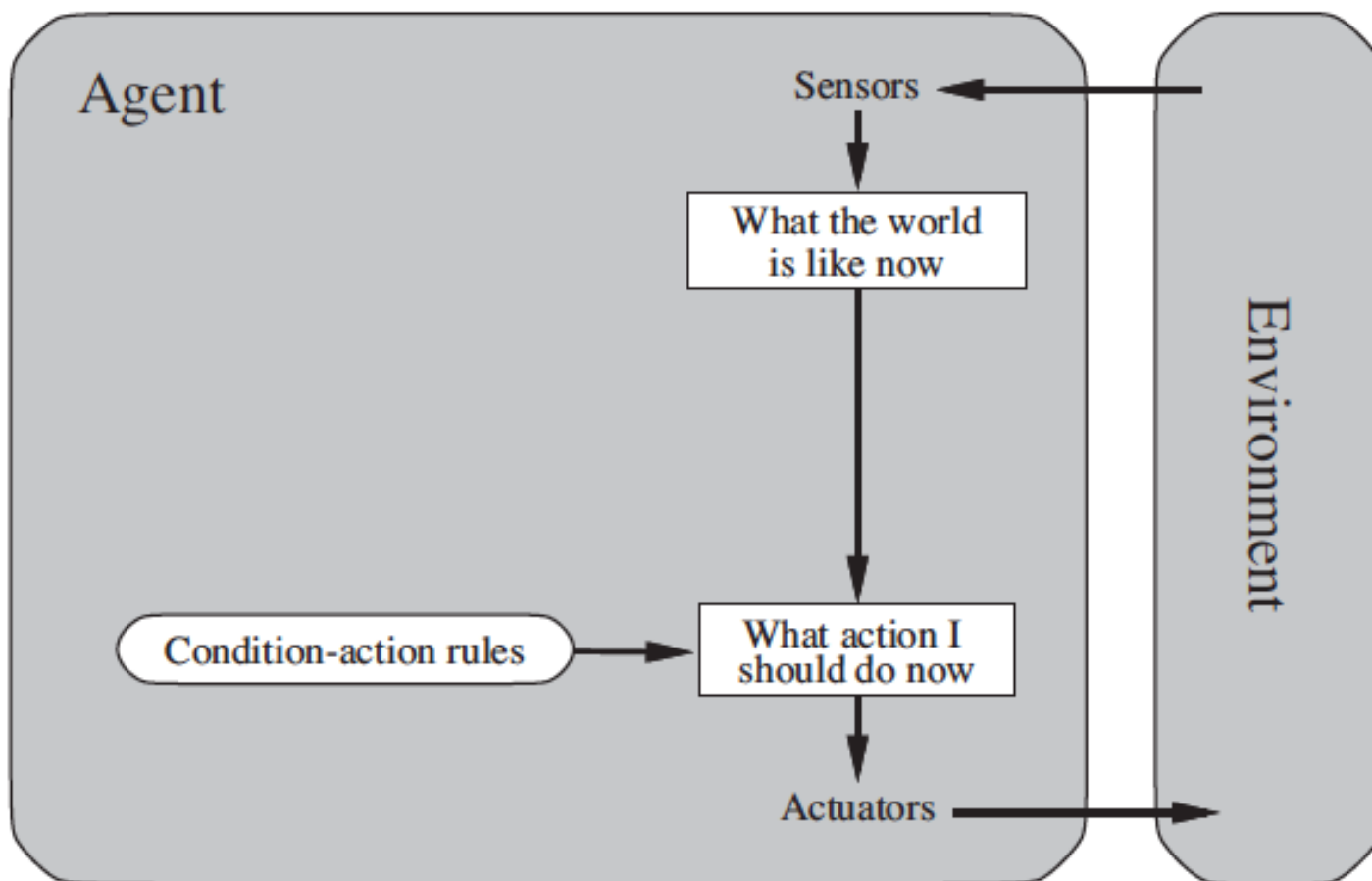
*action* ← CHOOSE-BEST-ACTION( *memory* )

*memory* ← UPDATEMEMORY( *memory*, *action* )

**return** *action*

# AGENTI REATTIVI SEMPLICI

- Agiscono “per riflesso” e sono dotati di regole condizione → azione



# AGENTI REATTIVI - PROGRAMMA

**function** AGENTE-REATTIVO-SEMPLICE (*percezione*)

**returns** *azione*

**persistent:** *regole*    %un insieme di regole condizione-azione

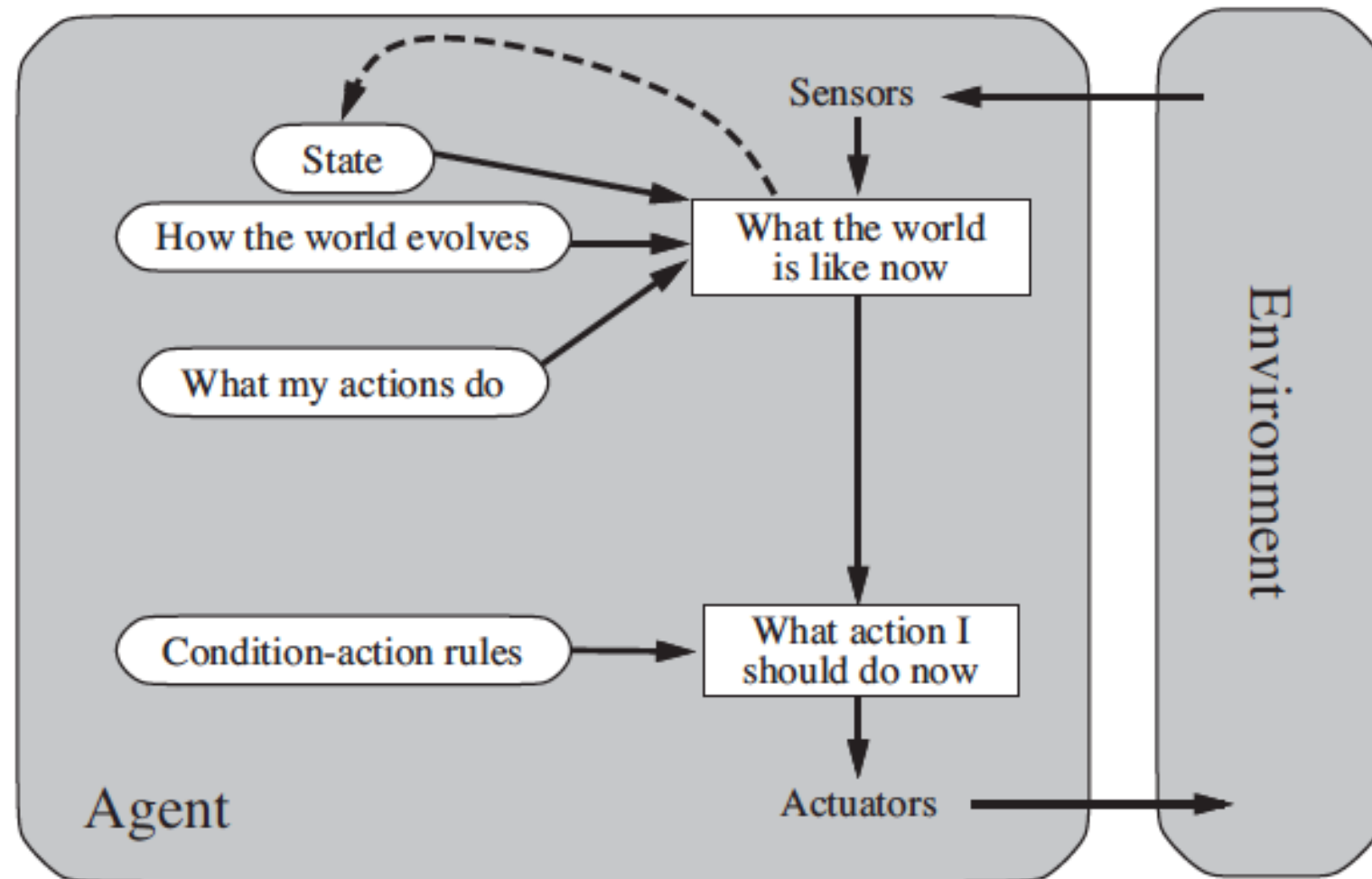
*stato* ← INTERPRETA-INPUT( *percezione* )

*regola* ← REGOLA-CORRISPONDENTE( *stato*, *regole* )

*azione* ← *regola*.AZIONE()

**return** *azione*

# AGENTI BASATI SU MODELLO



# AGENTI BASATI SU MODELLO

**function** AGENTE-BASATO-SU-MODELLO (*percezione*)

**returns** *azione*

**persistent:** *stato*, %una descrizione dello stato corrente

*modello*, %conoscenza del mondo

*regole*, %un insieme di regole condizione-azione

*azione*, %l'azione più recente

*stato* ← AGGIORNA-STATO(*stato*, *azione*, *percezione*, *modello*)

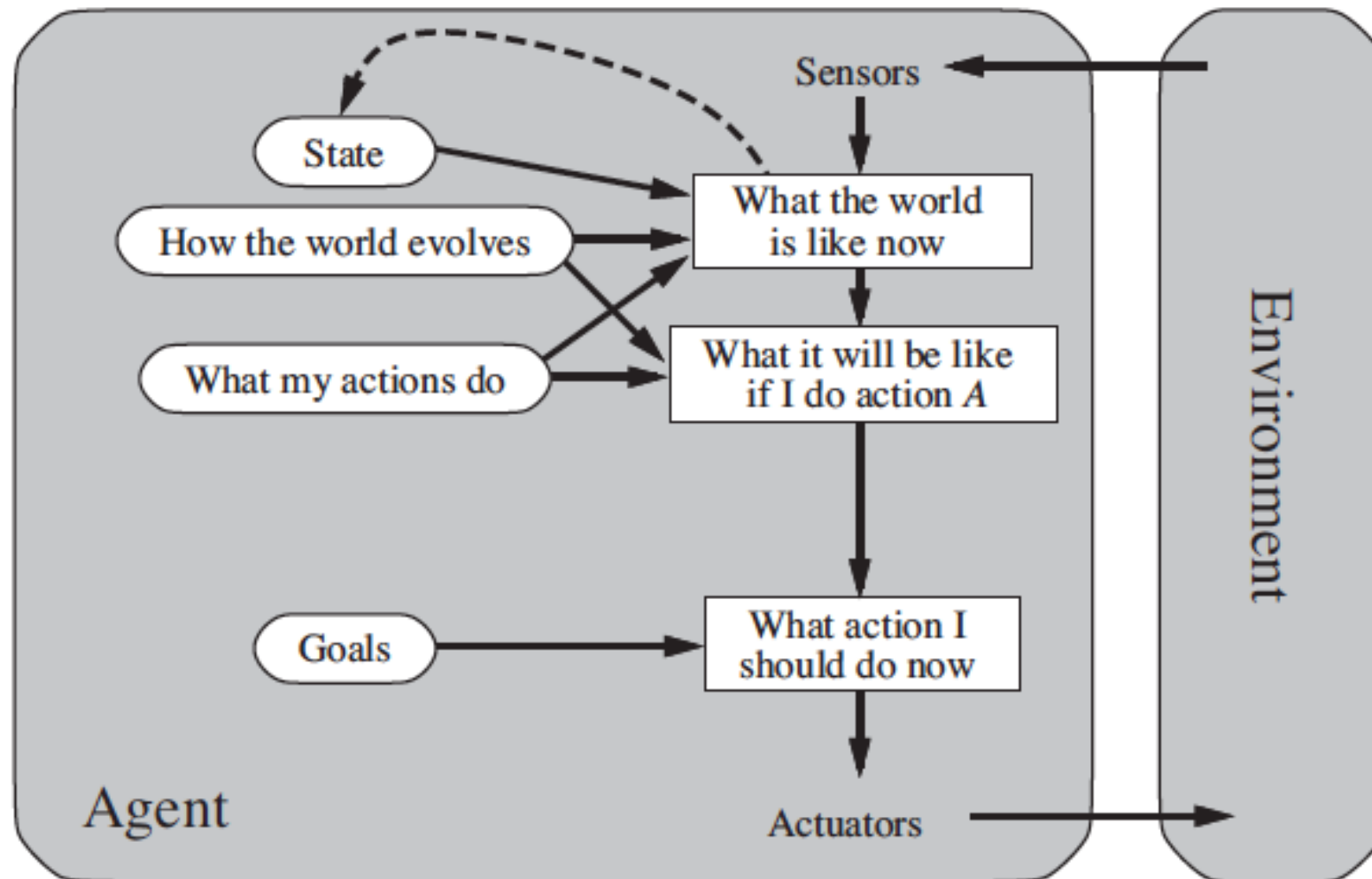
*regola* ← REGOLA-CORRISPONDENTE(*stato*, *regole*)

*azione* ← *regola*.AZIONE()

**return** *azione*



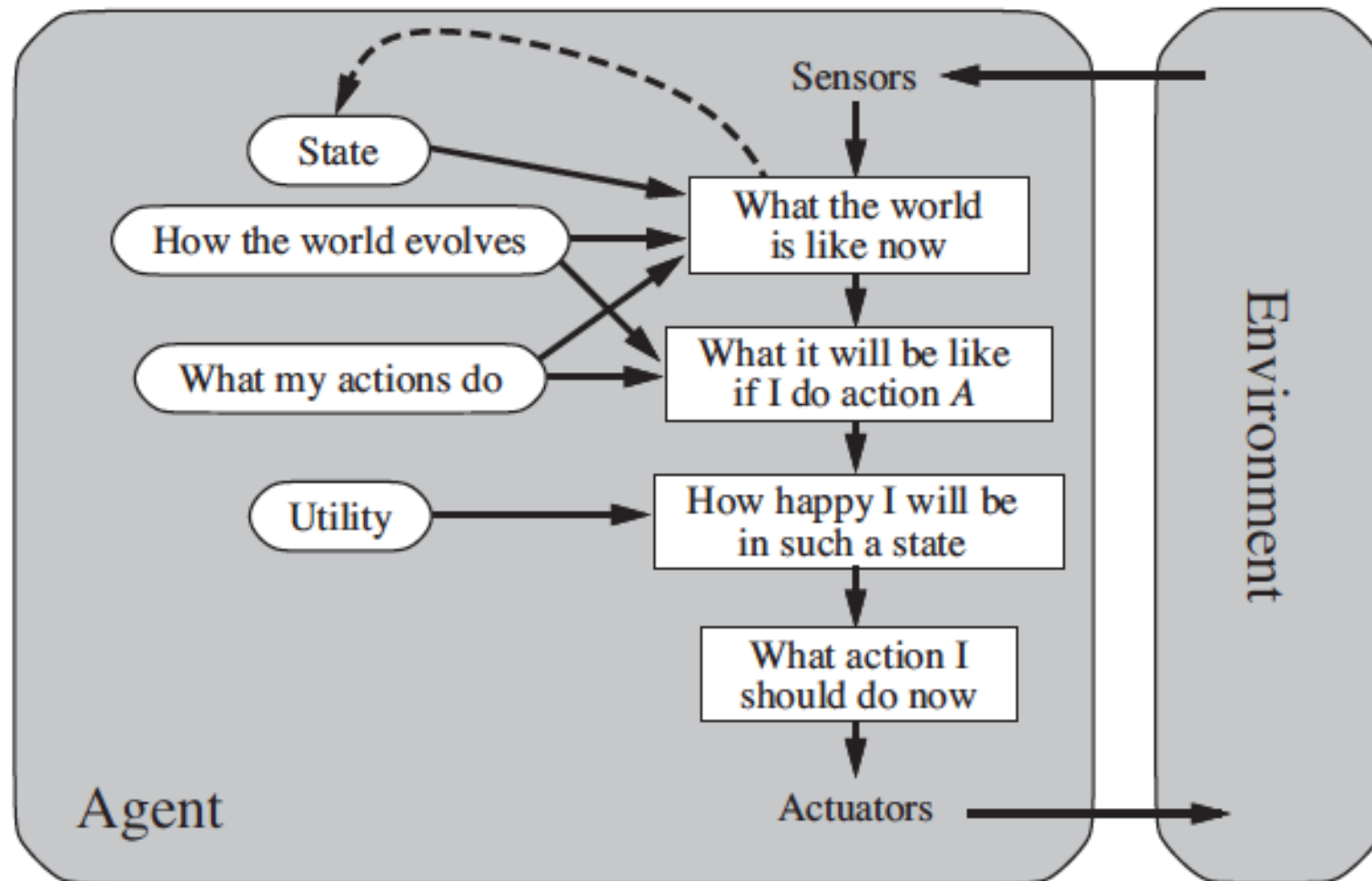
# AGENTI CON OBIETTIVO



# AGENTI CON OBIETTIVO

- Sono guidati da un obiettivo nella scelta dell'azione
  - A volte l'azione migliore dipende da qual è l'obiettivo da raggiungere (es. da che parte devo girare?).
  - Devono pianificare una sequenza di azioni per raggiungere l'obiettivo.
  - Meno efficienti ma più flessibili di un agente reattivo

# AGENTI CON VALUTAZIONE DI UTILITÀ

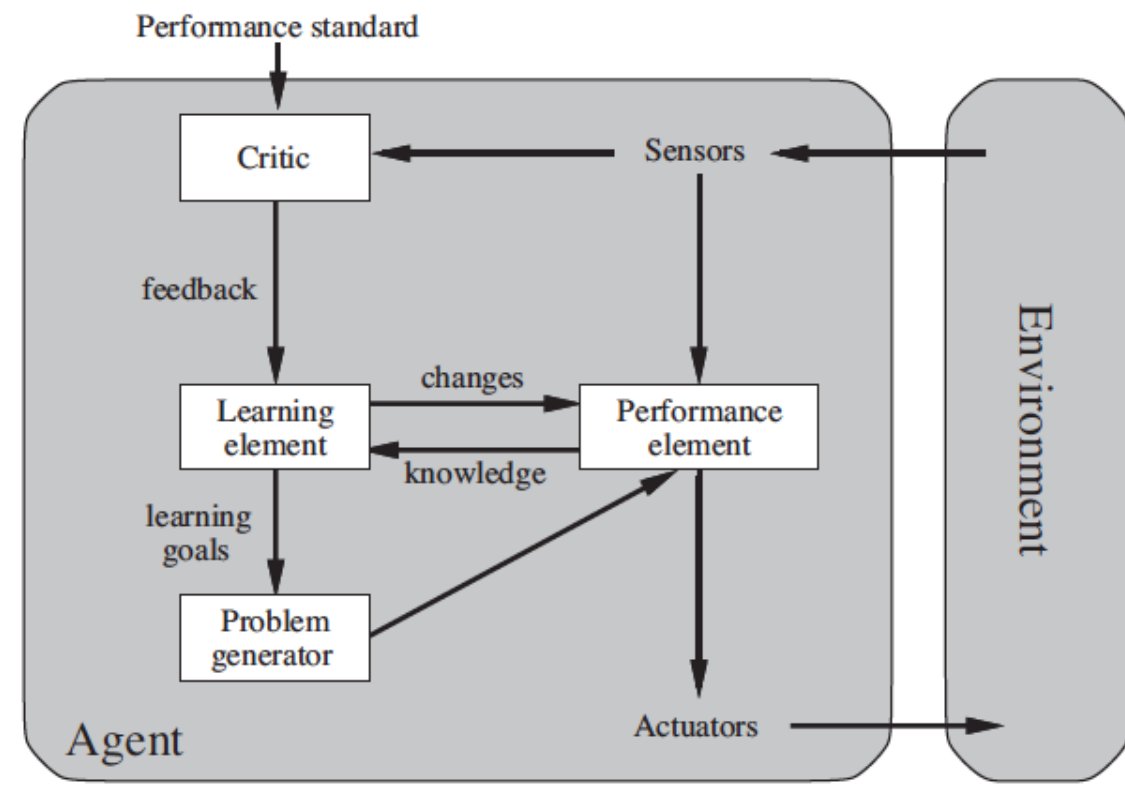


# AGENTI CON VALUTAZIONE DI UTILITÀ

- Obiettivi alternativi
  - l'agente deve decidere verso quali di questi muoversi.
  - necessaria una funzione di utilità (che associa ad uno stato obiettivo un numero reale).
- Obiettivi più facilmente raggiungibili di altri
  - la funzione di utilità tiene conto anche della probabilità di successo: utilità attesa

# AGENTI CHE APPRENDONO

- Componente di apprendimento
  - Produce cambiamenti al programma agente
- Elemento esecutivo
  - Il programma agente
- Elemento critico
  - Osserva e da feedback sul comportamento
- Generatore di problemi
  - Suggerisce nuove situazioni da esplorare



# PROBLEMI APERTI

- Come Rappresentare il modello?
- Come Rappresentare le soluzioni?
- Come tenere traccia delle azioni?
  - Mondi osservabili e deterministici
  - Mondo parzialmente osservabili o non deterministici

