



UTN – FRVM

Agenda



- Docentes, Objetivos, Metodología de Trabajo, Regularización, Bibliografía.
- Preliminares IA. Problema. Naturaleza. Inteligencias Múltiples.
- Pre-Historia. Nacimiento. Evolución. Actualidad. Técnicas.

Docentes



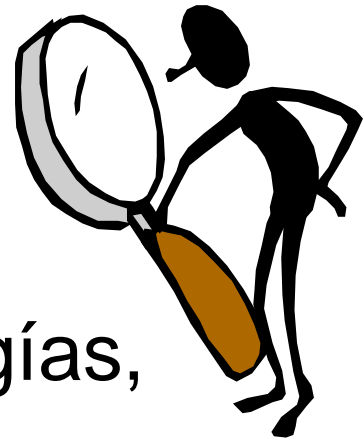
- **Prof. Adj.:** Dr. Jorge A. Palombarini
- **JTP:** Ing. Juan Cruz Barsce
- **Colaborador Alumno:** Cecilia Cabaña
- Inteligenciafrvm at gmail dot com
- <https://github.com/inteligenciafrvm/inteligenciafrvm>
- <https://cvirtual.frvn.utn.edu.ar/course/view.php?id=253>
- **Consultas fuera del ámbito de Clase:** Lunes de 10:00 a 12:00 hs.



Objetivo General de la Materia

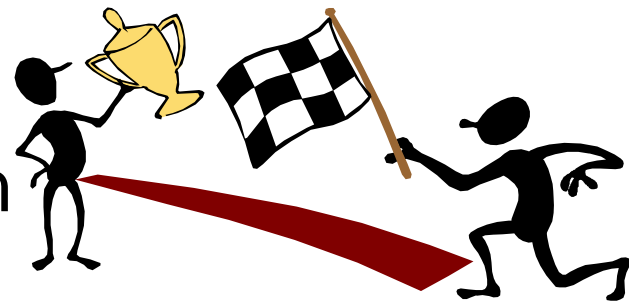
- La asignatura brinda un panorama general de los **problemas** que aborda la Inteligencia Artificial
- Proporciona **fundamentos y técnicas** para el desarrollo de Sistemas Inteligentes y Sistemas Cognitivos.
- En ese marco, se analizan problemáticas relacionadas con la **generación, representación, abstracción y generalización** de conocimiento y su integración en agentes y sistemas autónomos.
- Al abordar el estudio de los Sistemas Inteligentes (estructura, dominios apropiados) y Sistemas Cognitivos, se espera incorporar aspectos prácticos del área a través del **desarrollo de prototipos software**.

Metodología de Trabajo



- Clase Teórica. Conceptos, Metodologías, Algoritmos
- Resolución de Casos de Estudio Prácticos: Programación de Prototipos, Lectura de papers proporcionados por la cátedra, debate en clase.
- Uso de Herramientas Computacionales (Python).

Regularización y Aprobación



- Contemplar la Ordenanza 1549/16 CS UTN

Aprobación Directa	Regularización
<ul style="list-style-type: none">- Cumplir con las condiciones de Aprobación Directa de la Parte Práctica	<ul style="list-style-type: none">- Cumplir con las condiciones de Regularidad de la Parte Práctica
<ul style="list-style-type: none">- Aprobar un parcial Teórico con nota igual o superior a 8.<ul style="list-style-type: none">- Fecha: 03/07/2020- Recuperatorio: 08/07/2020	
<ul style="list-style-type: none">- Asistir al 75% de las clases.	<ul style="list-style-type: none">- Asistir al 75% de las clases.

Bibliografía

- Ver Programa de la Materia en Github



Agenda



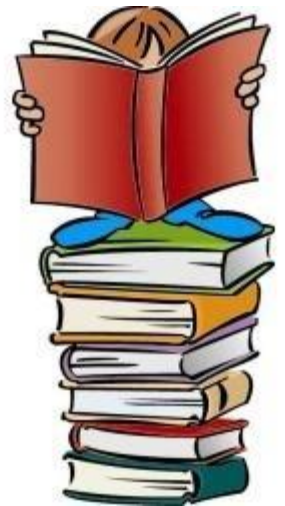
- Docentes, Metodología de Trabajo, Regularización, Bibliografía.
- Preliminares IA. Problema. Naturaleza.
- Pre-Historia. Nacimiento. Evolución. Técnicas. Cognitive Systems and Architectures

IA ≠ Tecnología

<http://simpleprogrammer.com/2015/03/16/11-rules-all-programmers-should-live-by/>

11 Rules All Programmers Should Live By

“1- Technology is how you get to the solution, it is not THE solution.”