# LOGICA COME LINGUAGGIO DI RAPPRESENTAZIONE

MODELLI, SEMANTICA E RAGIONAMENTO AUTOMATICO





# ELEMENTI DI DATA MANAGEMENT

MODELLI RELAZIONALI DEI DATI, SISTEMI DI GESTIONE DELLE BASI DI DATI,  
RAGIONAMENTO E BASI DI CONOSCENZA

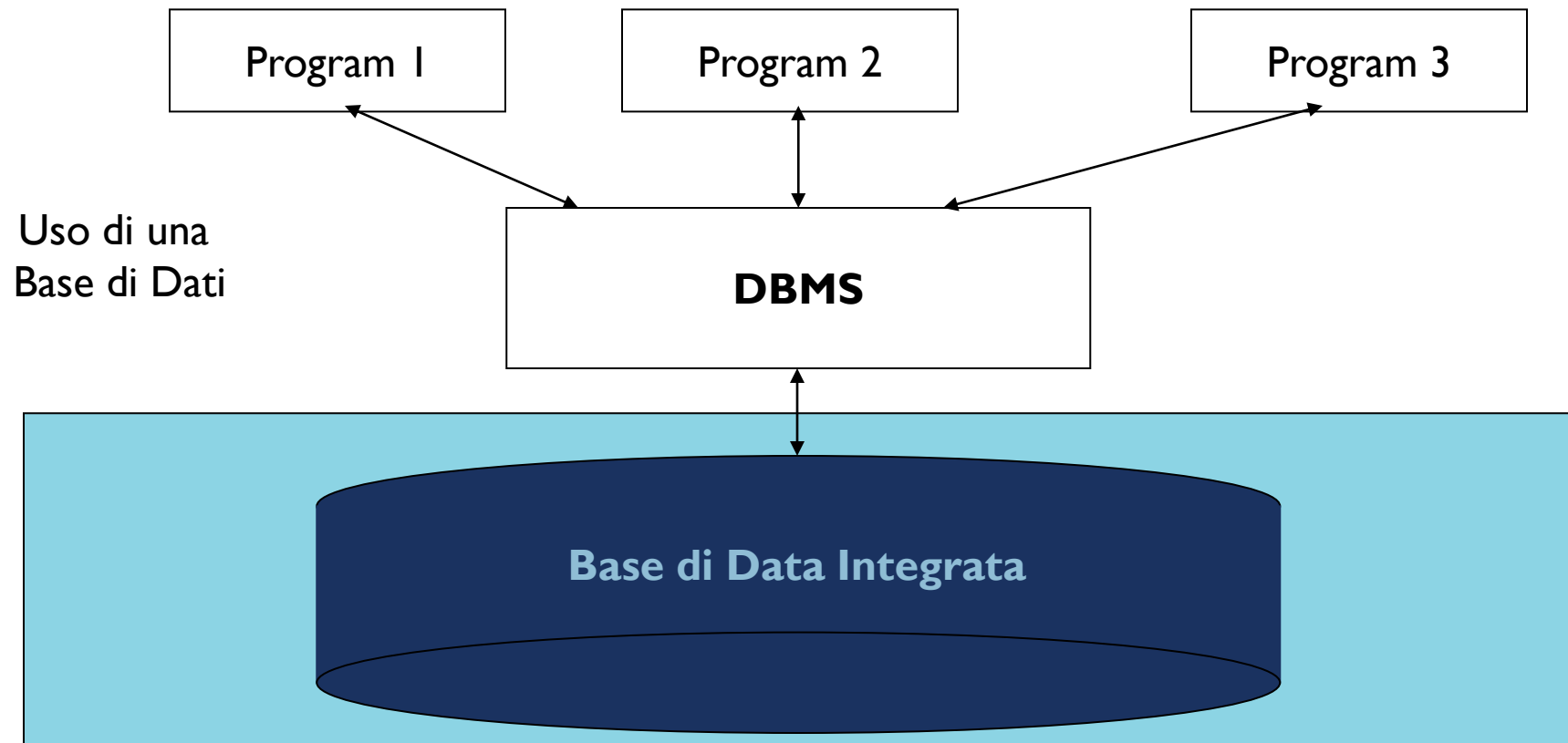




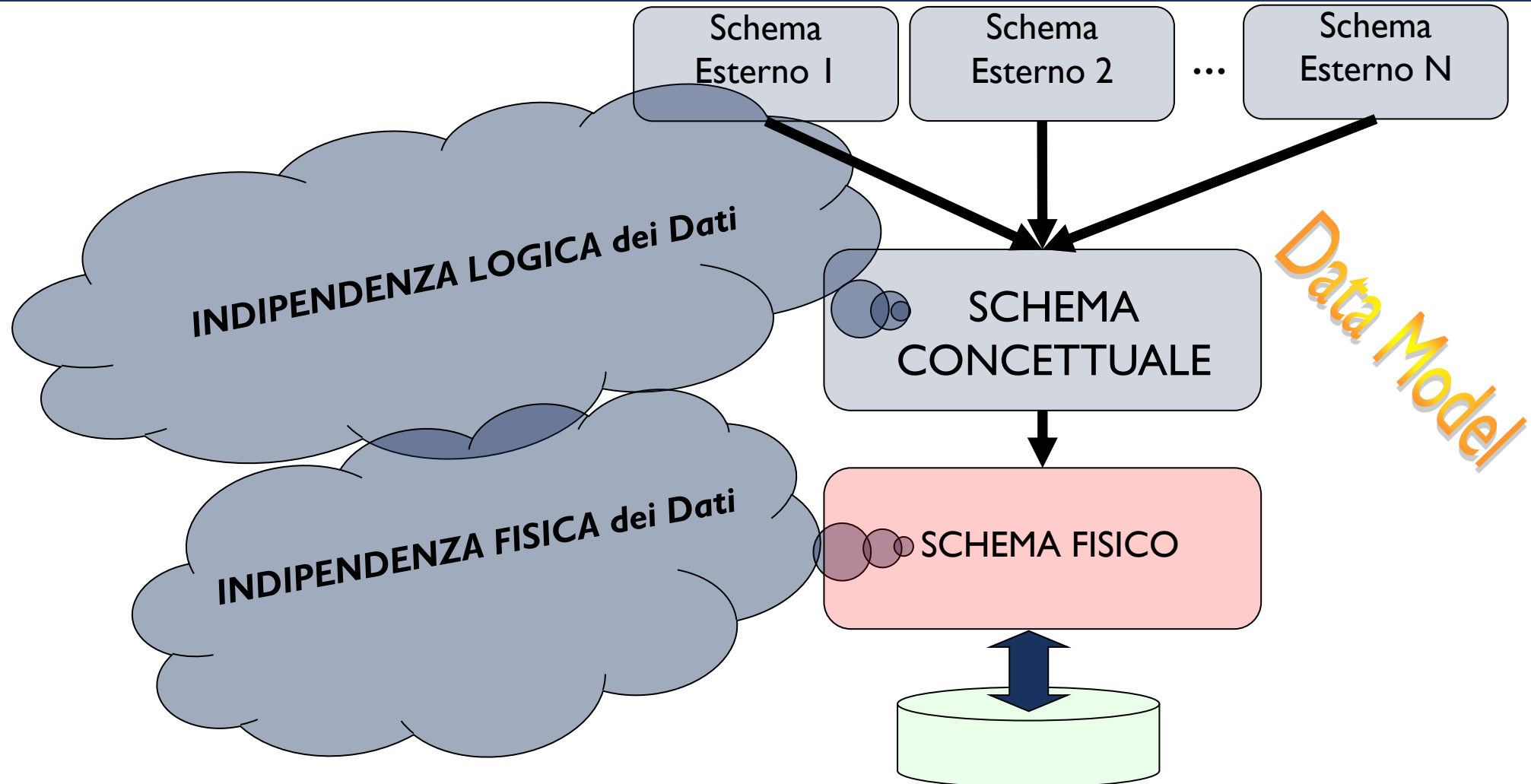
# DBMS

- Un **Data Base Management System (DBMS)** è un sistema software per lo storage, la gestione e la fornitura efficiente di dati secondo logiche eterogenee e verso applicazioni diverse.
- Vantaggi:
  - Indipendenza ed accesso efficiente ai dati
  - Riduzione del *tempo di sviluppo*
  - *Garanzia di Integrità e sicurezza* dei dati
  - Gestione degli accessi ai dati e Supporto all'*accesso concorrente*
  - Gestione delle *transazioni*
  - Gestione dei *Crash* e supporto al *Recovery*

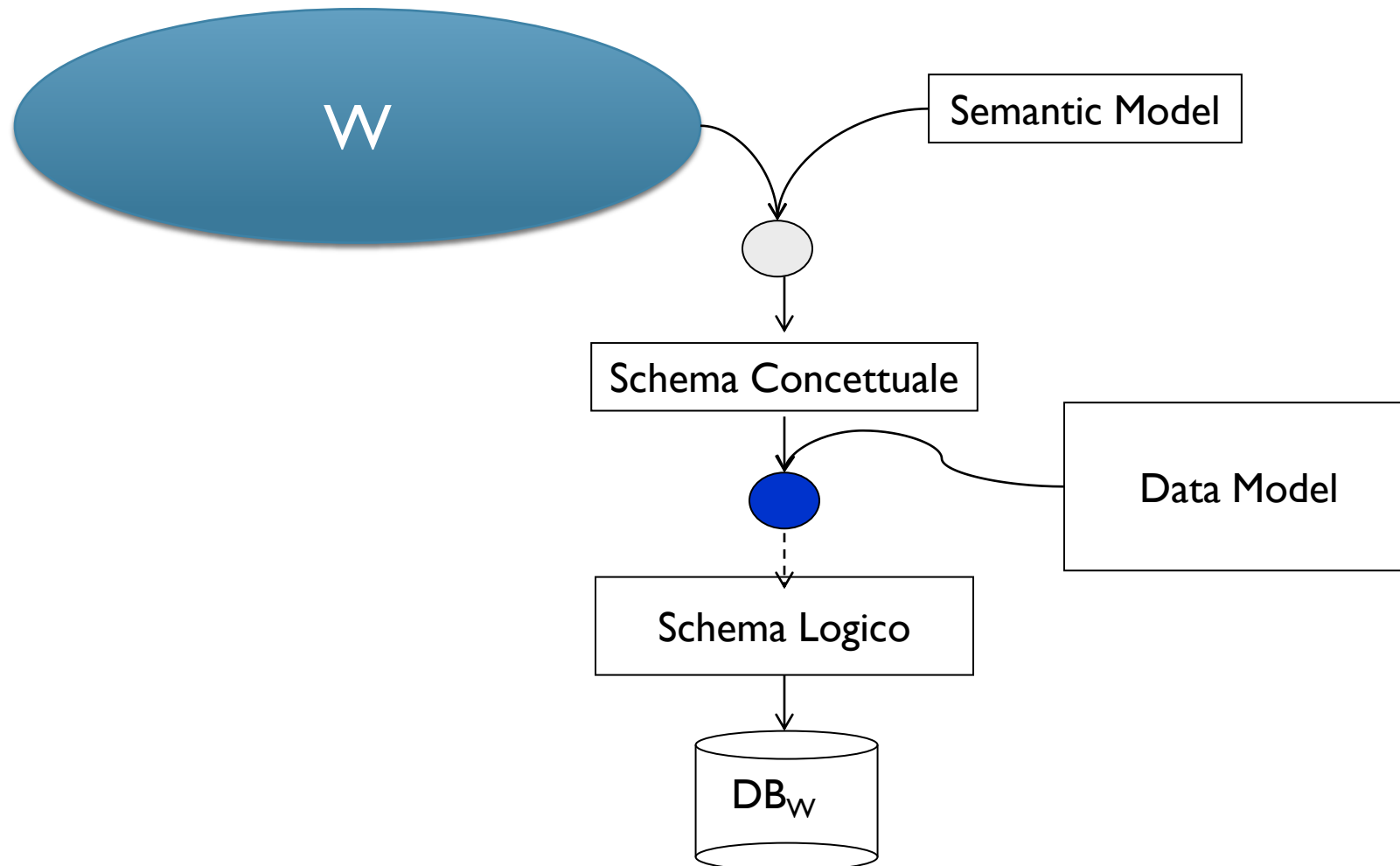
# ARCHIVI E BASI DI DATI



# LIVELLI DI ASTRAZIONE



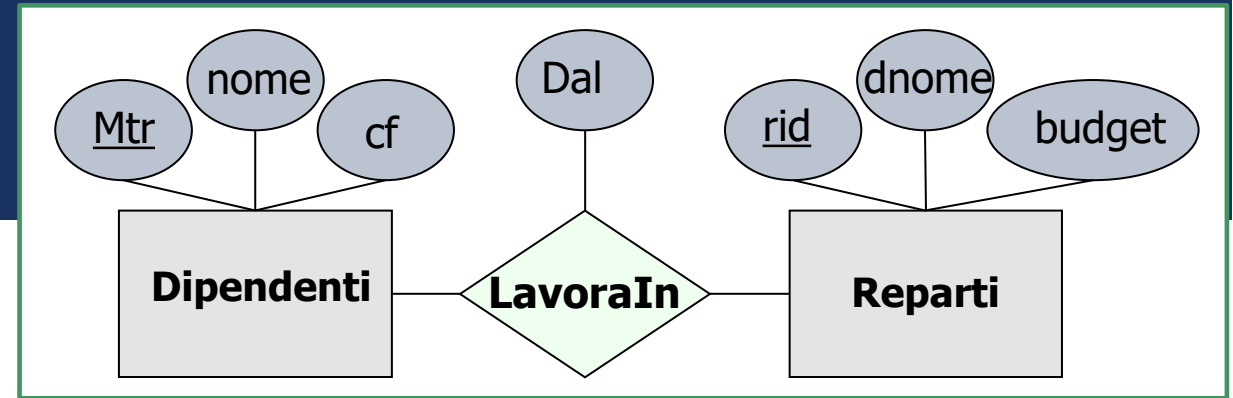
# PROGETTAZIONE CONCETTUALE(3)



# PROGETTAZIONE CONCETTUALE

- **Obbiettivi**
  - Quali sono le **entità** e le **relazioni** dell'organizzazione?
  - Quali **informazioni** su queste entità e relazioni dovrebbero essere memorizzate nella base di dati?
  - Quali sono i **vincoli** di integrità o le business rules in vigore?
- **Vantaggi**
  - Uno “schema” di base di dati nel modello ER può essere rappresentato graficamente (diagrammi ER)
  - Si può tradurre un diagramma ER in uno schema relazionale

## ELEMENTI DEL MODELLO ER



- ◆ ***Entità*** è un oggetto del mondo reale distinguibile da altri oggetti. E' descritta in un modello ER da un insieme di attributi e dal loro dominio.
- ◆ ***Insieme di Entità:*** Una collezione di entita' simili, per esse definisco un attributo chiave che le distingue.

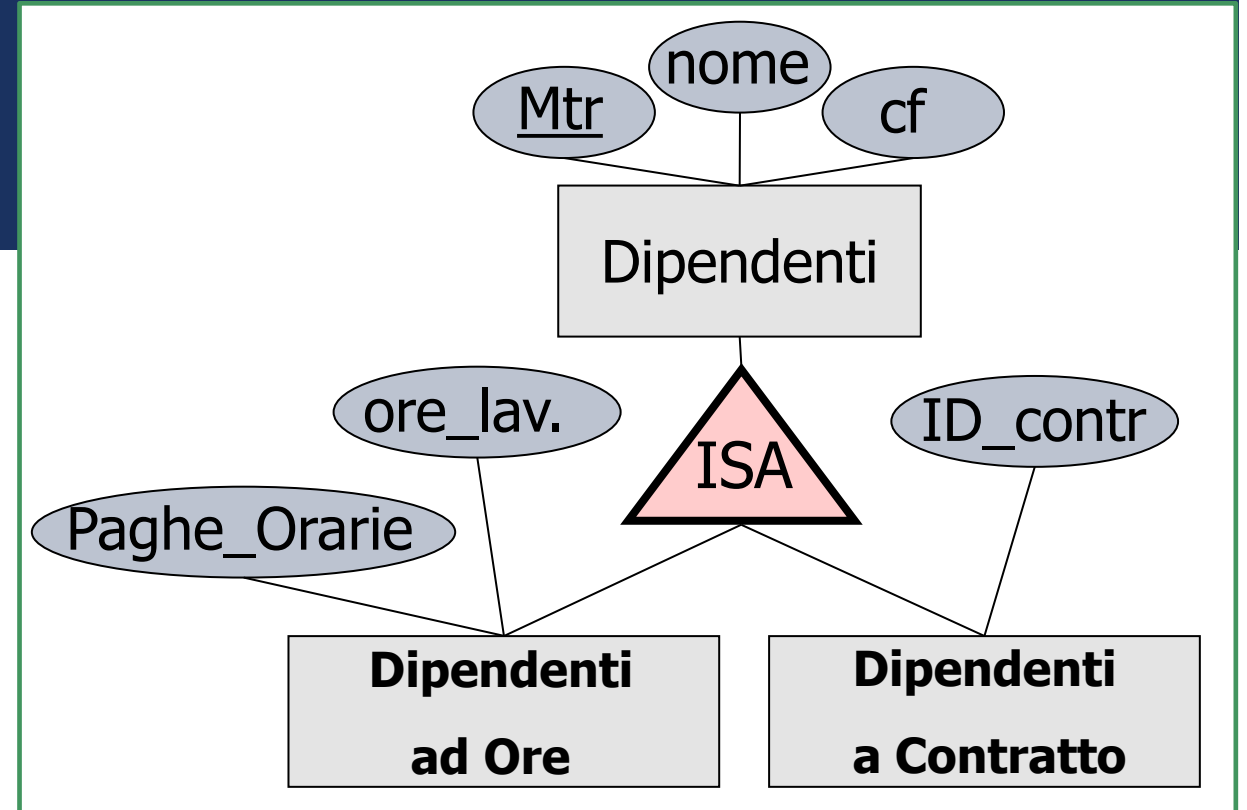
**Dipendenti**(Mtr:char(20),Nome:char(20),cf:char(20))

- ◆ ***Relazione:*** Un'associazione tra due o piu' entità
- ◆ ***Insieme di Relazioni:*** Una collezione di associazioni  
Ogni relazione e' individuata solo dalle entita' che vi partecipano

**LavoraIn**={(m,r) | m ∈ Dipendenti.Mtr, r ∈ Reparti.rid}

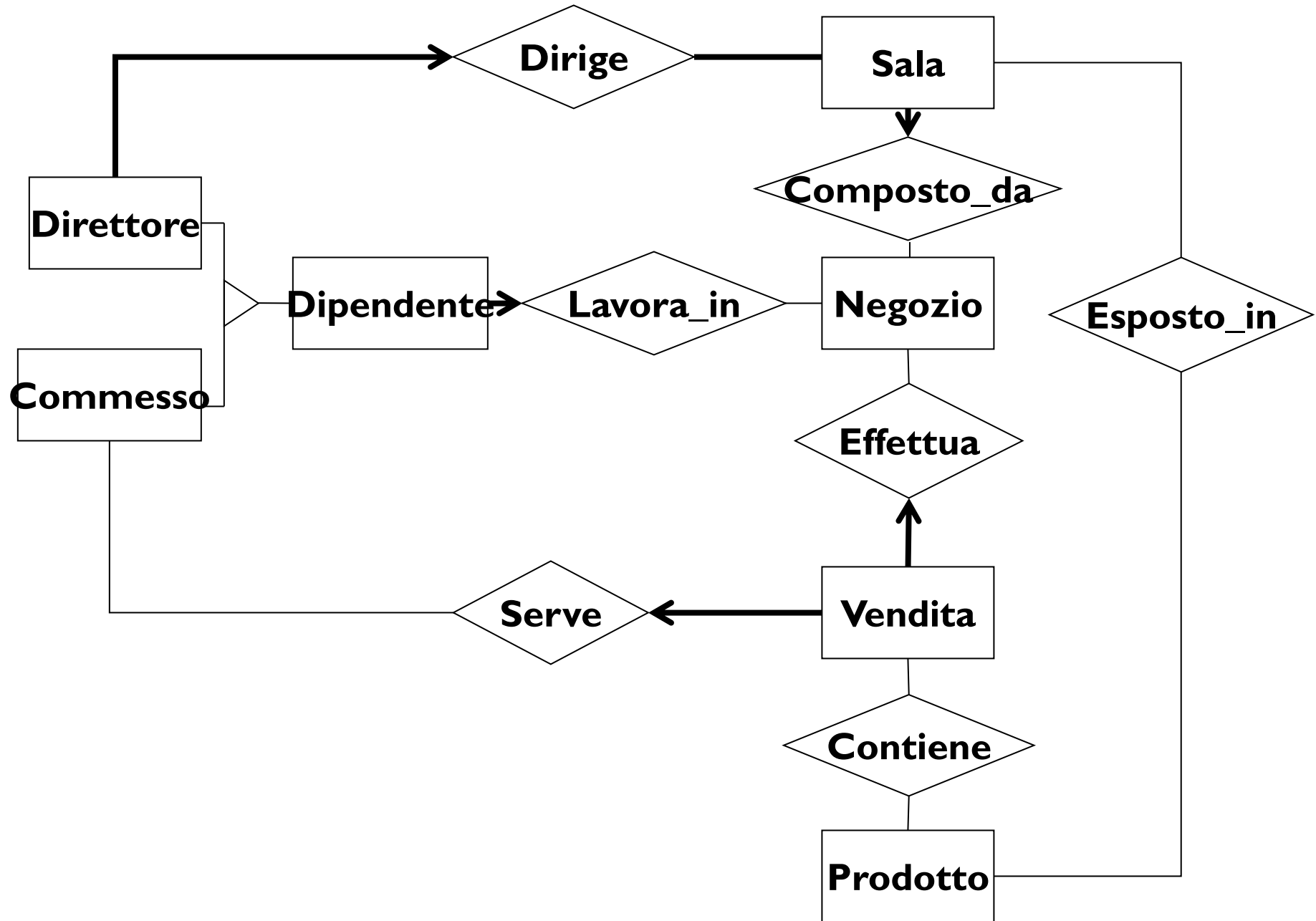
# GERARCHIE ISA

Legame logico tra un insieme entità E detto padre e più insiemi entità  $E_1, E_2, \dots, E_n$  dette figlie.



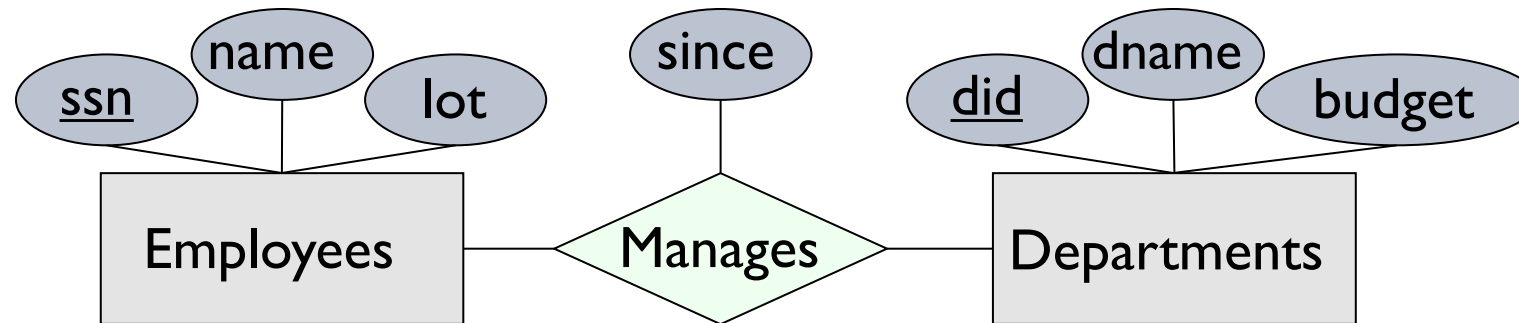
**Vincolo di Sovrapposizione:** sono permessi overlap tra le entità presenti negli insiemi entità figlie  $E_1, E_2, \dots, E_n$

**Vincolo di Copertura:** ogni entità in  $E$  deve essere presente in almeno un insieme delle entità figlie





# DISEGNO DELLO SCHEMA LOGICO- DALLO SCHEMA ER AL DATO RELAZIONALE



```
CREATE TABLE Employees
(ssn CHAR(11),
name CHAR(20),
lot INTEGER,
PRIMARY KEY (ssn) )
```

```
CREATE TABLE Departments
(did INTEGER,
dname CHAR(20),
budget REAL,
PRIMARY KEY (did) )
```

# ESEMPIO: RELAZIONE STUDENTI



## Istanza

sid	nome	login	età	gpa
0012	Rossi	<u>ro@ec</u>	18	3.4
0072	Bianchi	<u>bi@ec</u>	19	3.2
0033	Bianchi	<u>bi@tt</u>	18	3.8

- cardinalità 3
- grado 5

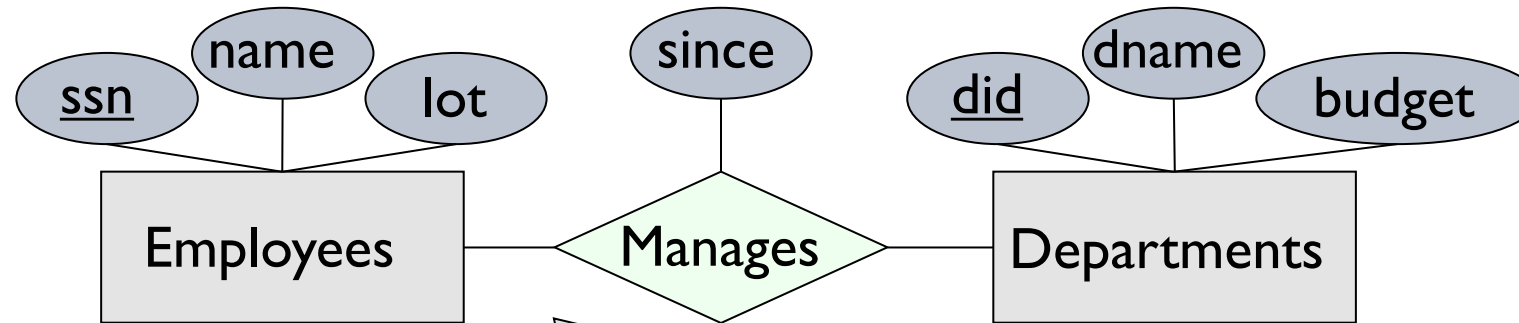
## Schema

Studenti (

```
sid:string,  
nome:string,  
login:string,  
età:integer,  
gpa:real)
```


- Non è definito alcun ordinamento tra le  $n$ -uple/righe
- Le  $n$ -uple/righe di una relazione sono distinte
  - Due  $n$ -uple uguali (per tutti i valori) sono LA STESSA  $n$ -pla
- Ciascuna  $n$ -upla è ordinata al suo interno; all' $i$ -esimo valore corrisponde l' $i$ -esimo dominio definito nello schema (Constraint/Vincolo di Dominio)

# DISEGNO DELLO SCHEMA LOGICO - MAPPATURA DELLE RELAZIONI




```
CREATE TABLE Manages
(ssn CHAR(11),
 did INTEGER,
 since DATE,
 PRIMARY KEY (ssn, did),
 FOREIGN KEY (ssn) REFERENCES Employees,
 FOREIGN KEY (did) REFERENCES Departments)
```

# IL LINGUAGGIO DI INTERROGAZIONE SQL



sid	nome	login	eta'	gpa
0012	Rossi	<u>ro@ec</u>	18	3.4
0072	Bianchi	<u>bi@ec</u>	19	3.2
0033	Bianchi	<u>bi@tt</u>	18	3.8

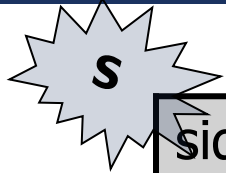
```
SELECT *  
FROM Studenti S  
WHERE S.eta'=18
```



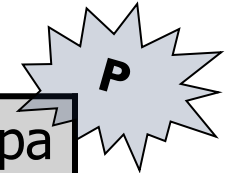
sid	nome	login	eta'	gpa
0012	Rossi	<u>ro@ec</u>	18	3.4
0033	Bianchi	<u>bi@tt</u>	18	3.8

Estrae dalla tabella S tutti gli studenti diciottenni!

# IL LINGUAGGIO DI INTERROGAZIONE SQL



sid	nome	login	eta'	gpa
0012	Rossi	<u>ro@ec</u>	18	3.4
0072	Bianchi	<u>bi@ec</u>	19	3.2
0033	Bianchi	<u>bi@tt</u>	18	3.8



sid	cid	grade	gpa
0012	Mate1	C	3.4
0012	Bd1	A	3.2
0033	Bd1	A	3.8

```
SELECT S.nome, P.cid
FROM Studenti S,
      pianoDS P
WHERE S.sid=P.sid
AND P.grade='A'
```



S.nome	P.cid
Rossi	Bd1
Bianchi	Bd1

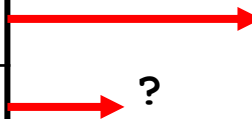
Vincolo di integrità referenziale

# SQL: FOREIGN KEY

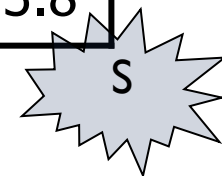
```
CREATE TABLE PianoDS
(sid: CHAR(20),
cid: CHAR(20),
grade CHAR(2),
PRIMARY KEY (sid,cid),
FOREIGN KEY(sid) REFERENCES Studenti(SSID))
```



sid	cid	grade
0012	Mate1	C
<del>0011</del>	<del>Bd1</del>	<del>A</del>
0033	Bd1	B



SSID	nome	login	eta'	gpa
0012	Rossi	<u>ro@ec</u>	18	3.4
0072	Bianchi	<u>bi@ec</u>	19	3.2
0033	Bianchi	<u>bi@tt</u>	18	3.8



Cosa accade se si tenta di inserire in **PianoDS** una tupla con sid non presente nella tabella **Studenti**?

E se viene cancellata una tupla di **S** utilizzata da **PianoDS**?



# LOGICA COME LINGUAGGIO DI RAPPRESENTAZIONE

MODELLI RELAZIONALI E TEORIE LOGICHE: RAGIONAMENTO E BASI DI CONOSCENZA



# OVERVIEW

- La necessità di formalizzare informazioni di dominio:
  - Dal modello del mondo alla conoscenza per agire e prendere decisioni
- Inferenza e conoscenza
  - Ontologia: tipi generali (classi/categorie), entità e relazioni
  - Decision-making: decisioni su istanze della realtà proiezioni dei tipi generali
- Un formalismo per la conoscenza: le Reti semantiche
  - Semantica nelle reti semantiche
- Dati e Reti semantiche nel Web: il Semantic Web
- Dai grafi di Conoscenza del Web alle rappresentazioni analitiche (vettoriali)



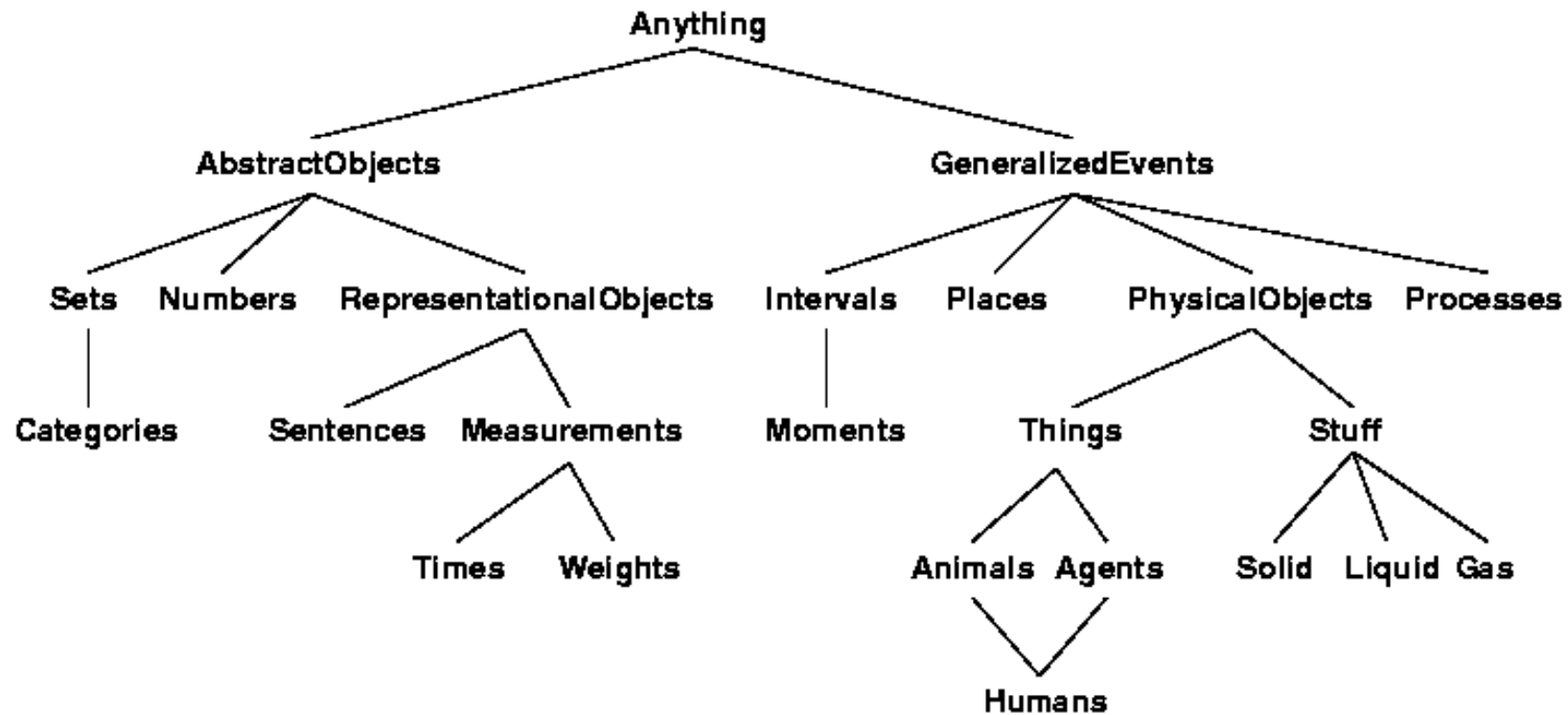
# INGEGNERIA DELLA CONOSCENZA

- Tra i vari problemi di rappresentazione
  - le ontologie generali e le categorie
  - oggetti fisici e sostanze
  - azioni e cambiamento
  - eventi, tempo
  - conoscenze, credenze ... attitudini mentali
  - ... il ragionamento non monotono, incerto, o probabilistico

## Focus:

Il ragionamento su categorie (ontologie e reti semantiche),  
Le risorse enciclopediche come esempi di KB e Lessici

# ONTOLOGIA GENERALE O SUPERIORE



# APPROCCIO PSICOLOGICO-LINGUISTICO ALLA R.C.

- L'approccio *logico*: per formalizzare il ragionamento valido
  - nato per la matematica e poi esteso al ragionamento di "senso comune".
- L'approccio *cognitivo-linguistico*: pone l'enfasi sui meccanismi per l'acquisizione, strutturazione ed uso della conoscenza
- Forti sinergie con:
  - Studi sul linguaggio naturale
  - Filosofia del linguaggio e Epistemologia
  - Studi di psicologia cognitiva

# RAPPRESENTAZIONI A GRAFO

- Precursore: grafi esistenziali di Charles Peirce (1909)
- In logica i simboli sono manipolati sintatticamente a prescindere dal loro significato
  - $\forall x \text{ Fragola}(x) \Rightarrow \text{Rossa}(x)$
- La verità di una formula dipende unicamente dalla verità delle sue sotto-formule ma non dal significato dei simboli né dalle relazioni semantiche tra i simboli

# ORGANIZZAZIONE GERARCHICA DEI CONCETTI: ESPERIMENTI (COLLINS, QUILLIAN, 1969)

- Date le seguenti domande:

1. "Un canarino è un uccello?"

2. "Un canarino vola?"

3. "Un canarino respira?"

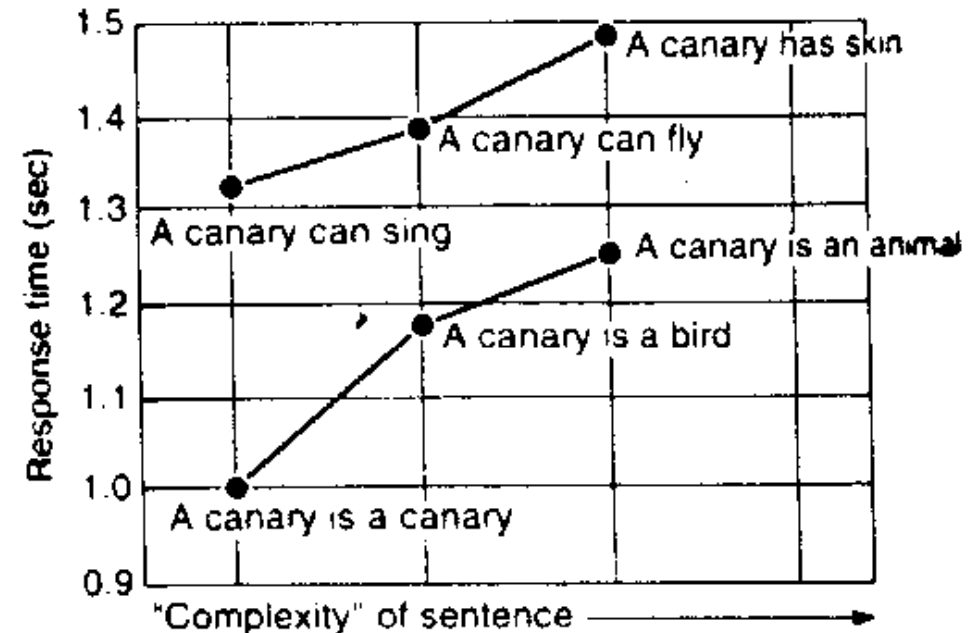
- Si osserva che i tempi di risposta  $T_i$  sono:

$$T_1 < T_2 < T_3$$

- Eccezioni a questa situazione sono risposte a domande quali:

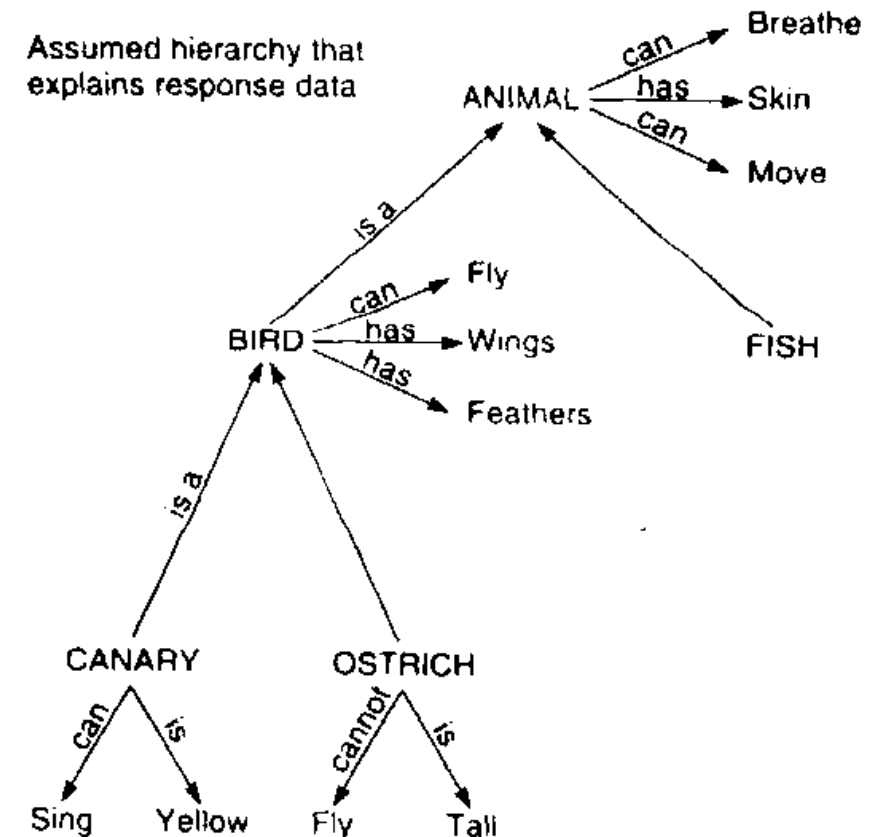
- "Uno struzzo vola?"

In cui la risposta è immediata



# ORGANIZZAZIONE GERARCHICA DEI CONCETTI: INTERPRETAZIONE

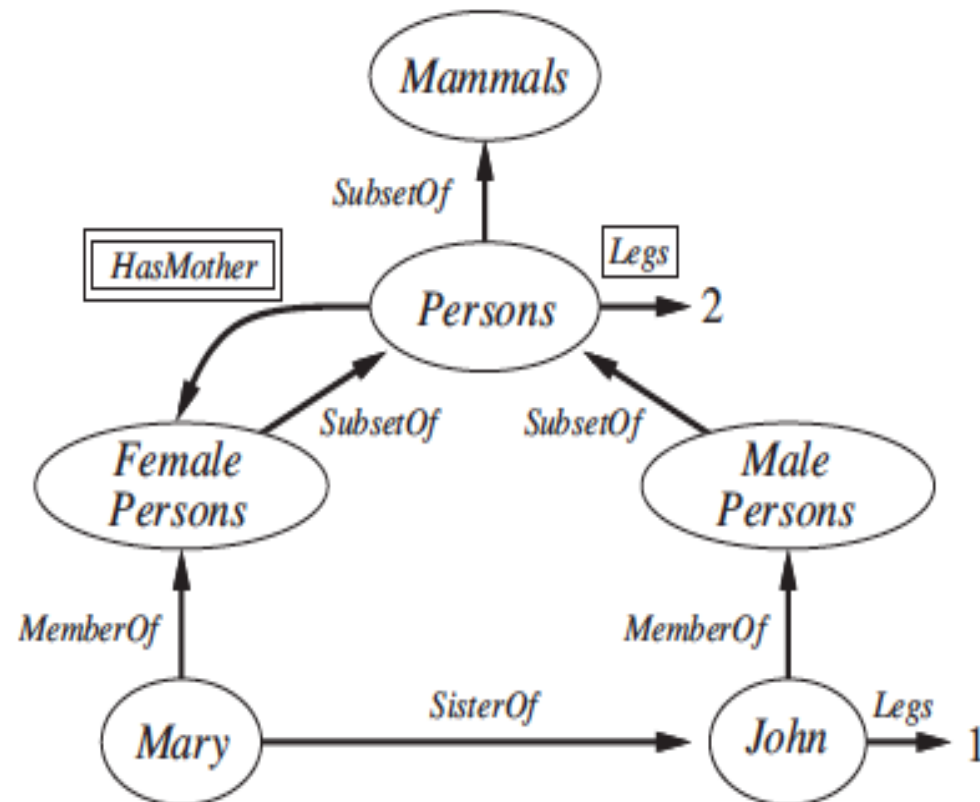
- Le proprietà sono "legate" al concetto più generale a cui si applicano
- Trattamento eccezioni
  - Le eccezioni sono memorizzate direttamente con l'oggetto
- Il successo della strutturazione gerarchica è anche confermato dalle tecniche di progettazione ad oggetti largamente influenti oggi nell'ingegneria del SW



# DEFINIZIONE DI RETE SEMANTICA

- Le reti semantiche sono una grande famiglia di schemi di rappresentazione "a grafo".
- Una **rete semantica** è un grafo in cui:
  - I nodi, etichettati, corrispondono a concetti (individui o classi)
  - Gli archi, etichettati e orientati, a relazioni binarie tra concetti (dette anche ruoli).
- Due particolari relazioni “primitive” sono sempre presenti:
  - IS, relazione di sotto-classe ( $\subseteq$ )
  - IS-A, relazione di appartenenza ( $\in$ )

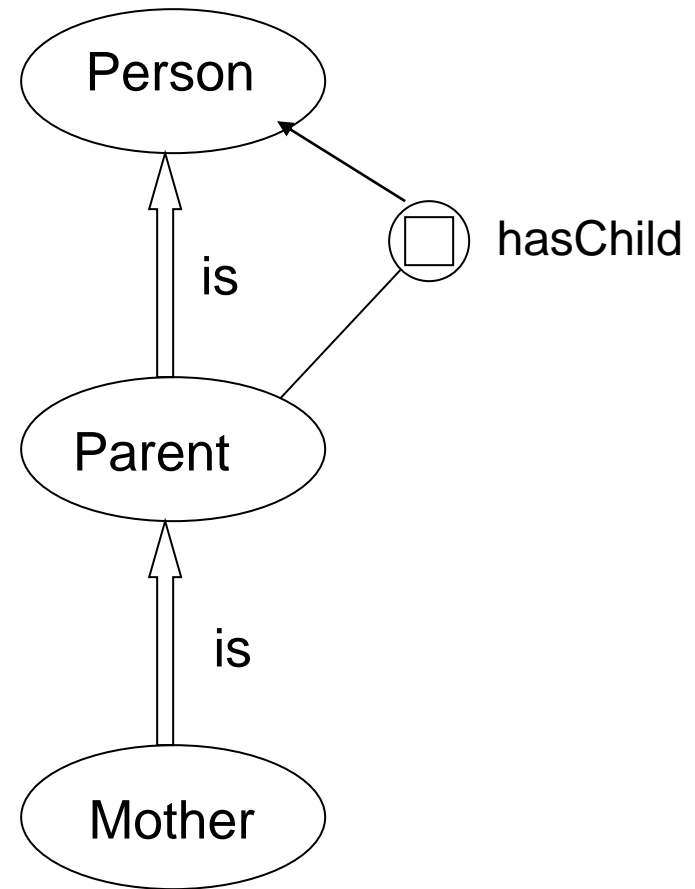
# UN ESEMPIO DI RETE SEMANTICA





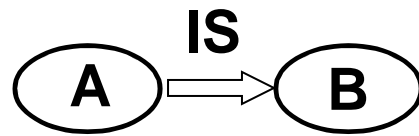
# EREDITARIETÀ NELLE RETI SEMANTICHE

- *Ereditarietà* come una particolare inferenza legata alla transitività di IS
- Facilmente implementabile come *link traversal*
- Ereditarietà **multipla**