Preliminares: IA



- “La IA tiene como propósito ***reproducir las acciones y el razonamiento*** de los seres vivos inteligentes en dispositivos artificiales”
- “La inteligencia artificial, es la rama de la ciencia computacional que investiga procesos simbólicos, razonamientos no algorítmicos y representaciones simbólicas de conocimientos usados en ***máquinas con desempeño inteligente*** que imitan o simulan el comportamiento y la actividad inteligente del ser humano”

IA

- La IA busca identificar procedimientos, métodos y técnicas que se asocian a la capacidad de ***pensar y razonar de manera inteligente.***
- Se pretende ***automatizar las tareas del pensamiento y razonamiento humano,*** proveyendo un modelo cognitivo de su funcionamiento.



¿Por qué estudiar IA?



- Plantear modelos computacionales de la inteligencia.
- Crear programas que lleven a cabo funciones para las cuales se requiere ***inteligencia***.
- Mejorar habilidades para el modelado y resolución general de problemas.
- Agregar valor a productos software.

Problemas



- Los seres humanos no saben realmente cómo realizan sus actividades.
- Las computadoras no se enfrentan a esas tareas de la misma manera que los seres humanos, dado que tienen que estar previamente programadas en lenguajes en los que sólo es posible expresar conceptos muy elementales. (Problema de representación)

● ● ● | Problema Principal

**¿Qué es la
INTELIGENCIA?**





Naturaleza de la Inteligencia

- En el estudio de los sistemas cognitivos cabe diferenciar dos perspectivas: la de la ciencia cognitiva que pretende elaborar una teoría general de los sistemas cognitivos o inteligentes, sean éstos naturales o artificiales;
- y la de la psicología cognitiva que se centra en el estudio de los sistemas inteligentes de los seres vivos, especialmente el ser humano.

Ficción Vs. Realidad



Más Ficción...





Preguntas



- Que capacidades tienen las AI's en la ciencia ficción?
- Cuáles de ellas aun no se han hecho realidad?
- Son auto-conscientes las AI's que aparecen en la ficción?