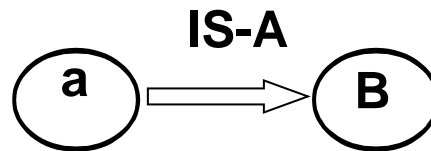
# TRADUZIONE IN LOGICA

- Le reti semantiche costituiscono una *notazione conveniente* per una parte della First Order Logic, ma sono pur sempre riconducibili ad un formalismo logico
- Alcuni aspetti però sono complessi da trasformare in una pura forma logica, poiché dipendono da aspetti extra-logici o costituiscono informazioni procedurali

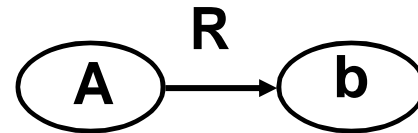


NOTA: In maiuscolo le classi, in minuscolo gli individui

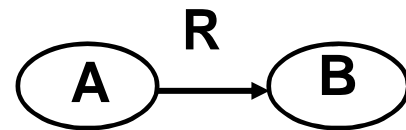
$$\forall x A(x) \Rightarrow B(x)$$



$$B(a)$$

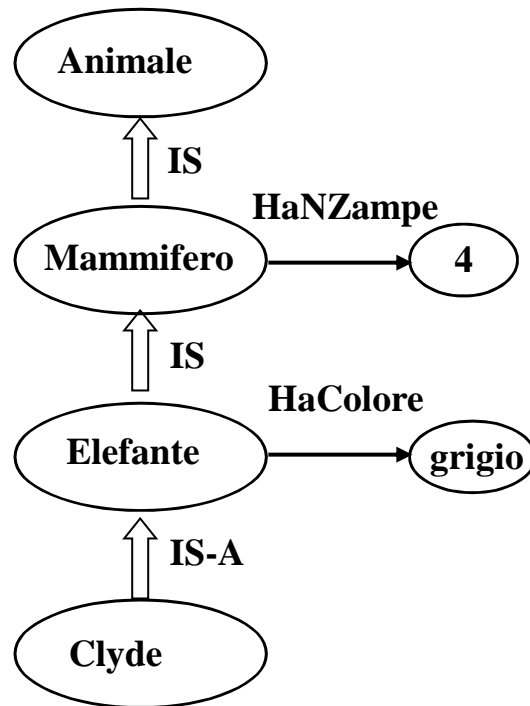


$$\forall x x \in A \Rightarrow R(x, b)$$



$$\forall x x \in A \Rightarrow \exists y y \in B \wedge R(x, y)$$

# UN ESEMPIO DI TRADUZIONE



$\forall x \text{ Mammifero}(x) \Rightarrow \text{Animale}(x)$

$\forall x \text{ Mammifero}(x) \Rightarrow \text{HaNZampe}(x, 4)$

$\forall x \text{ Elefante}(x) \Rightarrow \text{Mammifero}(x)$

$\forall x \text{ Elefante}(x) \Rightarrow \text{HaColore}(x, \text{grigio})$

$\text{Elefante}(\text{Clyde})$

È possibile dedurre:

$\text{Animale}(\text{Clyde})$

$\text{Mammifero}(\text{Clyde})$

$\text{HaNZampe}(\text{Clyde}, 4)$

$\text{HaColore}(\text{Clyde}, \text{grigio})$

L'*ereditarietà* scaturisce dalla semantica del quantificatore  $\forall E$ , del MP e dalla transitività della relazione  $\Rightarrow$

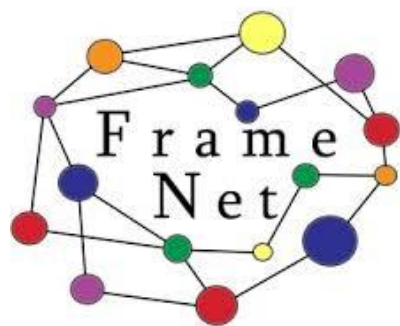
# ESEMPI DI SEMANTIC NETWORKS

PRINCETON UNIVERSITY

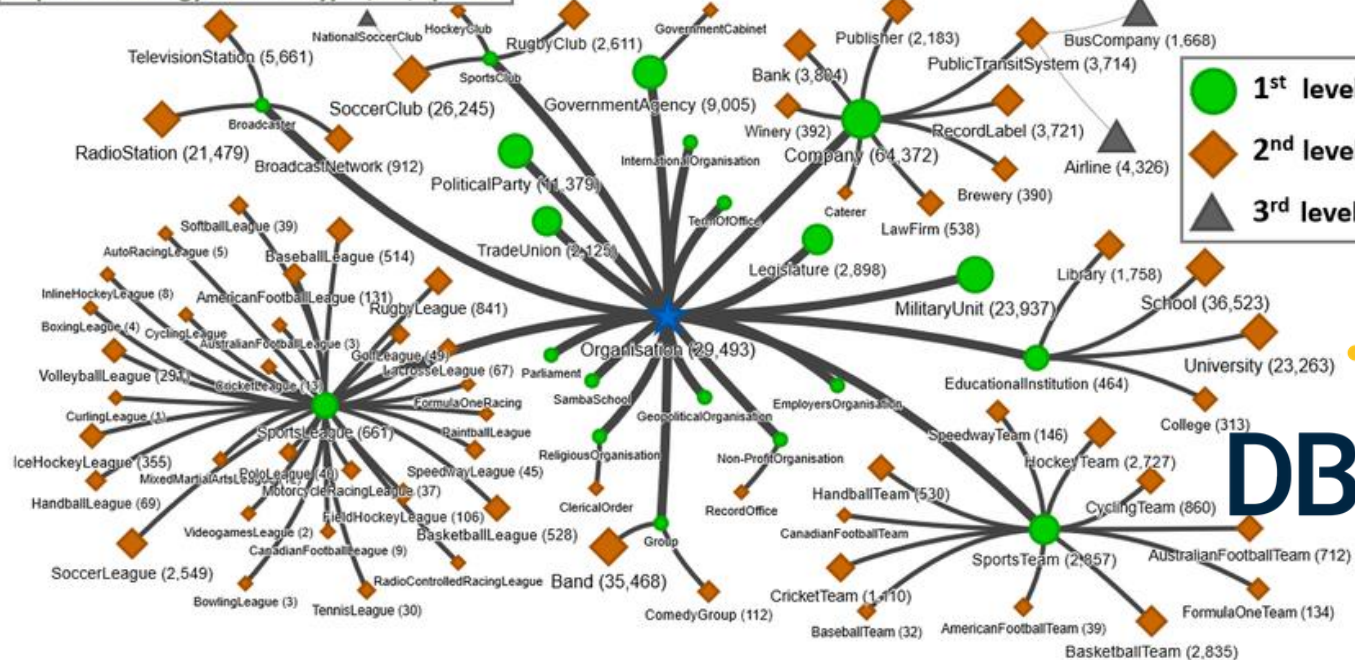
**WordNet**  
A lexical database for English



**WIKIDATA**



DBpedia Ontology instance types, en, specific



**DBpedia**

# SCOPI DI UN KG (GKG)

Miglioramenti ai processi Web, ad es. Google Search, in 3 dimensioni:

- *Find the right thing*

- Gestire la ambiguità delle lingue (cosa indica il nome proprio Taj Mahal? Un monumento? Un musicista?)

The screenshot shows a Google search for "taj mahal". The search bar contains "taj mahal" and the results are displayed below. On the left, there are navigation tabs for "Everything", "Images", "Maps", "Videos", "News", "Shopping", and "More". The "Everything" tab is selected. The search results are organized into several sections:

- Everything:** Lists search results for "Taj Mahal - Wikipedia, the free encyclopedia" (the monument), "Taj Mahal (musician) - Wikipedia, the free encyclopedia" (the musician), and "Atlantic City New Jersey Casino Hotels | Trump Taj Mahal, Atlantic...".
- Images:** Shows a grid of images related to the Taj Mahal monument.
- Maps:** Displays a map of the Taj Mahal monument in Agra, India, with a red pin and a street view image.
- Knowledge Panel:** Provides detailed information about the Taj Mahal monument, including its height (561 feet), opening time (1648), address, architectural style (Mughal architecture), and architect (Ustad Ahmad Lahauri).
- People also search for:** Lists related search terms like "Agra Fort", "Great Wall of China", and "Desai F...".
- See results about:** A section that highlights the ambiguity of the search term, showing results for "Taj Mahal Musician" (Henry Saint Clair Fredericks) and "Trump Taj Mahal Casino Resort".

# SCOPI DI UN KG (GKG)

## ■ *Summaries*

- Sintetizzare in forma strutturata i contenuti delle pagine Web

The image shows a Google search for "Marie Curie". The search results include a Wikipedia entry, a biography from Nobelprize.org, and a list of images. A structured summary overlay is positioned on the right side of the page, providing a concise overview of Marie Curie's life and achievements.

### Marie Curie

Marie Skłodowska-Curie was a French-Polish physicist and chemist famous for her pioneering research on radioactivity. She was the first person honored with two Nobel Prizes—in physics and chemistry. [Wikipedia](#)

**Born:** November 7, 1867, **Warsaw**  
**Died:** July 4, 1934, **Sancellemoz**  
**Spouse:** Pierre Curie (m. 1895–1906)  
**Children:** Irène Joliot-Curie, Ève Curie  
**Discovered:** Radium, Polonium  
**Education:** École Supérieure de Physique et de Chimie Industrielles de la Ville de Paris, University of Paris

**People also search for**

-  Albert Einstein
-  Pierre Curie
-  Ernest Rutherford
-  Louis Pasteur
-  John Dalton

[Report a problem](#)

# SCOPI DI UN KG (GKG)

## ■ *Informazione a più alti livelli di astrazione e più ampia*

- Suggestire nuovi fatti (derivati tramite inferenza)
- Anticipare le domande successive di un utente e fornire l'informaizione in modo natural e tempestivo (sulla base della storia delle domande dell'utente e della storia delle domande di tutti gli altri utenti)

Circa 5.540.000 risultati (0,48 secondi)



**Luciano Spalletti**

Allenatore di calcio

Panoramica

Video



Wikipedia

[https://it.wikipedia.org/wiki/Luciano\\_Spalletti](https://it.wikipedia.org/wiki/Luciano_Spalletti)

### Luciano Spalletti

**Luciano Spalletti** (Certaldo, 7 marzo 1959) è un allenatore di calcio italiano, commissario tecnico della nazionale italiana.

[Certaldo](#) · [Aurelio Andreazzoli](#) · [Virtus Entella](#) · [Unione Calcio Cuioiopelli](#)

Notizie principali :

### Informazioni

Luciano Spalletti è un allenatore di calcio ed ex calciatore italiano, commissario tecnico della nazionale italiana. Terminata la carriera da calciatore nel 1993, ha intrapreso il percorso di allenatore. [Wikipedia](#)

**Nascita:** 7 marzo 1959 (età 64 anni), [Certaldo](#)

**Allenatore:** [Nazionale di calcio dell'Italia](#)

**Squadre allenate:** [Nazionale di calcio dell'Italia](#) (Allenatore di calcio, dal 2023), [ALTRO](#)

**Figli:** [Samuele Spalletti](#), [Federico Spalletti](#)

**Fratelli e sorelle:** [Marcello Spalletti](#)

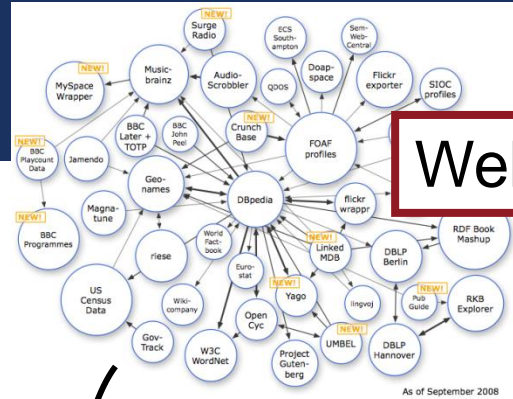
**Altezza:** 1,8 m

**Coniuge:** [Tamara Spalletti](#)

[Rivendica scheda informativa](#)

[Feedback](#)

# WEB OF DATA



Web of Data

Semantic Web



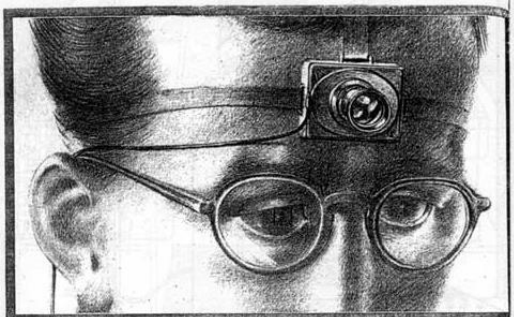
Web

Semantic Annotations

Hypermedia

Hypertext

?

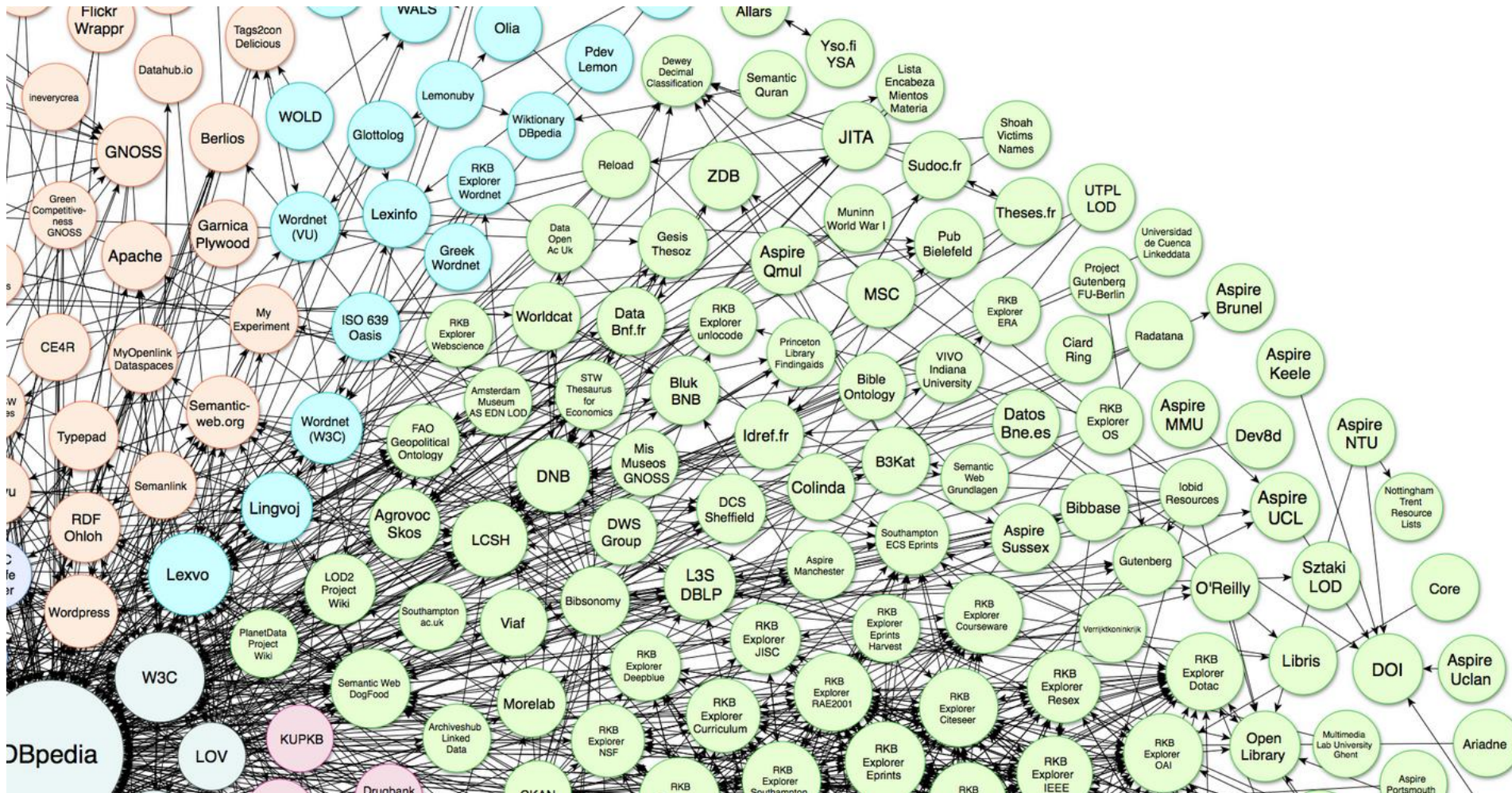


“As We May Think”, 1945

Picture from <http://www.theatlantic.com/doc/194507/bush>



# LINKED OPEN DATA ([HTTPS://LOD-CLOUD.NET/#ABOUT](https://lod-cloud.net/#about))



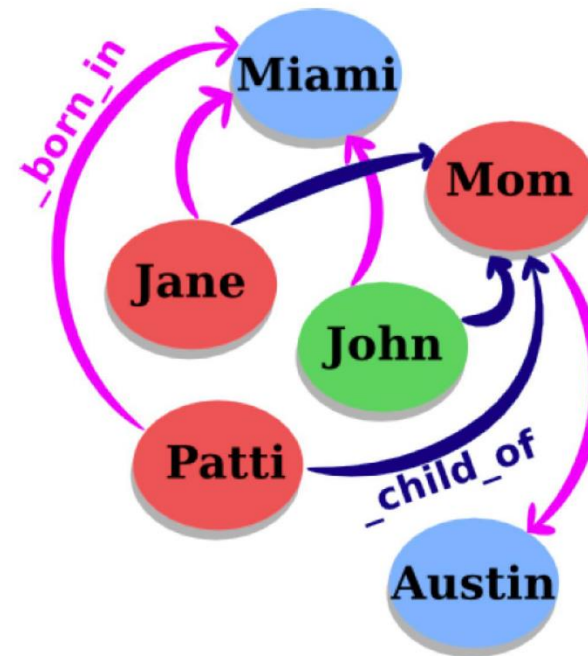
# KG:TRIPLE (HEAD, RELATION, TAIL)

## Conoscenza in grafi:

- Nodi  $\leftrightarrow$  entità (*Miami, Jane, ...*)
- Archi  $\leftrightarrow$  relazioni (`_BORN_IN`, `_CHILD_OF`)

## Fatti: (*head, relation, tail*)

- head=subject entity
- relation=relation type
- tail=object entity

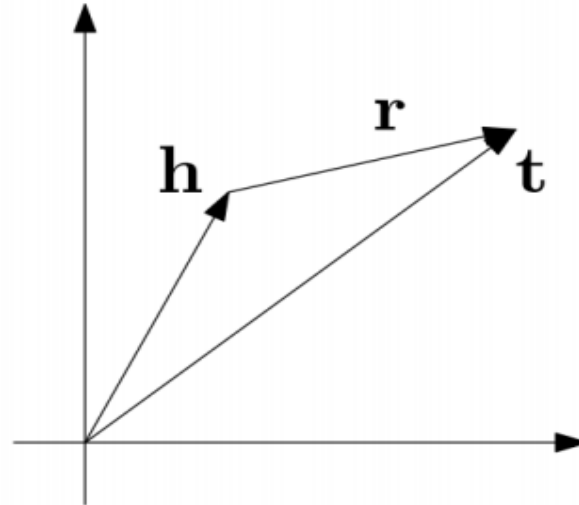


## KG esistenti (esemplificativi):

- WordNet: Linguistic KG (lessico computazionale)
- WikiData (World DBs, URL: [https://www.wikidata.org/wiki/Wikidata:Main\\_Page](https://www.wikidata.org/wiki/Wikidata:Main_Page))

# DALLA LOGICA ALLE RAPPRESENTAZIONI QUANTITATIVE: INFERENZA GEOMETRICA

Per ogni tripla (*head*, *relation*, *tail*), la relazione corrisponde ad una *mappatura* da una *head* alla corrispondente *tail*



L'obiettivo del learning nello sviluppo delle basi di conoscenza di larga scala è l'apprendimento della logica (vettoriale) delle relazioni *r* in modo che al variare di  $\underline{h}$  e  $\underline{t}$  si abbia sempre:

$$\underline{h} + \underline{r} = \underline{t}$$

# PERCHÉ LA LOGICA E' UTILE?

- Perché è rigorosa ed un programma dotato di sole formule consistenti con la realtà, non farà mai errori se usato in modo deduttivo
- E' modulare, poiché le parti di teorie assiomatiche sono RIUSABILI
- E' dichiarativo e, come nelle basi di dati, semplice da interpretare
- E' fornito di procedure automatiche per la deduzione, quindi da uno stato del mondo e dagli assiomi si può dedurre tramite la conseguenza logica lo stato raggiungibile, e giudicarne la utilità
  - Ragionamento e Azione basati sul modello
  - Obiettivo come funzione degli stati
  - Pianificazione come dimostrazione automatica di teoremi: scegli l'azione la cui conseguenza logica, cioè lo stato raggiungibile, massimizza il vantaggio
  - Mondi possibili

# SISTEMI BASATI SU CONOSCENZA: LIMITI

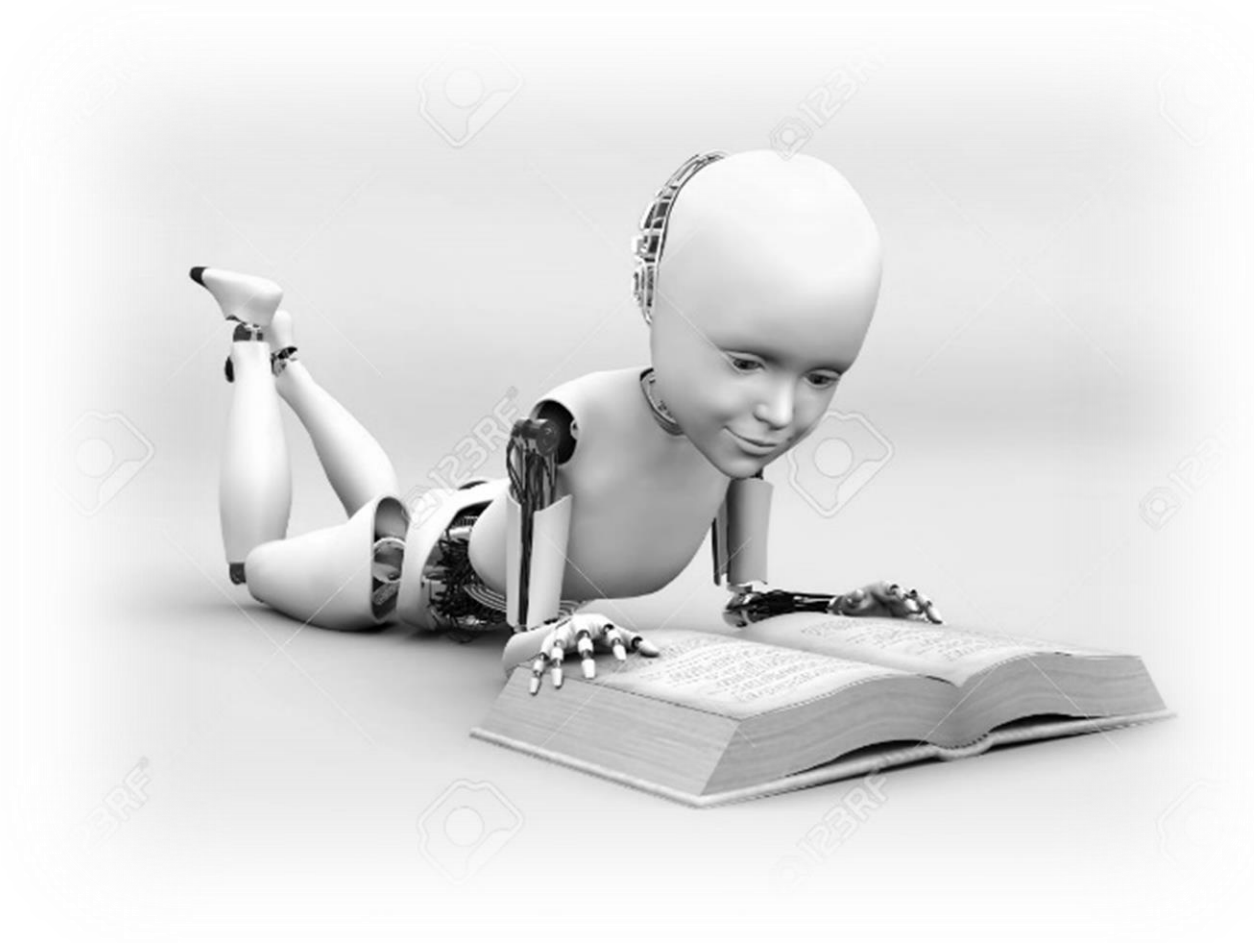
- L'utilizzo della logica ha alcune LIMITAZIONI:
  - Utilizzo di stati discreti (simbolici)
  - Costo dello sviluppo di basi di conoscenza (teorie) di grandi dimensioni
  - Complessità nel ragionamento automatico
    - Algoritmi di model checking efficienti solo per frammenti della Logica del Primo Ordine
    - Scarsa scalabilità
  - Complessità nella ingegnerizzazione e manutenzione delle basi di conoscenza

# IL RUOLO DEL MACHINE LEARNING

INDURRE CONOSCENZA DAI DATI, FUNZIONI DI DECISIONE *MODEL-FREE*

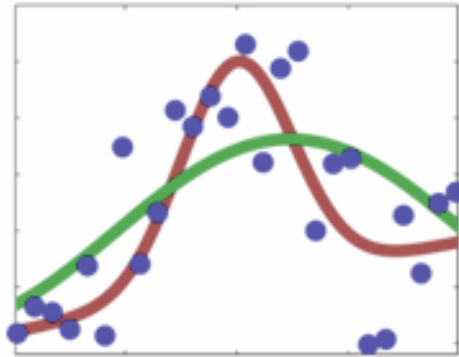


# COSA SIGNIFICA APPRENDERE DAI DATI?

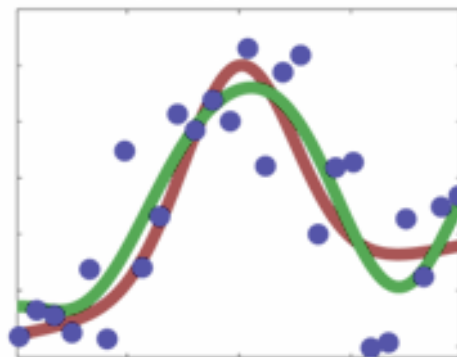


# LEARNING MACHINES

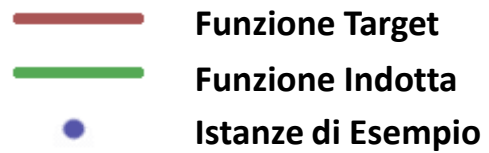
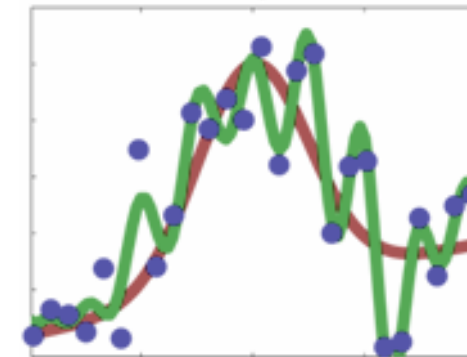
Funzione indotta  
& modello troppo semplice



Funzione indotta  
& modello adeguato

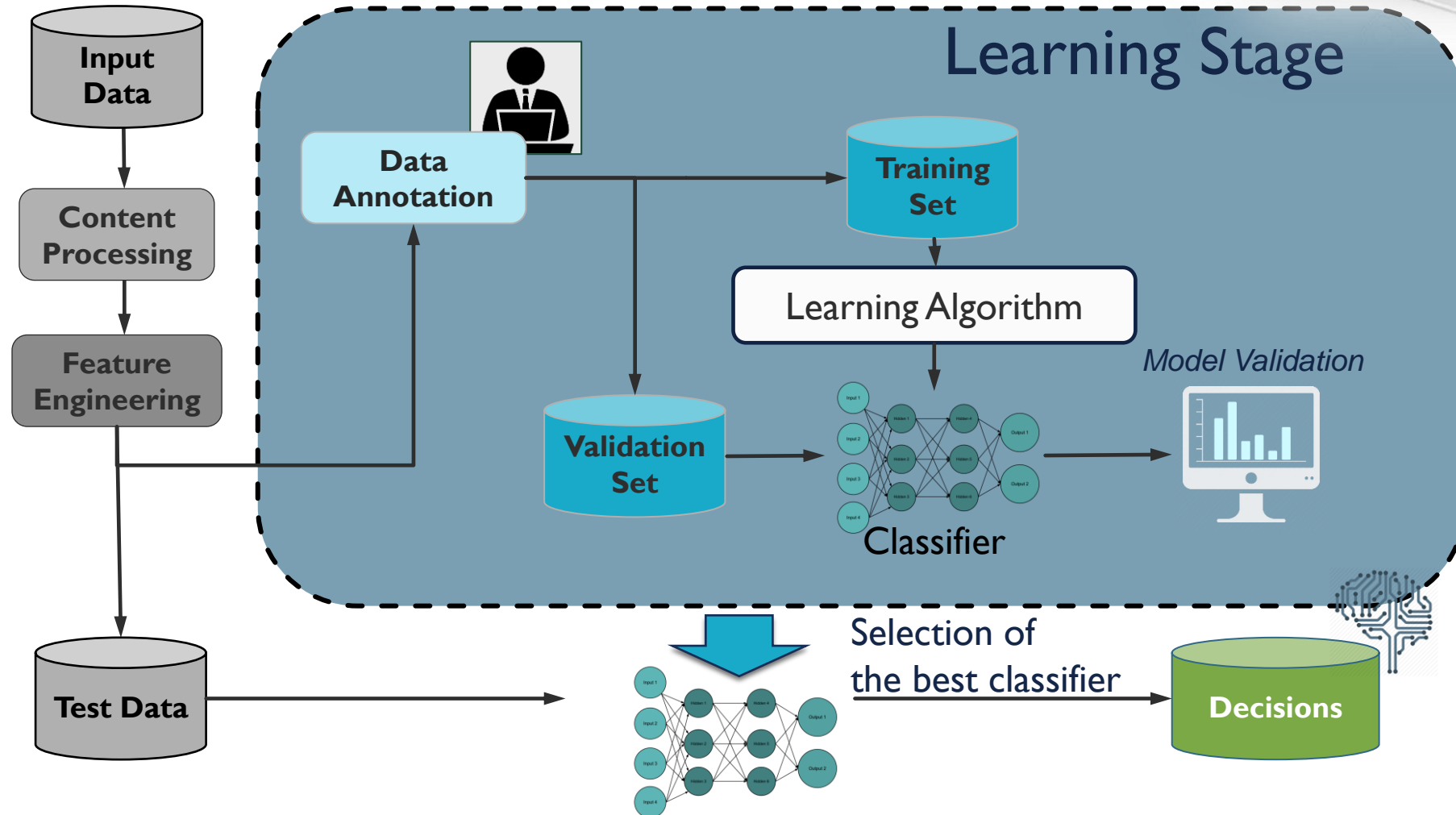


Funzione indotta  
& modello troppo complesso

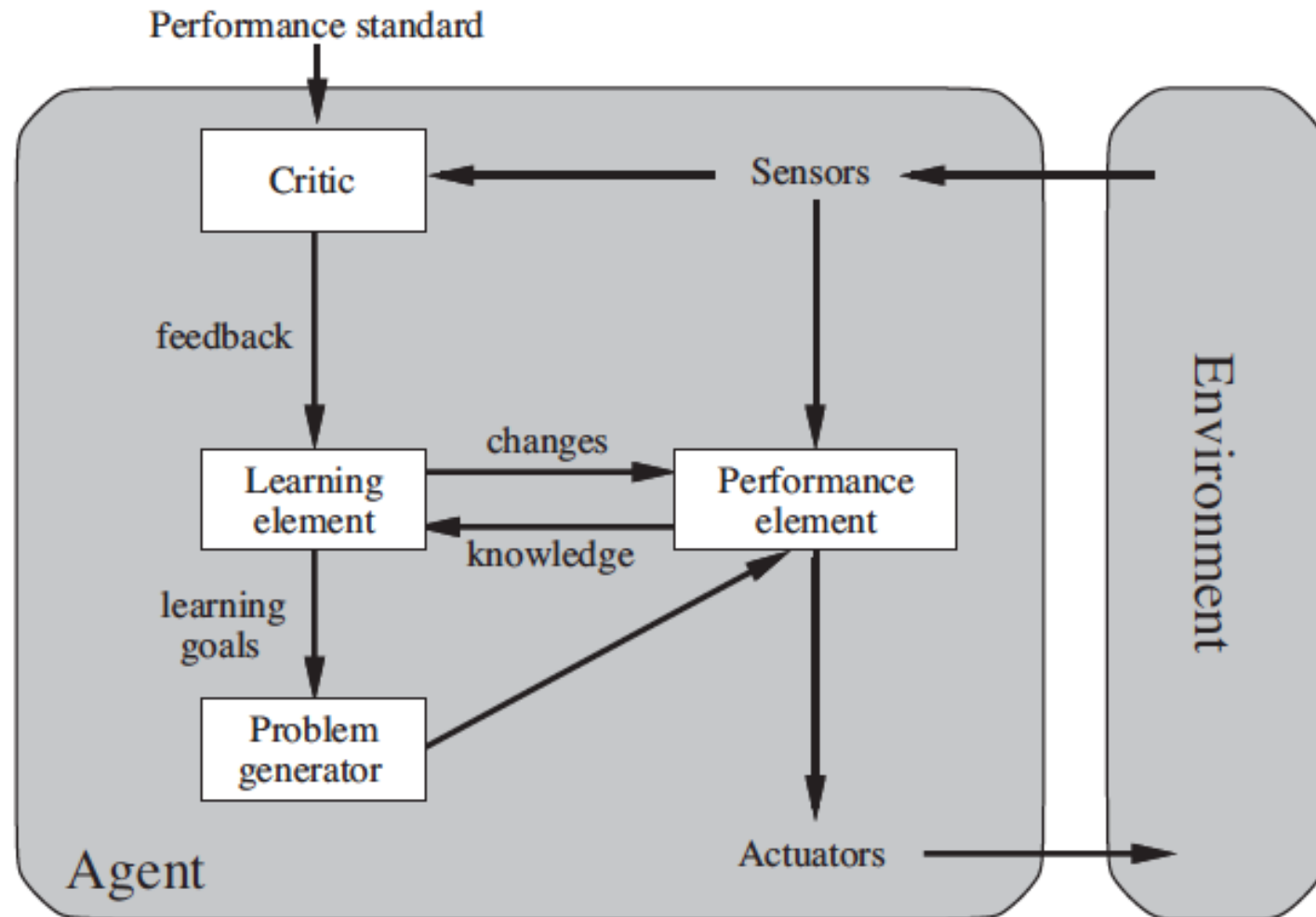




# MACHINE LEARNING WORKFLOW



# AIMA LEARNING ARCHITECTURE



# MACHINE LEARNING: DEFINITION

- *A computer program is said to **learn** from experience  $E$  with respect to some class of tasks  $T$  and performance measure  $P$ , if its performance at tasks in  $T$ , as measured by  $P$ , improves with experience  $E$  [Mitchell]*
- Problem definition for a learning agent
  - Task  $T$
  - Performance measure  $P$
  - Experience  $E$

# DESIGNING A LEARNING SYSTEM

1. Choosing the training experience
  - Examples of best moves, games outcome ...
2. Choosing the target function
  - board-move, board-value, ...
3. Choosing a representation for the target function
  - linear function with weights (hypothesis space)
4. Choosing a learning algorithm for approximating the target function
  - A method for parameter estimation

# INDUCTIVE LEARNING: LEARN A FUNCTION FROM EXAMPLES

- Simplest form: learn a function from examples

$f$  is the **target function**

An **example** is a pair  $(x, f(x))$

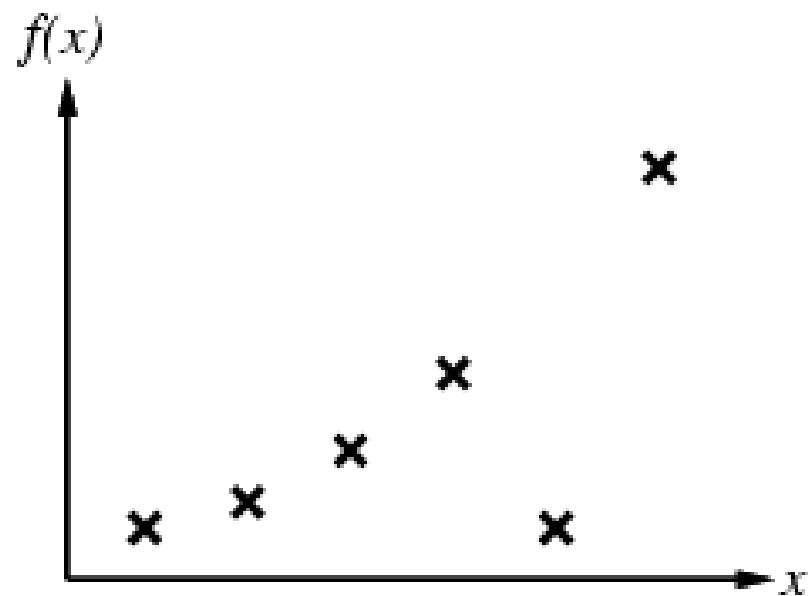
Problem: find a **hypothesis**  $h$

such that  $h \approx f$

given a **training set** of examples

(This is a highly simplified model of real learning:

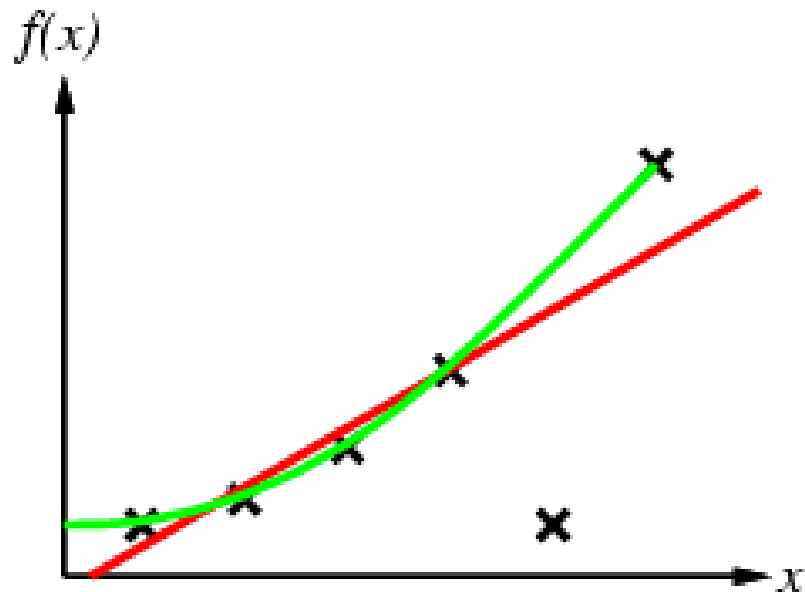
- Ignores prior knowledge
- Assumes examples are given)



# INDUCTIVE LEARNING METHOD

- Construct/adjust  $h$  to agree with  $f$  on training set  
( $h$  is **consistent** if it agrees with  $f$  on all examples)

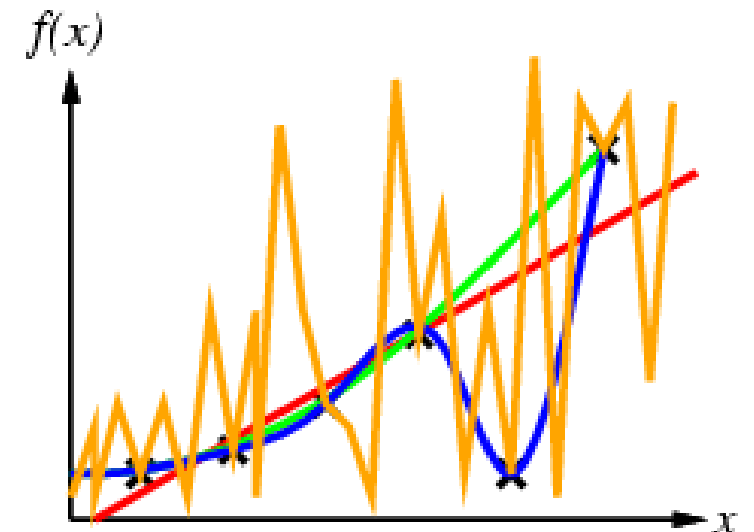
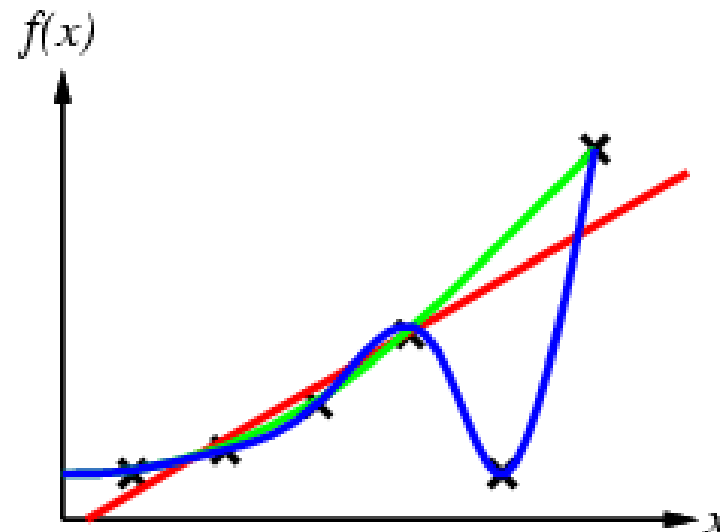
e.g., curve fitting:



# INDUCTIVE LEARNING METHOD

- Construct/adjust  $h$  to agree with  $f$  on training set  
( $h$  is **consistent** if it agrees with  $f$  on all examples)

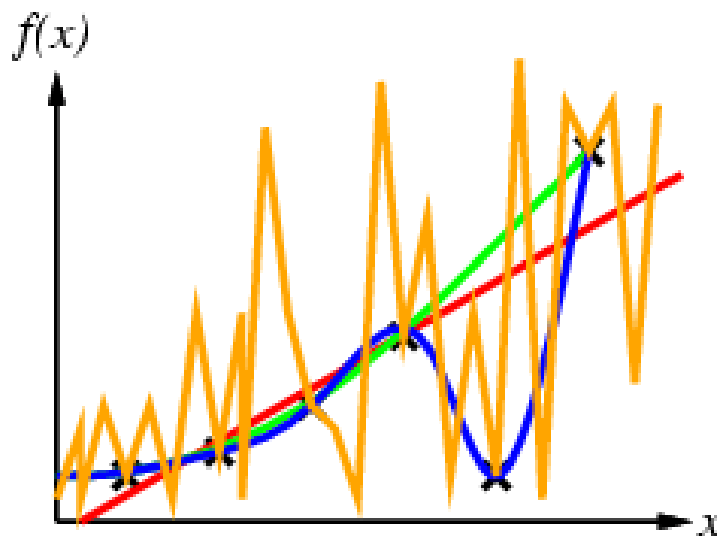
e.g., curve fitting:



# INDUCTIVE LEARNING METHOD

- Construct/adjust  $h$  to agree with  $f$  on training set ( $h$  is **consistent** if it agrees with  $f$  on all examples)

E.g., curve fitting:



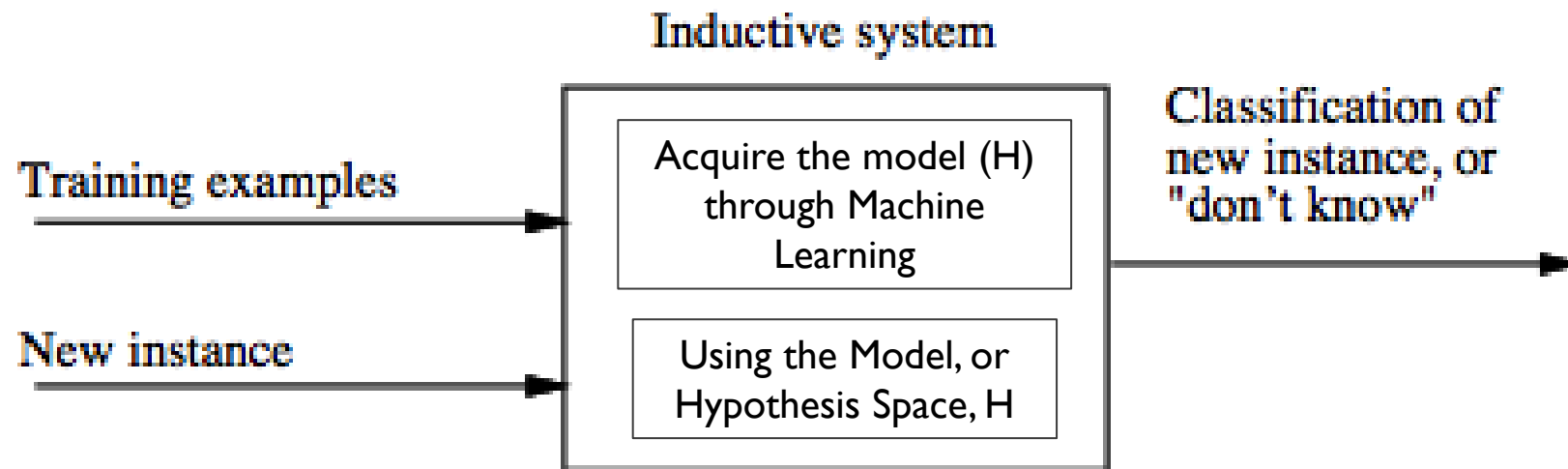
*novacula Occami*

**D**ico ergo ad q̄onem q̄  
qz pluralitas  
non est ponenda sine necessitate ⁊ non  
ē necessitas quare debeat poni t̄pus di-  
scretum mensurās motum angeli. naz

Ockham's razor: *prefer the simplest hypothesis consistent with data*



# INDUCTIVE SYSTEM



# LEARNING DECISION TREES

Problem: decide whether to wait for a table at a restaurant, based on the following attributes:

1. Alternate: is there an alternative restaurant nearby?
2. Bar: is there a comfortable bar area to wait in?
3. Fri/Sat: is today Friday or Saturday?
4. Hungry: are we hungry?
5. Patrons: number of people in the restaurant (None, Some, Full)
6. Price: price range (\$, \$\$, \$\$\$)
7. Raining: is it raining outside?
8. Reservation: have we made a reservation?
9. Type: kind of restaurant (French, Italian, Thai, Burger)
10. WaitEstimate: estimated waiting time (0-10, 10-30, 30-60, >60)

# ATTRIBUTE-BASED REPRESENTATIONS

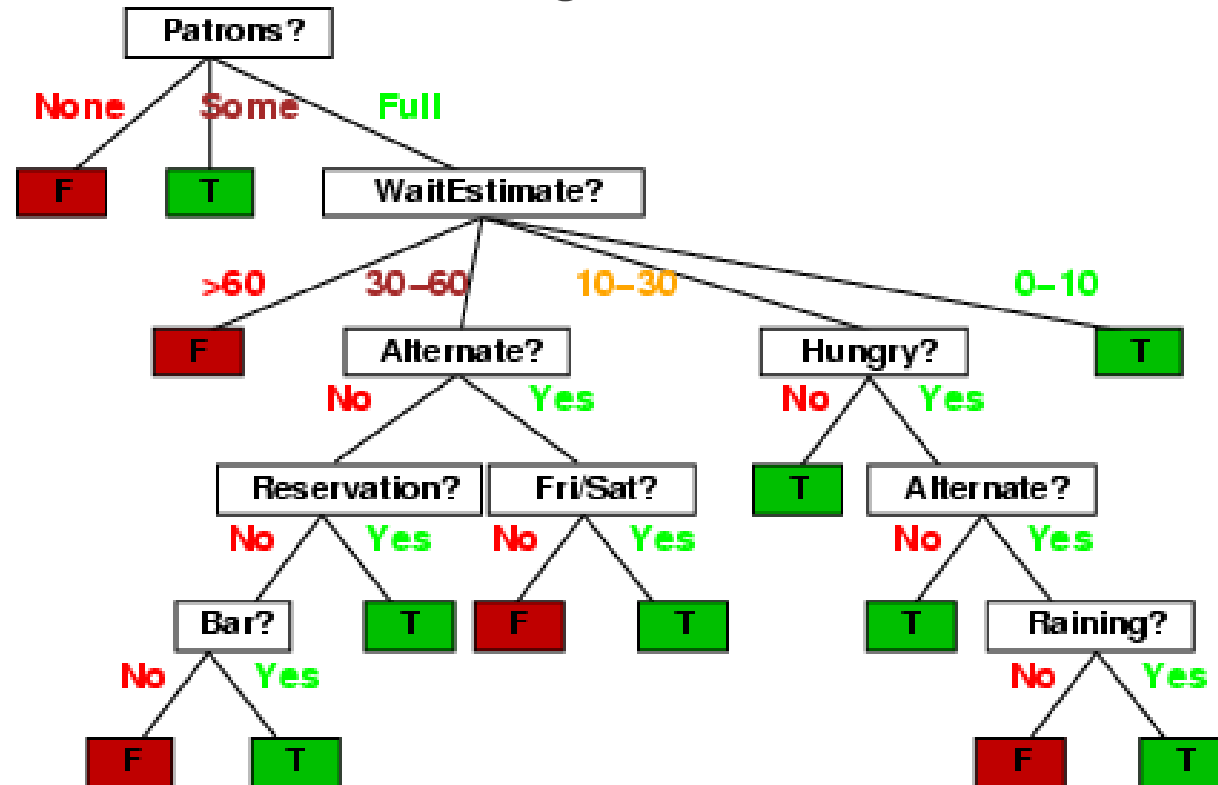
- Examples described by **attribute values** (Boolean, discrete, continuous)
- E.g., situations where I will/won't wait for a table:

Example	Attributes										Target <i>Wait</i>
	<i>Alt</i>	<i>Bar</i>	<i>Fri</i>	<i>Hun</i>	<i>Pat</i>	<i>Price</i>	<i>Rain</i>	<i>Res</i>	<i>Type</i>	<i>Est</i>	
$X_1$	T	F	F	T	Some	\$\$\$	F	T	French	0-10	T
$X_2$	T	F	F	T	Full	\$	F	F	Thai	30-60	F
$X_3$	F	T	F	F	Some	\$	F	F	Burger	0-10	T
$X_4$	T	F	T	T	Full	\$	F	F	Thai	10-30	T
$X_5$	T	F	T	F	Full	\$\$\$	F	T	French	>60	F
$X_6$	F	T	F	T	Some	\$\$	T	T	Italian	0-10	T
$X_7$	F	T	F	F	None	\$	T	F	Burger	0-10	F
$X_8$	F	F	F	T	Some	\$\$	T	T	Thai	0-10	T
$X_9$	F	T	T	F	Full	\$	T	F	Burger	>60	F
$X_{10}$	T	T	T	T	Full	\$\$\$	F	T	Italian	10-30	F
$X_{11}$	F	F	F	F	None	\$	F	F	Thai	0-10	F
$X_{12}$	T	T	T	T	Full	\$	F	F	Burger	30-60	T

- Classification of examples is **positive** (T) or **negative** (F)

# DECISION TREES

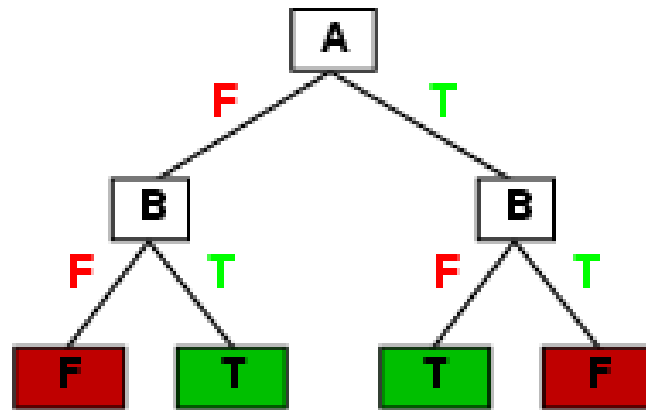
- One possible representation for hypotheses
- E.g., here is the “true” tree for deciding whether to wait:



# EXPRESSIVENESS

- Decision trees can express any function of the input attributes.
- E.g., for Boolean functions, truth table row  $\rightarrow$  path to leaf:

A	B	A xor B
F	F	F
F	T	T
T	F	T
T	T	F



- Trivially, there is a consistent decision tree for any training set with one path to leaf for each example (unless  $f$  nondeterministic in  $x$ ) but it probably won't generalize to new examples
- Prefer to find more **compact** decision trees

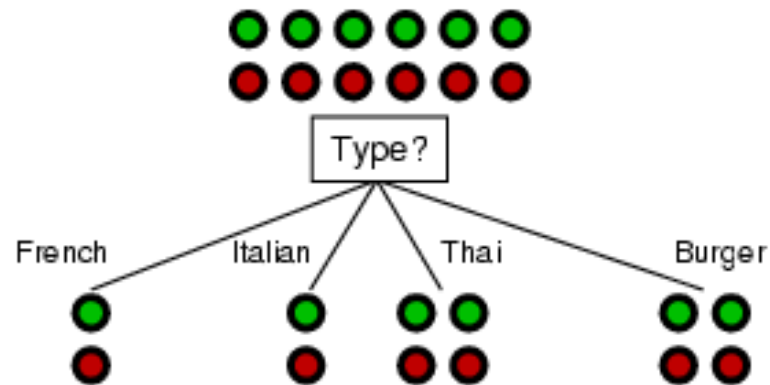
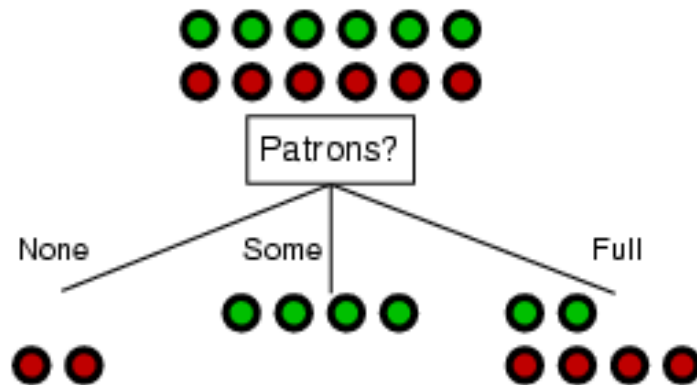
# DECISION TREE LEARNING

- Aim: find a small tree consistent with the training examples
- Idea: (recursively) choose "most significant" attribute as root of (sub)tree

```
function DTL(examples, attributes, default) returns a decision tree
  if examples is empty then return default
  else if all examples have the same classification then return the classification
  else if attributes is empty then return MODE(examples)
  else
    best ← CHOOSE-ATTRIBUTE(attributes, examples)
    tree ← a new decision tree with root test best
    for each value  $v_i$  of best do
      examplesi ← {elements of examples with  $best = v_i$ }
      subtree ← DTL(examplesi, attributes - best, MODE(examples))
      add a branch to tree with label  $v_i$  and subtree subtree
  return tree
```

# CHOOSING AN ATTRIBUTE

- Idea: a good attribute splits the examples into subsets that are (ideally) "all positive" or "all negative"

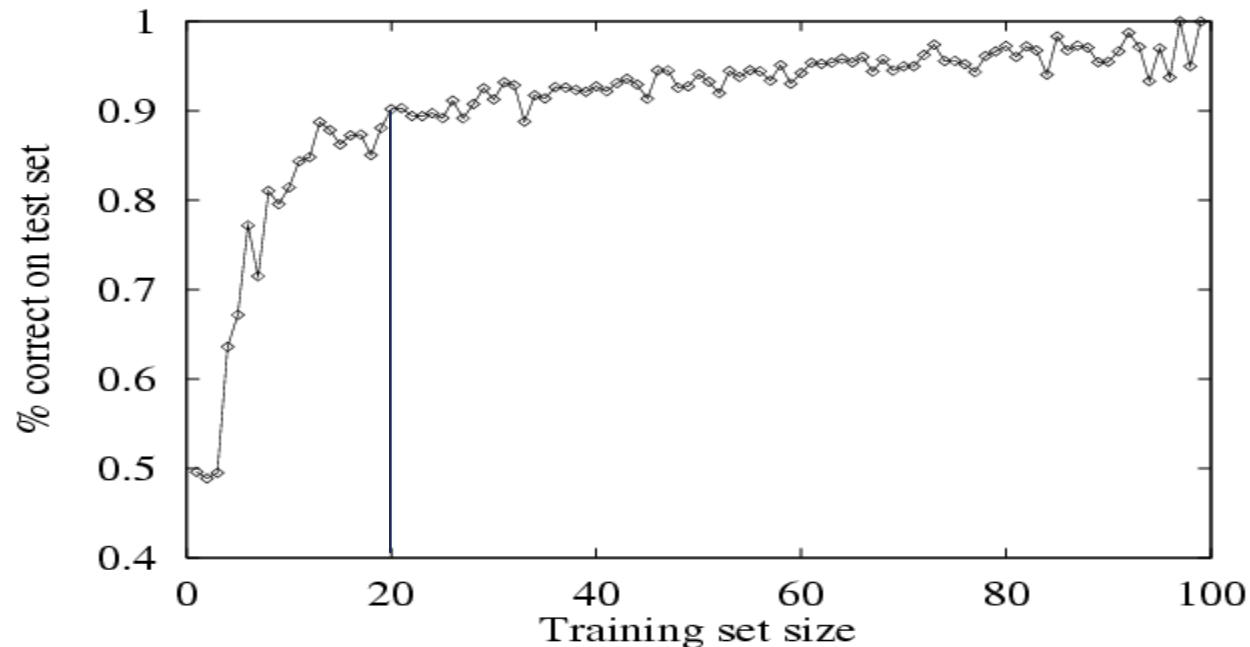


- *Patrons?* is a better choice

# PERFORMANCE MEASUREMENT

- How do we know that  $h \approx f$ ?
  1. Use theorems of computational/statistical learning theory
  2. Try  $h$  on a new **test set** of examples  
(use **same** distribution over example space as training set)

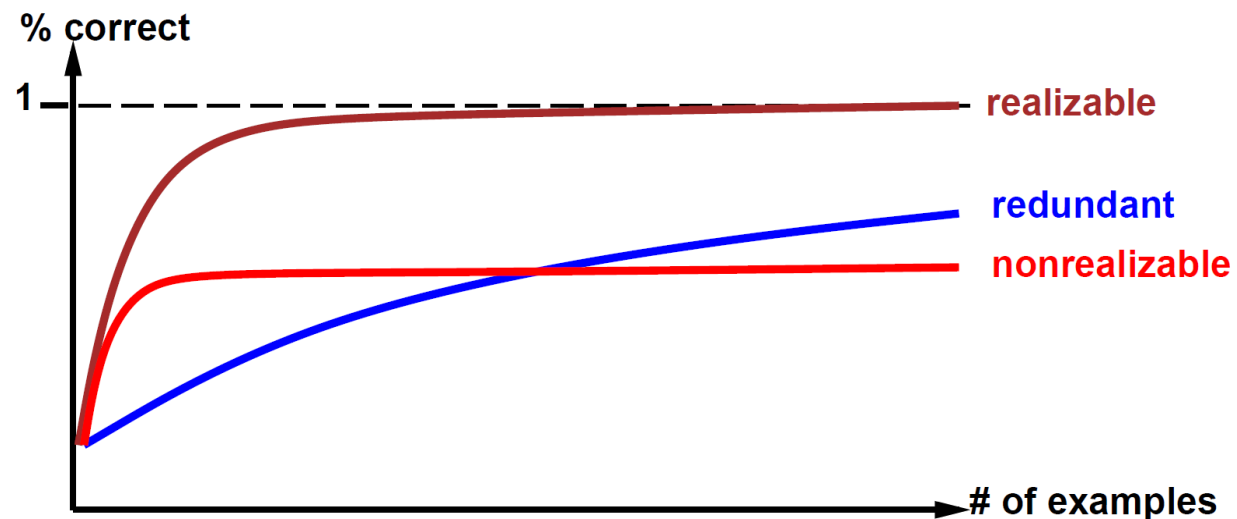
**Learning curve** = % correct on test set as a function of training set size



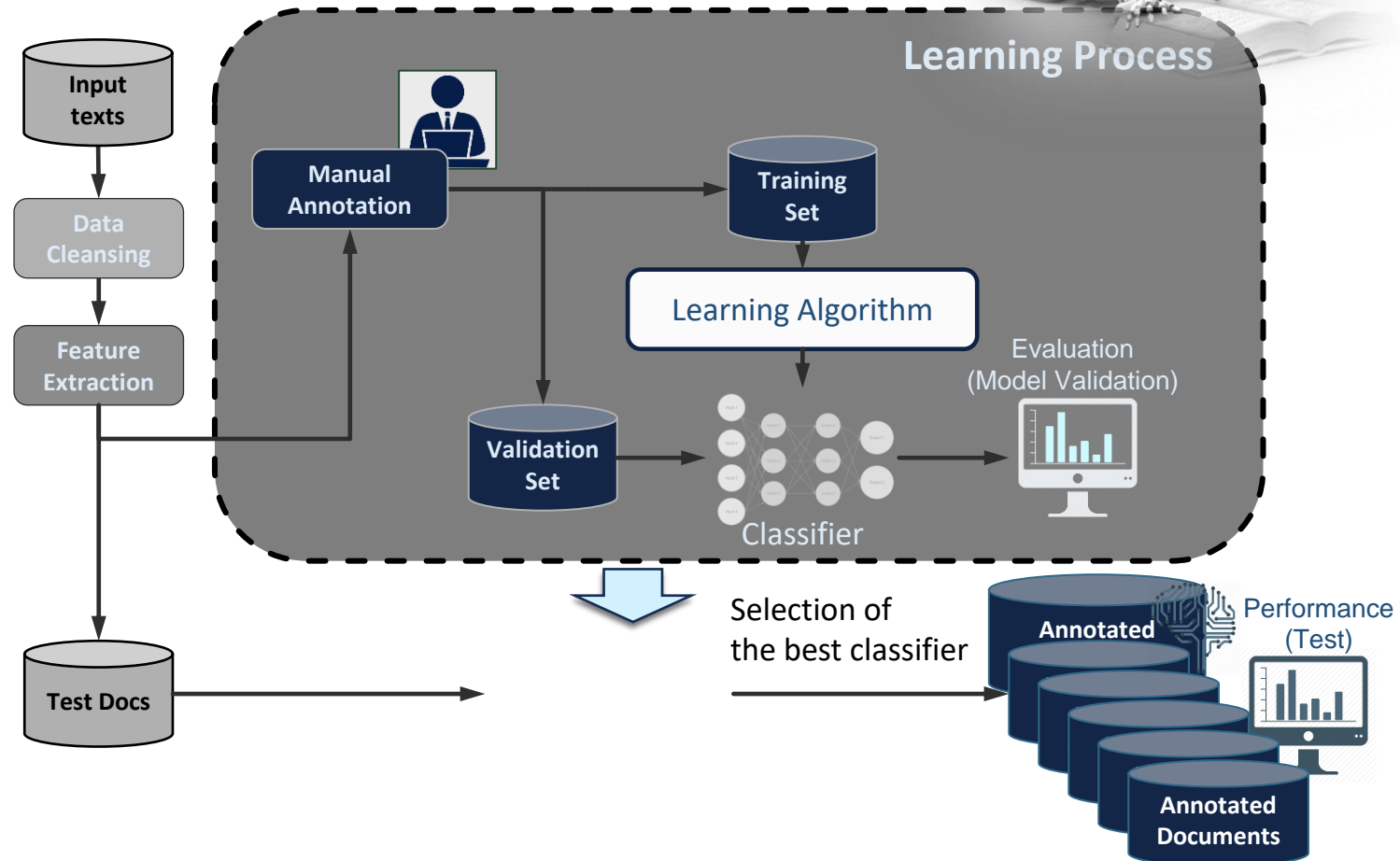
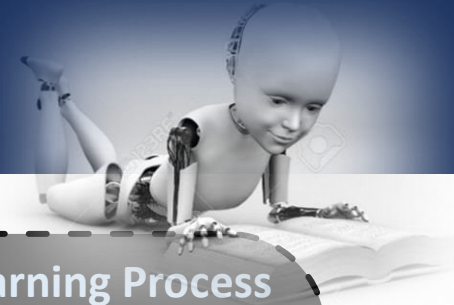


# PERFORMANCE MEASUREMENTS (2)

- Learnability depends on
  - **realizable** kind of performances vs.
  - ... **non-realizable** ones
  - **Non-realizability** depends on
    - Missing attributes
    - Limitation on the hypothesis space (e.g. non expressive functions)
  - **Redundant expressiveness** is related to cases where a large number of irrelevant attributes are used

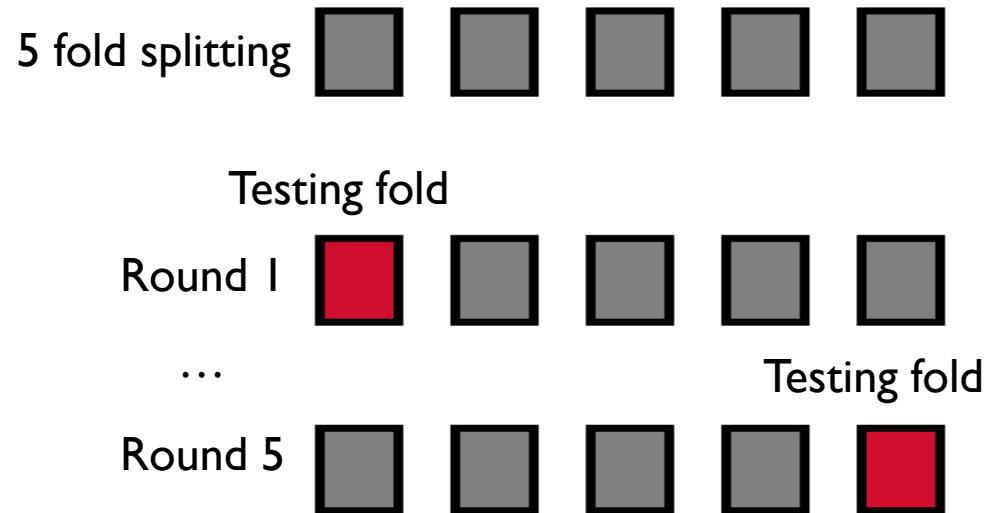


# MACHINE LEARNING WORKFLOW



# N-FOLD CROSS VALIDATION

- Data is split into  $n$  subsets of equal size
- Each subset in turn is used for testing and the remainders  $n-1$  for training
- The metrics estimated in each round are averaged