Respuestas



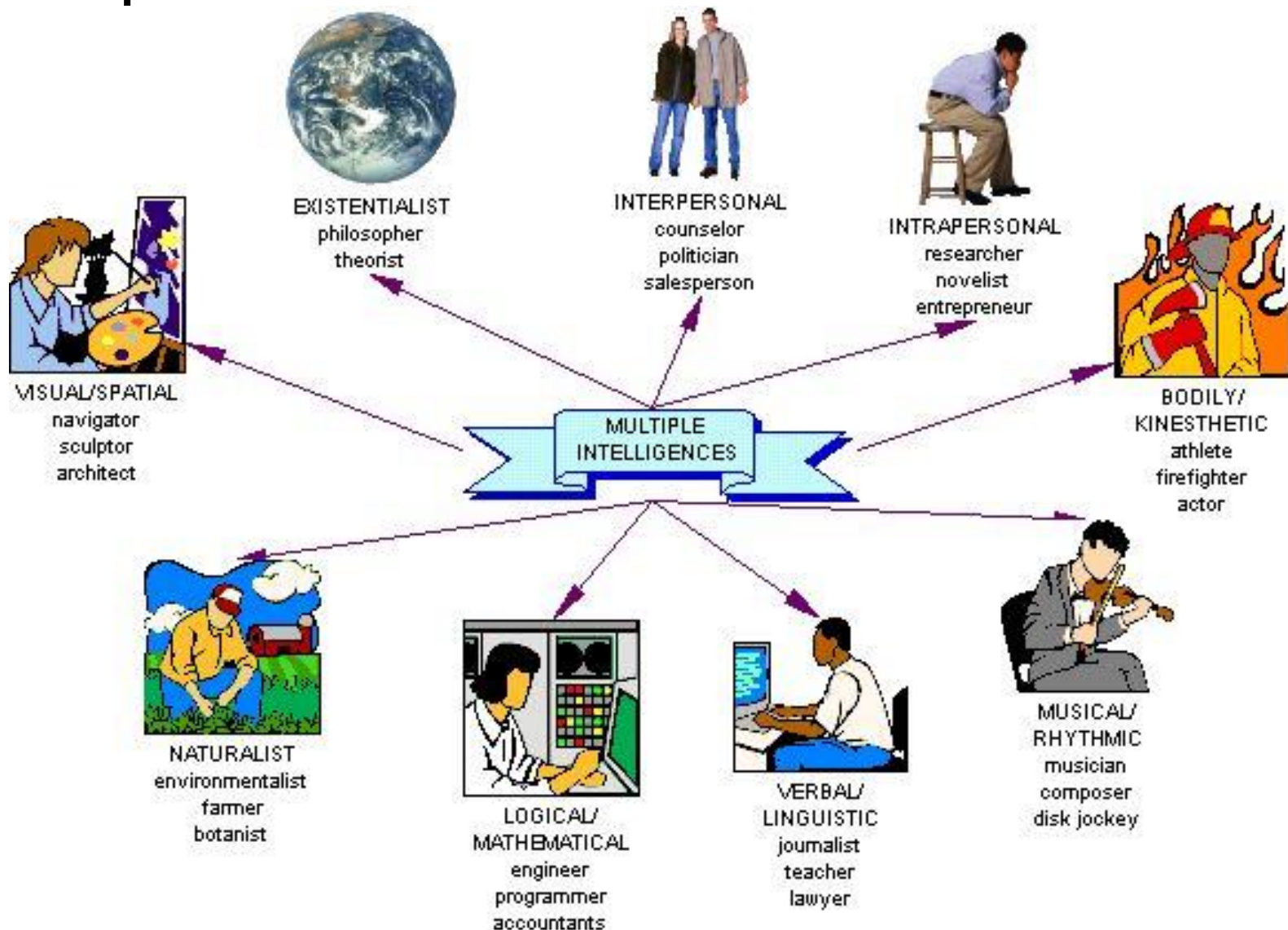
- Razonar, crear estrategias, resolver enigmas, emitir juicios bajo incertidumbre.
- Representar conocimiento respecto de objetos, acciones, situaciones o sentido común.
- Planificar.
- Aprender.
- Comunicarse en lenguaje natural.
- **Integrar todas las capacidades anteriores para lograr objetivos.**

● ● ● | Inteligencia



- Surge y se desarrolla a lo largo de la evolución de las especies ligada a la organización cada vez más compleja del sistema nervioso y de la corticalización y encefalización crecientes.
- Precisamente el grado de autonomía es un buen índice de la inteligencia de cada especie. Esta evolución tienen su nivel más alto en el hombre.

● ● ● | Inteligencias Múltiples



¿Qué es la Inteligencia?



- Habilidad de un sistema para **actuar apropiadamente** en entornos inciertos donde una acción apropiada es aquella que incrementa la probabilidad de **éxito**. **Éxito es la consecución de metas u objetivos** derivados del propósito del sistema. (D. Wechsler).
- Habilidad de enfrentar problemas de manera creativa, es decir, **no programada**. (S. J. Gould)
- Capacidad de aprender o mejorar a partir de la experiencia. (W. F. Dearborn)



¿Qué es la Inteligencia? (II)

- Habilidad de adaptarse adecuadamente a nuevas situaciones de la vida. (R. Pinter)
- Habilidad de un organismo para resolver nuevos problemas. (W. V. Bingham)
- Los individuos difieren entre sí en su habilidad para entender ideas complejas, adaptarse efectivamente al entorno, aprender de la experiencia, desarrollar nuevas formas de razonamiento y vencer obstáculos a través del pensamiento. (*American Psychological Association*)



¿Qué es la Inteligencia? (III)

- Capacidades cognitivas que habilitan a un individuo a adaptarse y prosperar en cualquier entorno en el que se encuentre. Tales capacidades incluyen aspectos como memoria, realimentación y resolución de problemas. (D. K. Simonton)
- Leer Paper “***A Collection of Definitions of Intelligence***” (Legg) y A Formal Measure of Machine Intelligence (Legg & Hutter)