# MACHINE LEARNING TASKS

## ■ SUPERVISED LEARNING DA ESEMPI

### ■ CLASSIFICATION

- Approcci discriminativi
- Approcci generative
- Outlier and deviation detection

### ■ REGRESSION

### ■ Dependency modeling

- Discovery di Associazioni/Relazioni, Sommari, Inferenza/Causalità

### ■ SEQUENCE CLASSIFICATION

- Temporal learning
- Trend analysis and change/anomaly detection

## ■ UNSUPERVISED LEARNING

### ■ Clustering

### ■ Embedding ottimo: Encoding/Decoding

- Representation Learning for Images
- PreTraining as optimal encoding

## ■ REINFORCEMENT LEARNING

### ■ Penalty/Reward function from the Environment

### ■ Autonomous Systems

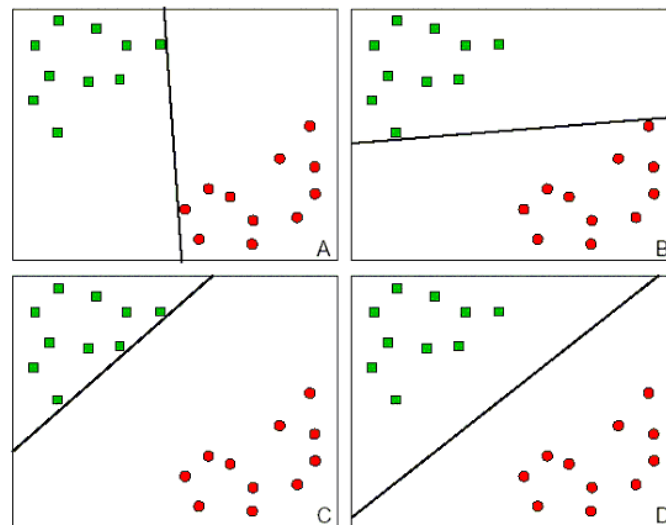
### ■ Hard for complex problems

# METODI DI ML: SELEZIONE DEI MODELLI

- Approcci discriminativi

- Lineari

$$h(\mathbf{x}) = \text{sign}(\mathbf{W} \cdot \mathbf{x} + b)$$



- Approcci probabilistici

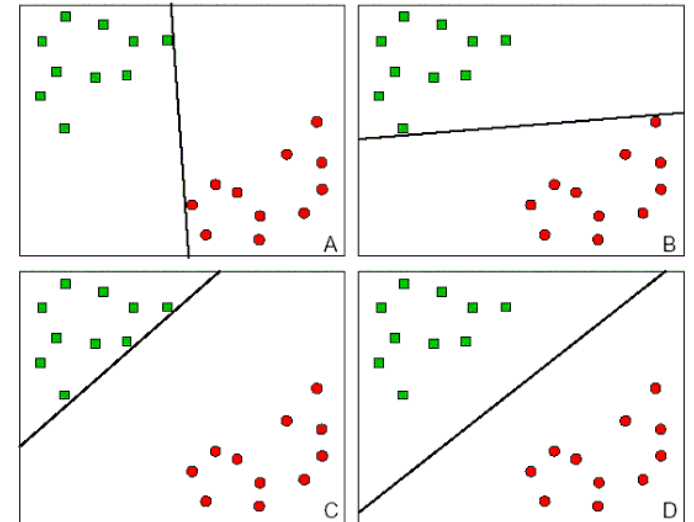
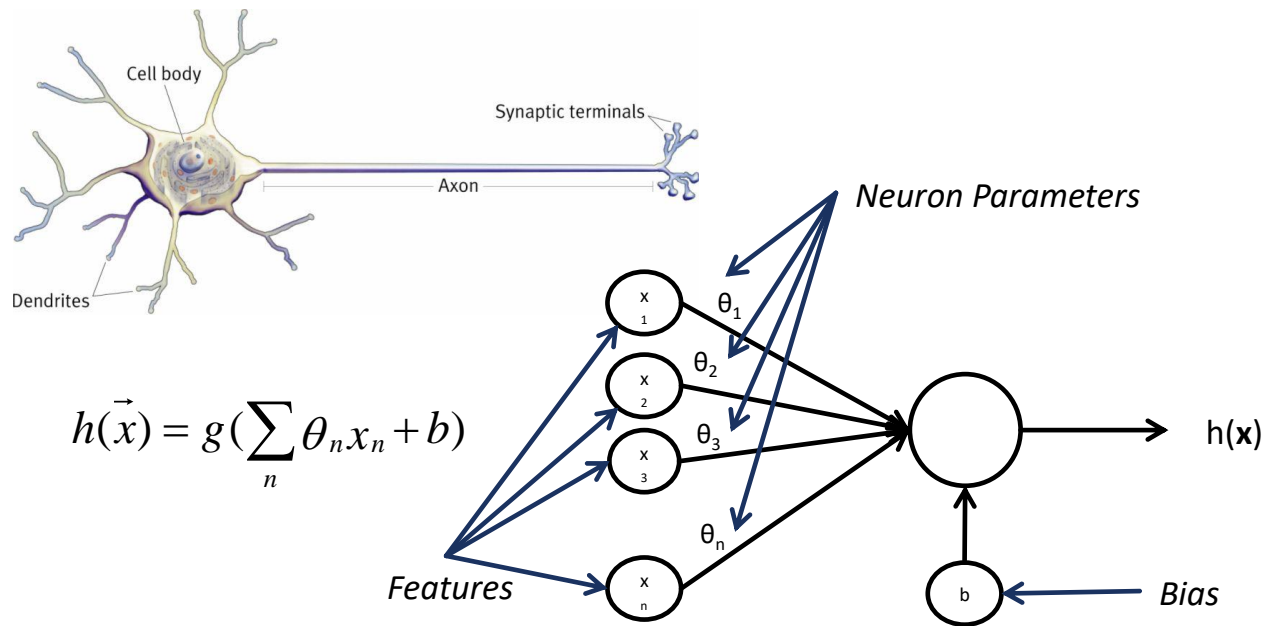
- Stima delle probabilità  $p(\mathcal{C}_k|\mathbf{x})$  attraverso un training set

- Modello generativo ed uso della inversione Bayesiana

$$p(\mathcal{C}_k|\mathbf{x}) = \frac{p(\mathbf{x}|\mathcal{C}_k)p(\mathcal{C}_k)}{p(\mathbf{x})}.$$

# PERCEPTRON (ROSENBLATT, 1958)

- Linear Classifier mimicking a neuron



# ADDING LAYERS ...

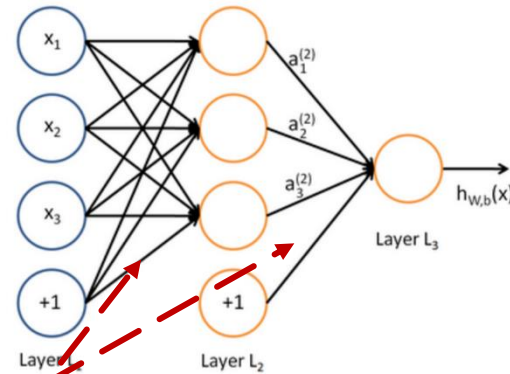
- From simple linear laws ...

$$h(\vec{x}) = g(\vec{x}; \vec{\theta}, b) = g\left(\sum_n \theta_n x_n + b\right)$$

- to feedforward structures. It can be made dependent on a sequence of functions  $g^{(1)}$  and  $g^{(2)}, \dots, g^{(k)}$  that give rise to a structured hypothesis:

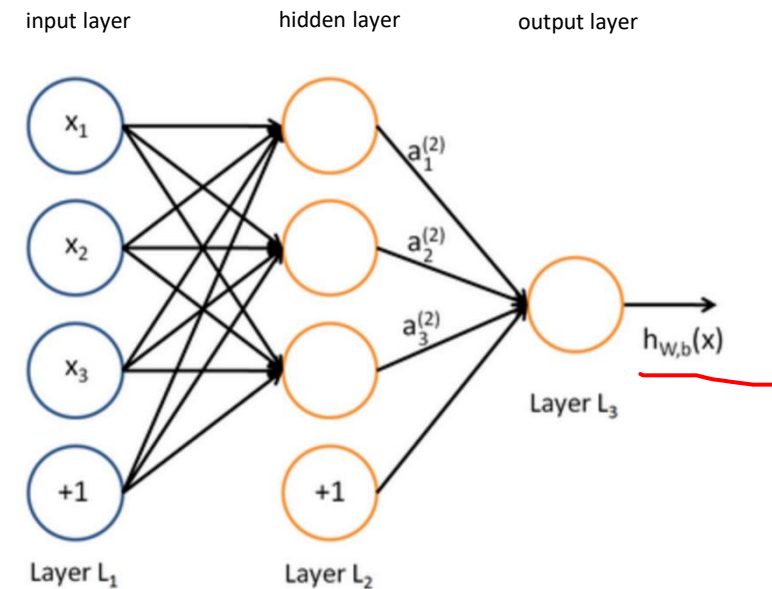
$$\begin{aligned} h(\vec{x}) &= g^{(2)}\left(g^{(1)}(\vec{x}; \vec{\theta}^{(1)}, b^{(1)}); \vec{\theta}^{(2)}, b^{(2)}\right) = \\ &= W^{(2)} g^{(2)}\left(g^{(1)}(W^{(1)} \cdot \vec{x} + b^{(1)}) + b^{(2)}\right) \end{aligned}$$

- Hidden layers  $h^{(1)}(\vec{x}) = g^{(1)}(W^{(1)} \vec{x} + b^{(1)})$



# NEURAL NETWORKS

- Each circle represent a **neuron** (or unit)
  - 3 **input**, 3 **hidden** and 1 **output**
- $n_f=3$  is the **number of layers**
- $S_l$  denotes the **number of units in layer l**
- Layers:
  - Layer l is denoted as  $L_l$
  - Layer l and l+1 are connected by a matrix  $W^{(l)}$  of parameters
    - $W^{(l)}_{i,j}$  connects neuron j in layer l with neuron i in layer l+1
- $b^{(l)}_i$  is the **bias** associated to neuron i in layer l+1

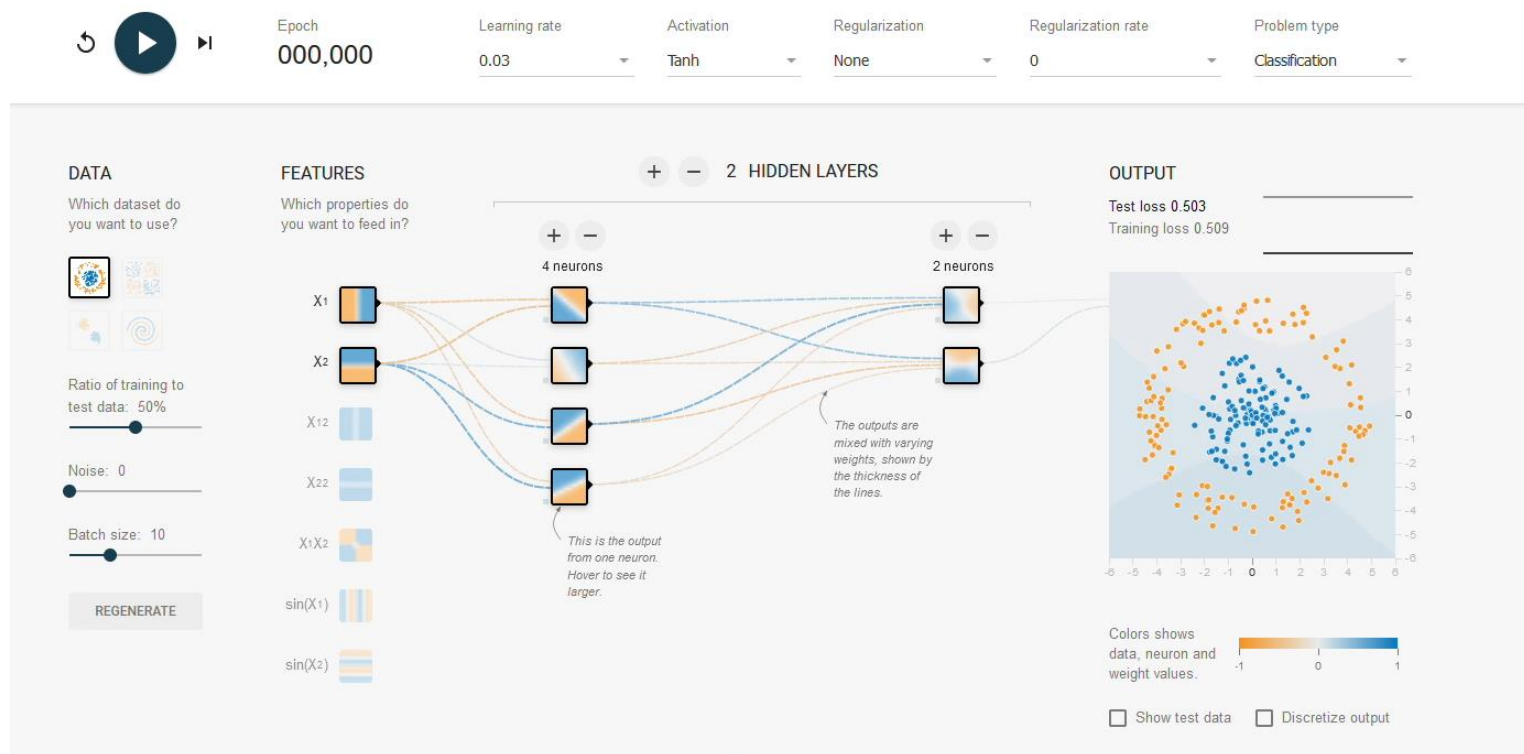




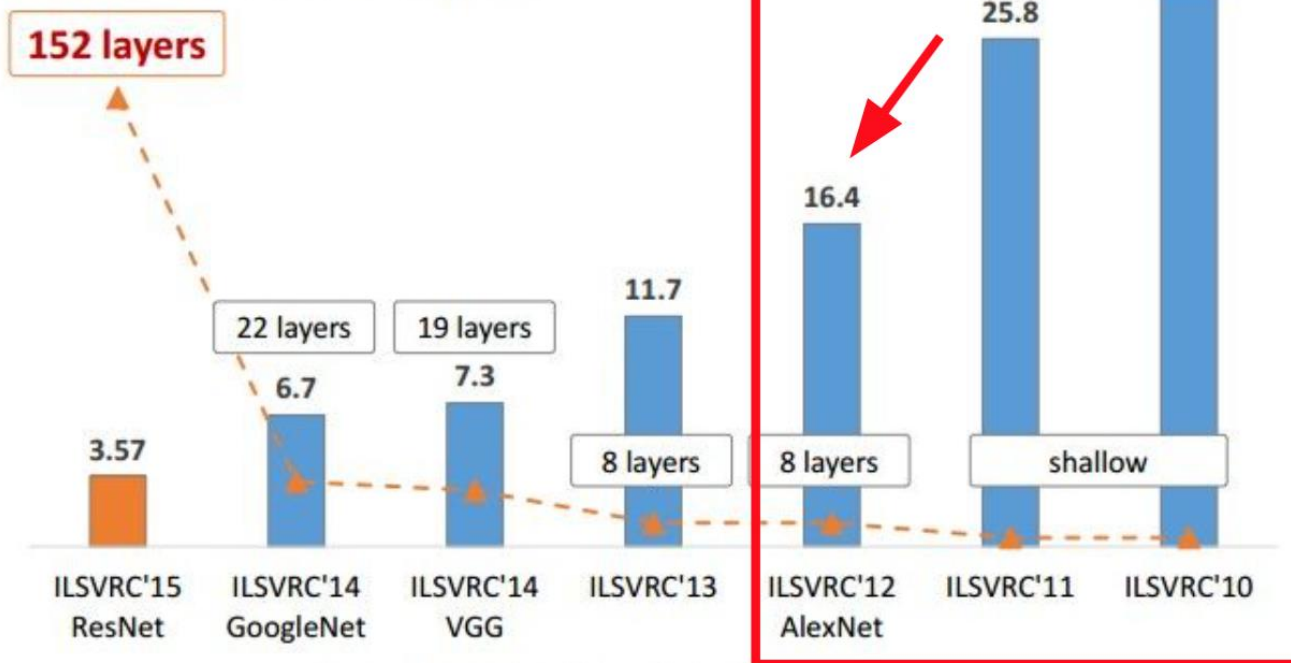
# A SIMPLE DEMO ON TENSORFLOW

- Look at: <https://playground.tensorflow.org/>

Tinker With a **Neural Network** Right Here in Your Browser.  
Don't Worry, You Can't Break It. We Promise.

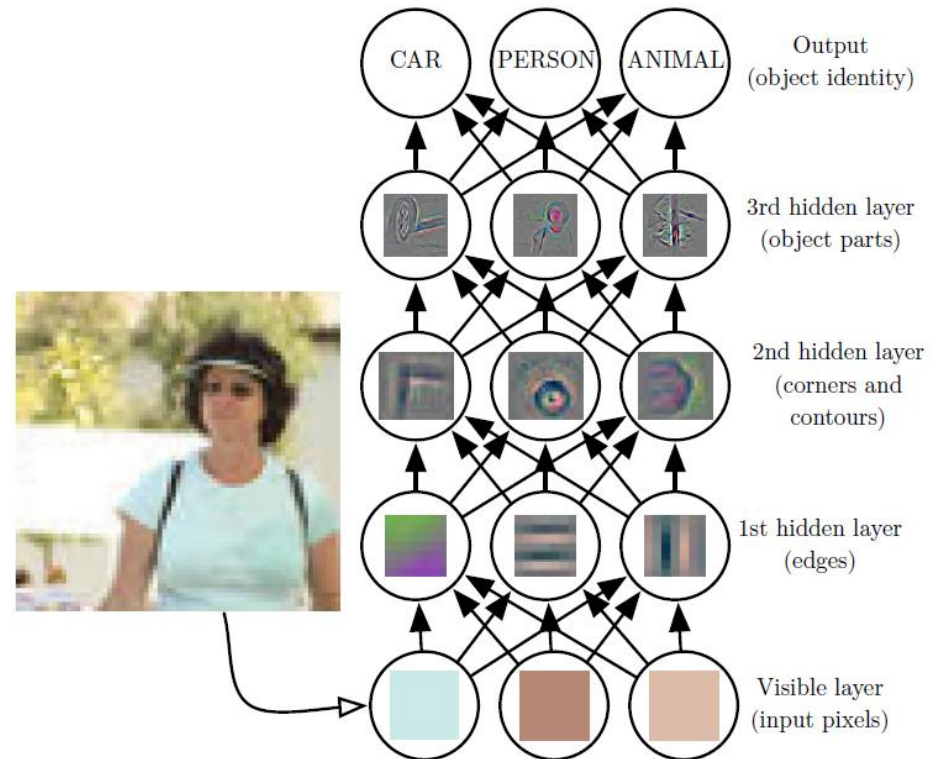


# Revolution of Depth



ImageNet Classification top-5 error (%)

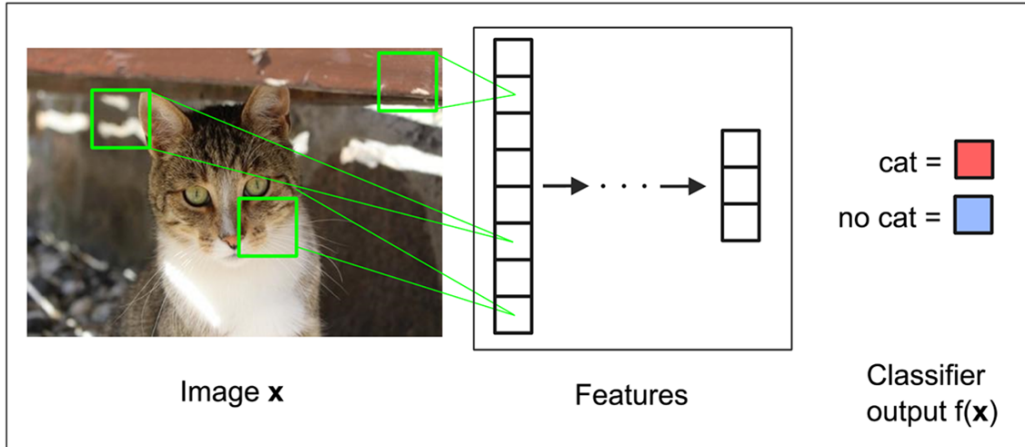
# REPRESENTATION AND LEARNING: THE ROLE OF DEPTH



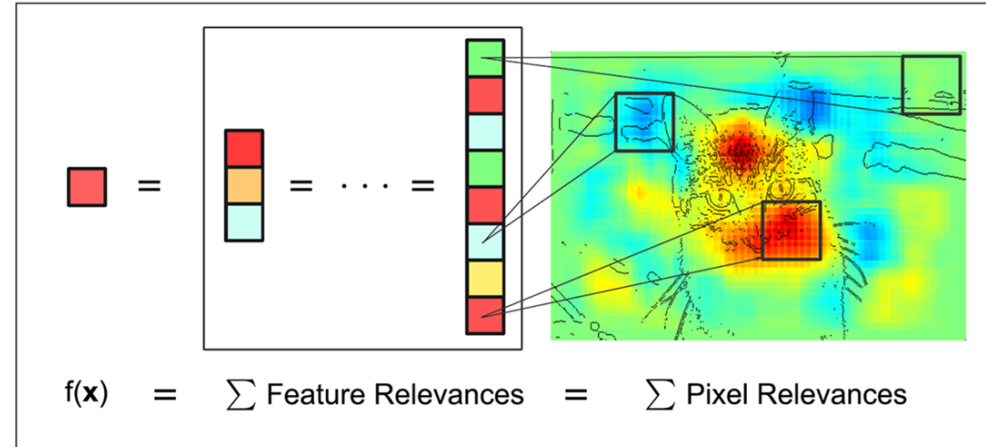
Zeiler and Fergus (2014)

# EXPLAINABILITY: DECISIONI E STATI DI ATTIVAZIONE

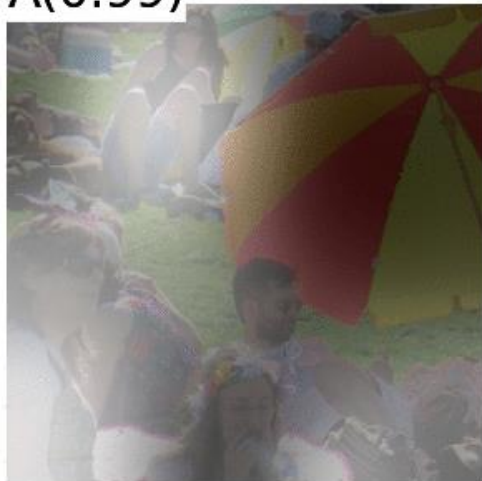
Classification



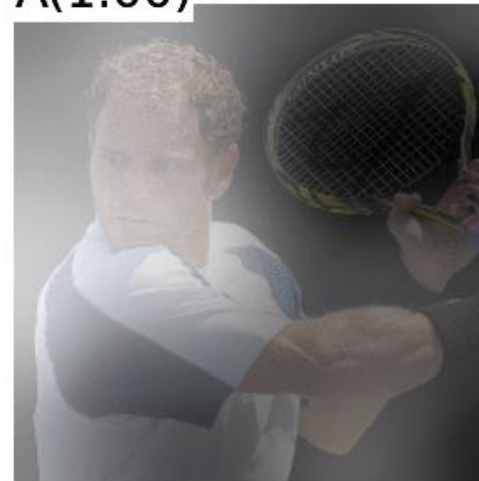
Pixel-wise Explanation



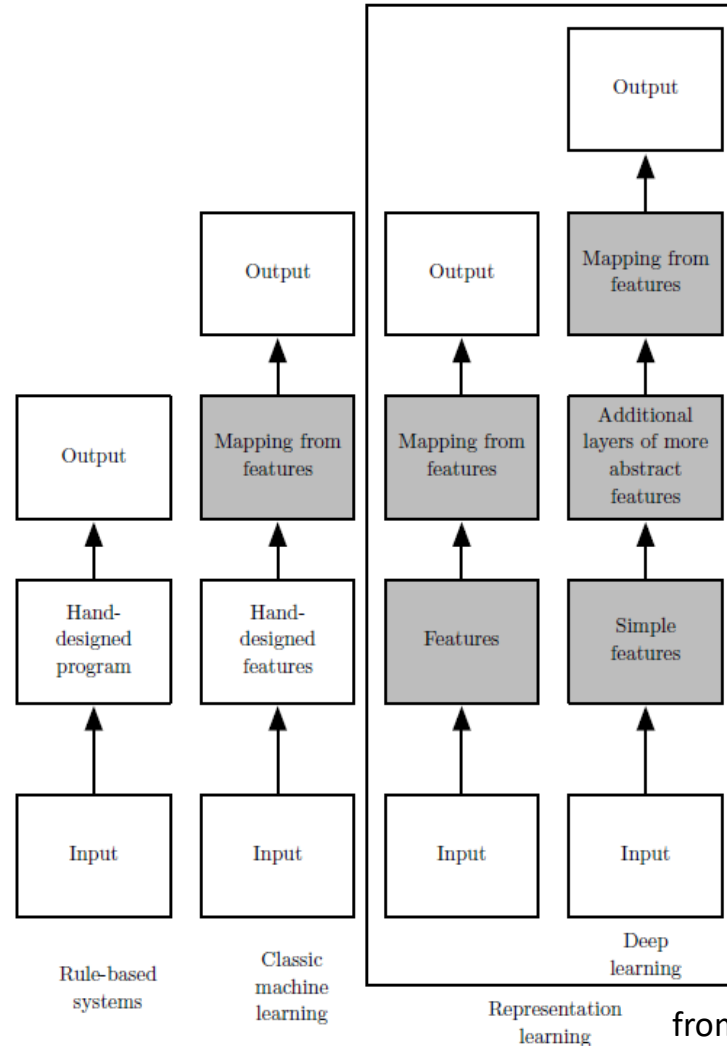
A(0.99)



A(1.00)

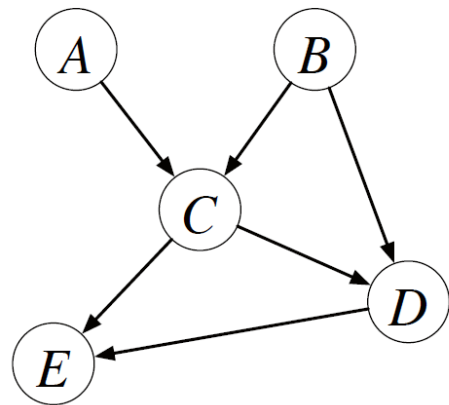


# LEARNING MULTIPLE COMPONENTS



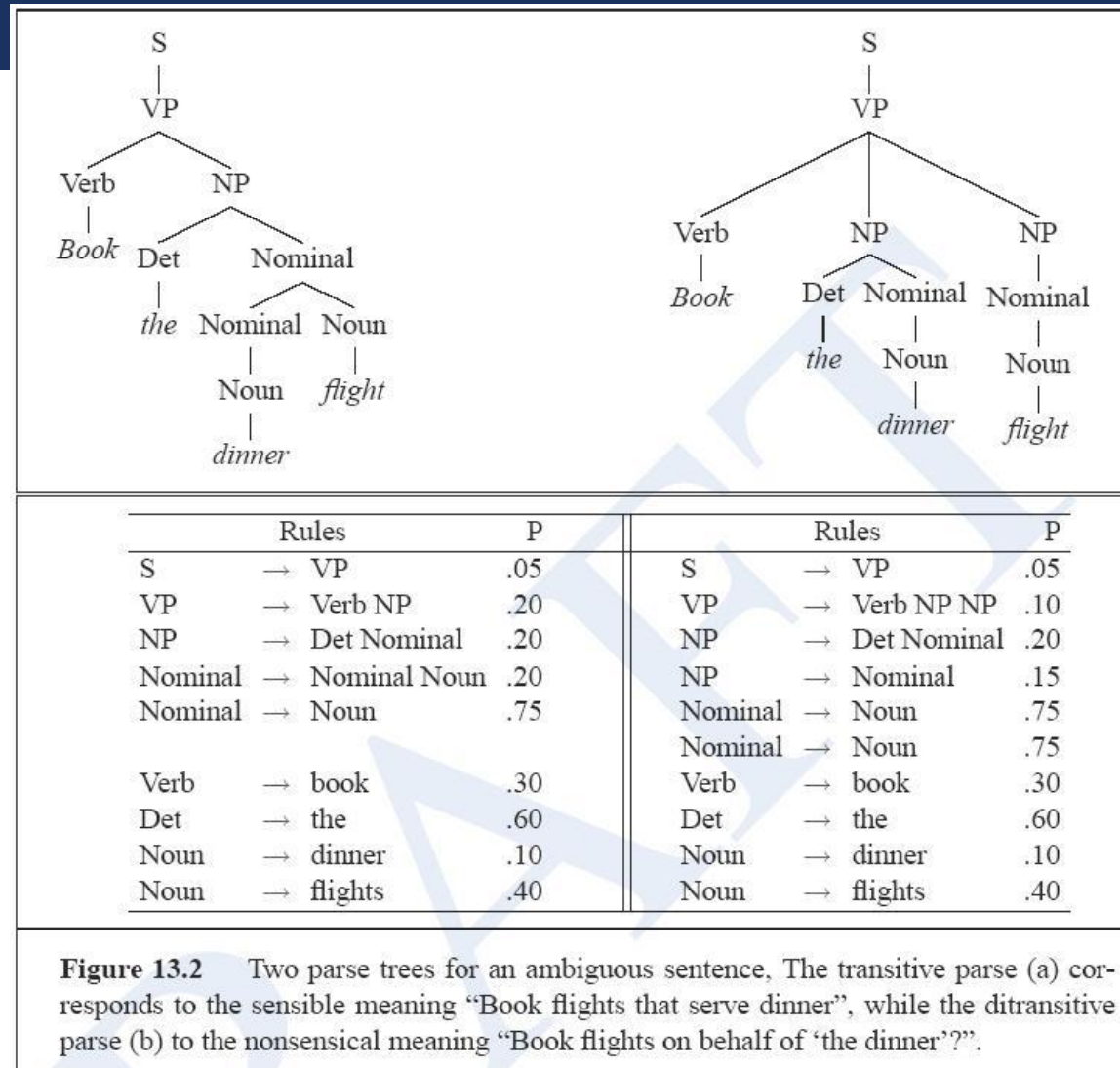
from Goodfellow et al., DL MIT book

# MODELLI GRAFICI

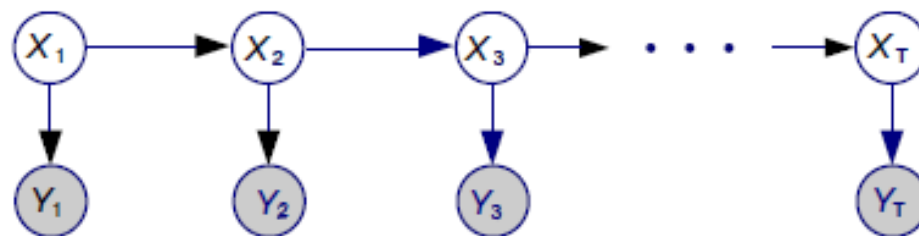


$$p(A, B, C, D, E) = p(A)p(B)p(C|A, B)p(D|B, C)p(E|C, D)$$

# GRAMMATICHE PROBABILISTICHE: TRA SINTASSI & STATISTICA



# SEQUENZE: HIDDEN MARKOV MODELS



$$p(X_1, \dots, X_T, Y_1, \dots, Y_T) = p(X_1) p(Y_1 | X_1) \prod_{t=2}^T [p(X_t | X_{t-1}) p(Y_t | X_t)]$$

- Stati = Categorie/Concetti/Proprietà
- Osservazioni: simboli di un certo linguaggio
- Emissioni vs. Transizioni
- Applicazioni:
  - Speech Recognition (simboli: fonemi, stati: segmentazione)
  - POS tagging (simboli: parole, stati: categorie grammaticali)





# APPLICAZIONI DELL'INTELLIGENZA ARTIFICIALE

ESEMPI ILLUSTRI E USE CASE INDUSTRIALI



# DATA-DRIVEN APPLICATIONS: ML, SEARCH & CONTENT ANALYTICS

## Enterprise Content Analytics

**Knowledge**



- Intelligent Business Analysis
- CyberSecurity
- Mental Health Monitoring
- Enterprise Ontology Management
- Design and Maintenance of conceptual catalogs
- Process Mapping
- ...

**Organizzazione Concettuale  
& Aggregazione**

**Information Extraction**



- Intelligent Query Processing
- Entity Recognition and Linking
- Event/Activity Recognition
- Document Classification
- Language Processing

**IE/ Analysis / Distillation**

**Data**

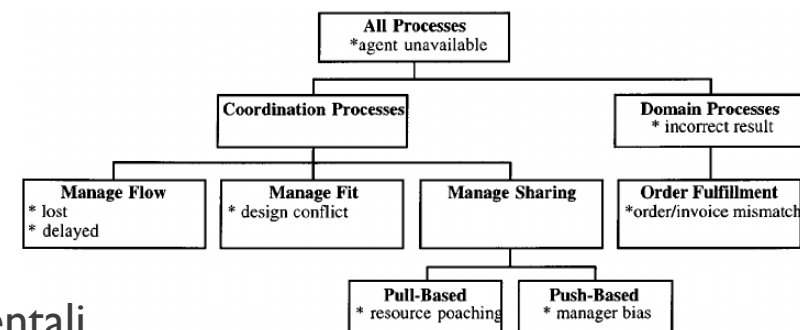


- Narrative and Legal Texts
- Legacy Models
- Social Media
- Open Web sources

**Data Gathering**

# BANKING: ABILABERT IN DECODE

- 5 banche coordinate da ABILAB
- Una Process Taxonomy condivisa e differenti Basi di Dati Documentali
- Automatic Text-driven Process Mapping basato su reti neurali Trasformers



DECODE automatic metadata creation for financial documents ABI Lab TOR VERGATA

Home Taxonomy Info

› Insert Text/Paragraph

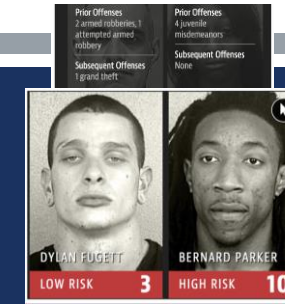
Text/Paragraph:

All'inizio di ogni anno il Responsabile dell'Area Finanza, Tesoreria e Capital Management, di concerto con il Responsabile della Funzione Governo Strategico del Rischio, definisce il budget da allocare all'attività di trading della Funzione Governo Strategico del Rischio ed i relativi limiti di VaR e stop-loss. Detto budget viene comunicato per iscritto ai livelli gerarchici inferiori contestualmente ai limiti di VaR e di stop-loss. Il budget ed i limiti delegati vengono comunicati alla Funzione Controlli dei Rischi.

SUBMIT CLEAR

› Upload File

# COMPAS: PROFILING



- COMPAS dataset (*Correctional Offender Management Profiling for Alternative Sanctions*)
  - raccoglie dati nell'ambito della giustizia penale utilizzati per prevedere il rischio di recidiva di un imputato.
  - pubblicato da ProPublica nel 2016 sulla base dei dati raccolti dalla contea di Broward.

Attributes	Type	Values	#Missing values	Description
sex	Binary	{Male, Female}	0	Sex
age	Numerical	[18 - 96]	0	Age in years
age_cat	Categorical	3	0	Age category
race	Categorical	6	0	Race
juv_fel_count	Numerical	[0 - 20]	0	The juvenile felony count
juv_misd_count	Numerical	[0 - 13]	0	The juvenile misdemeanor count
juv_other_count	Numerical	[0 - 17]	0	The juvenile other offenses count
priors_count	Numerical	[0 - 38]	0	The prior offenses count
c_charge_degree	Binary	{F, M}	0	Charge degree of original crime
score_text	Categorical	3	0	ProPublica-defined category of decile score
v_score_text	Categorical	3	0	ProPublica-defined category of v_decile_score
two_year_recid	Binary	{0, 1}	0	Whether the defendant is rearrested within two years

**Caratteristiche** Contiene 7.214 istanze. Ogni imputato è descritto da 52 attributi (31 categorici, 6 binari, 14 numerici e un attributo nullo)

**Task** L'obiettivo è **prevedere se un individuo viene nuovamente arrestato entro due anni dal primo arresto**

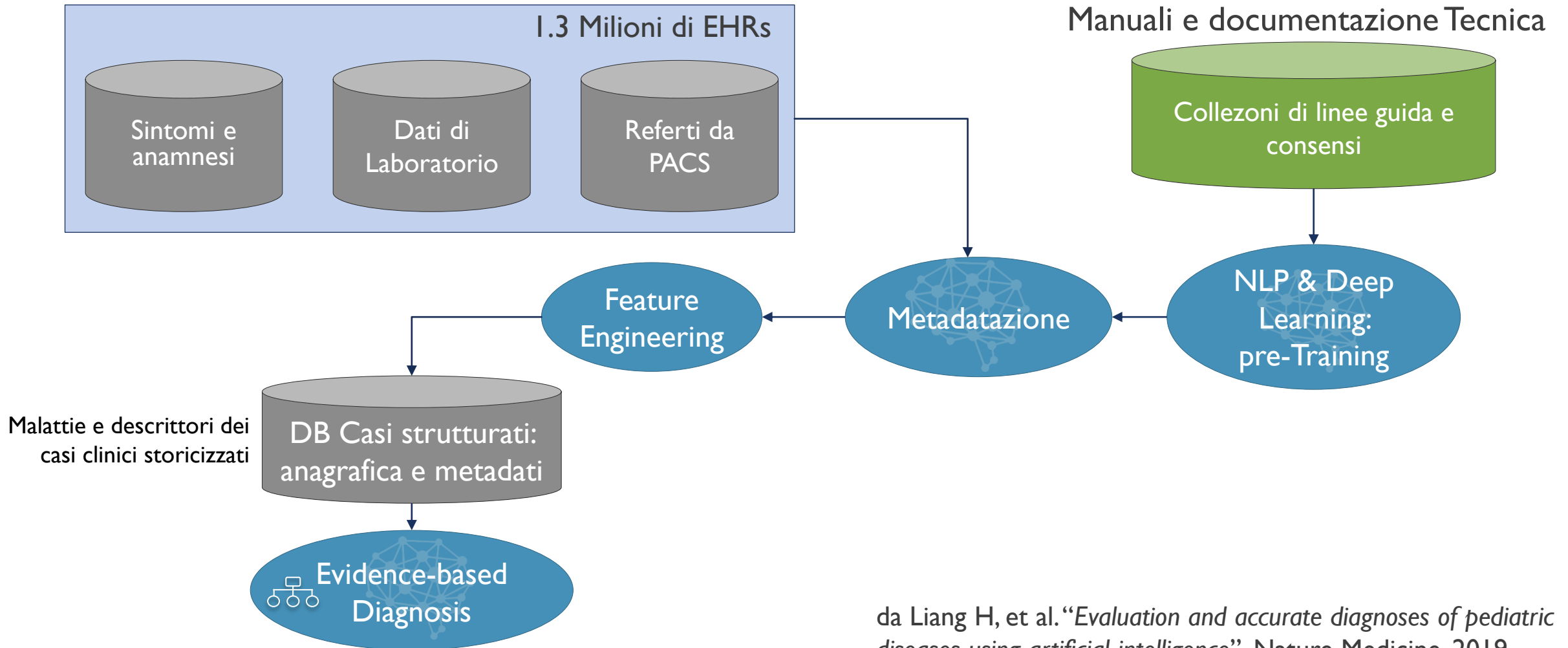
### Possibili rischi

Alcuni gruppi sociali (gli afroamericani) hanno maggiori probabilità di essere erroneamente etichettati come a rischio più elevato rispetto agli altri (i caucasici). Eticamente ingiusto.

Obiettivo: ottenere un sistema equo tra gruppi sociali diversi.

<https://github.com/propublica/compas-analysis>

# DIAGNOSI MALATTIE PEDIATRICHE: UN WORKFLOW ORIENTATO AL ML



da Liang H, et al. "Evaluation and accurate diagnoses of pediatric diseases using artificial intelligence", Nature Medicine, 2019

# MEDICAL INFORMATION EXTRACTION

INPUT: "Si **osserva una lesione** nel **lobo superiore sinistro** del **polmone del paziente** ..."

INPUT

Word Embedding

LSTM

QUERIES

Q : Is the left upper lobe of the lung detectable?

Q: Is there a mass in the left upper lobe?

Q: Is there a detectable lesion in the left upper lobe?

Q: Is there a detectable obstruction in the bronchus?

Q : Is there an abnormality in the bronchus?

ANSWERS

A : 1

A : 1

A : 1

A : 0

A : 0



# EVIDENCE BASED DIAGNOSIS: RISULTATI (11,926 PAZIENTI)

**Table 2 | Illustration of diagnostic performance of our AI model and physicians**

Disease conditions	Our model	Physicians				
		Physician group 1	Physician group 2	Physician group 3	Physician group 4	Physician group 5
Asthma	0.920	0.801	0.837	0.904	0.890	0.935
Encephalitis	0.837	0.947	0.961	0.950	0.959	0.965
Gastrointestinal disease	0.865	0.818	0.872	0.854	0.896	0.893
Group: 'Acute laryngitis'	0.786	0.808	0.730	0.879	0.940	0.943
Group: 'Pneumonia'	0.888	0.829	0.767	0.946	0.952	0.972
Group: 'Sinusitis'	0.932	0.839	0.797	0.896	0.873	0.870
Lower respiratory	0.803	0.803	0.815	0.910	0.903	0.935
Mouth-related diseases	0.897	0.818	0.872	0.854	0.896	0.893
Neuropsychiatric disease	0.895	0.925	0.963	0.960	0.962	0.906
Respiratory	0.935	0.808	0.769	0.89	0.907	0.917
Systemic or generalized	0.925	0.879	0.907	0.952	0.907	0.944
Upper respiratory	0.929	0.817	0.754	0.884	0.916	0.916
Root	0.889	0.843	0.863	0.908	0.903	0.912
<b>Average F1 score</b>	<b>0.885</b>	<b>0.841</b>	<b>0.839</b>	<b>0.907</b>	<b>0.915</b>	<b>0.923</b>

# REFERENCES: AI TIMELINE, AGENTS AND KNOWLEDGE REPRESENTATION

- Charles Babbage, Note on the application of machinery to the computation of astronomical and mathematical tables, 1823.
- G. Boole, The Mathematical Analysis of Logic, 1848.
- G. Boole, An Investigation of the Laws of Thought, 1854.
- Frege, G. (1879). *Begriffsschrift, eine der arithmetischen nachgebildete Formelsprache des reinen Denkens*. Halle, Berlin.
- Turing, A. M. (1936). On computable numbers, with an application to the *entscheidungsproblem*. Proceedings of the London Mathematical Society, 2nd series, 42:230-265. Correction published in Vol. 43, pages 544-546.
- Turing, A. M. (1950). Computing machinery and intelligence. *Mind*, 59:433-460.
- Colmerauer, A. (1978). Metamorphosis grammars. In Bolc, L., editor, *Natural Language Communication with Computers*. Springer-Verlag, Berlin. English translation of [Colmerauer, 1975]
- Brachman, R. J. and Levesque, H. J., editors (1985). *Readings in Knowledge Representation*. Morgan Kaufmann, San Mateo, California.
- Genesereth, M. R. and Nilsson, N. J. (1987). *Logical Foundations of Artificial Intelligence*. Morgan Kaufmann, San Mateo, California.
- Quillian, M. R. (1968). Semantic memory. In Minsky, M. L., editor, *Semantic Information Processing*, pages 216-270. MIT Press, Cambridge, Massachusetts.
- Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning*, 1:81-106.
- Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning internal representations by error propagation. In Rumelhart, D. E. and McClelland, J. L., editors, *Parallel Distributed Processing*, volume 1, chapter 8, pages 318-362. MIT Press, Cambridge, Massachusetts.
- Russell S., Norvig, P., *Artificial Intelligence, A modern Approach, the intelligent agent book*. Prentice Hall, 2003-2020
- Ian Goodfellow and Yoshua Bengio and Aaron Courville, *Deep Learning*, MIT Press, <http://www.deeplearningbook.org>, 2016}





GRAZIE  
DELL'ATTENZIONE

[BASILI@INFO.UNIROMA2.IT](mailto:BASILI@INFO.UNIROMA2.IT)