Inteligencia para sistemas artificiales

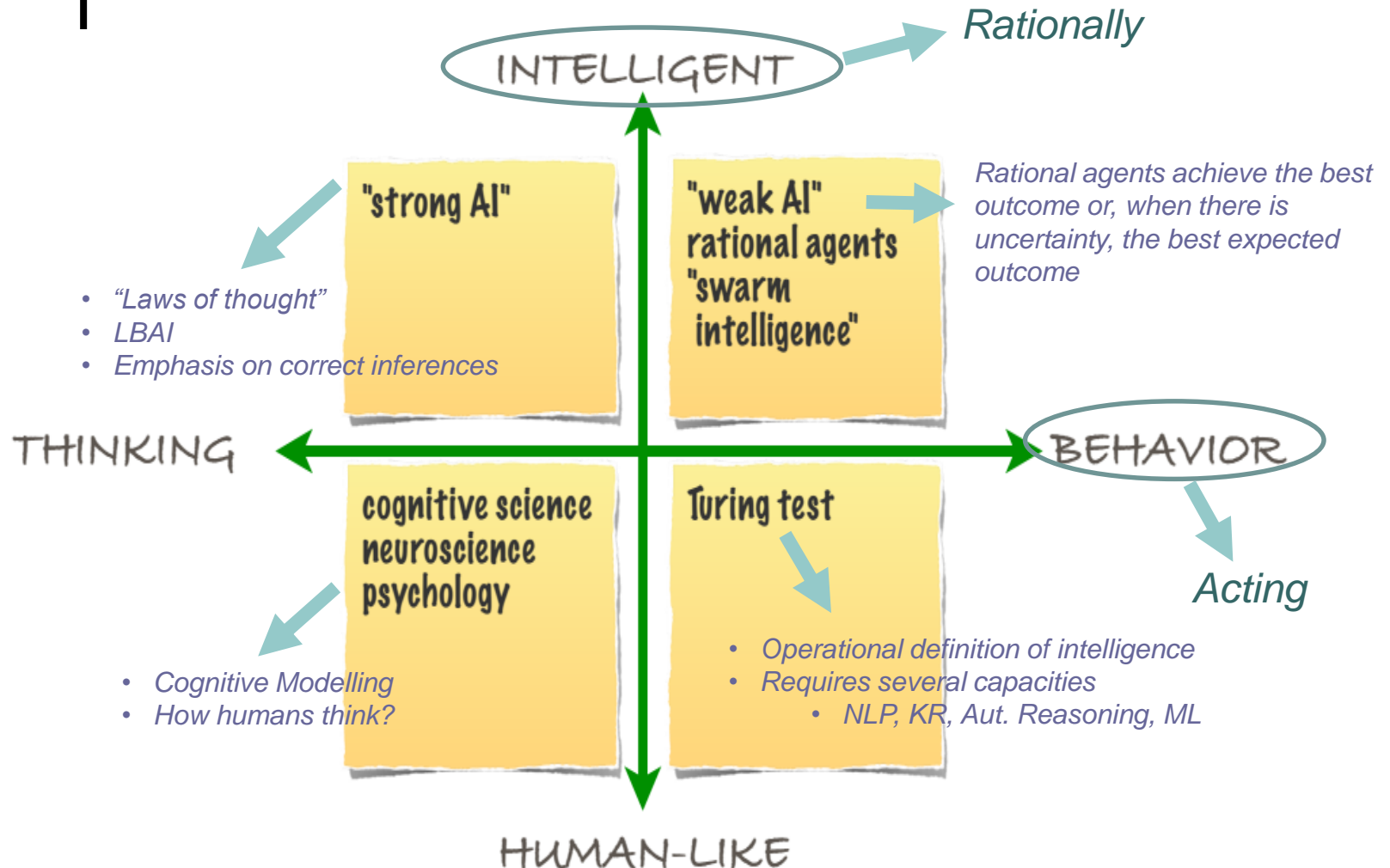
- Ingredientes clave para una definición:
- La inteligencia es una **propiedad** de una entidad (artificial) llamada “Agente”, la cuál **interactúa** con un entorno externo, problema o situación.
- La Inteligencia de tal agente estará relacionada con su **habilidad para tener éxito** en un **entorno**.
- La Inteligencia del agente no será entonces la habilidad para tratar con un entorno fijo y conocido, sino la capacidad de interactuar con un **rango de posibilidades** que no pueden ser **totalmente anticipadas**.

Inteligencia: Definición



- **“La inteligencia mide la habilidad que posee un Agente para alcanzar objetivos o metas en un amplio rango de entornos”**

Filosofía de la IA



Agenda

- Docentes, Metodología de Trabajo, Regularización, Bibliografía.
- Preliminares IA. Problema. Naturaleza. Inteligencias Múltiples.
- Pre-Historia. Nacimiento. Evolución. Técnicas. Cognitive Systems and Architectures



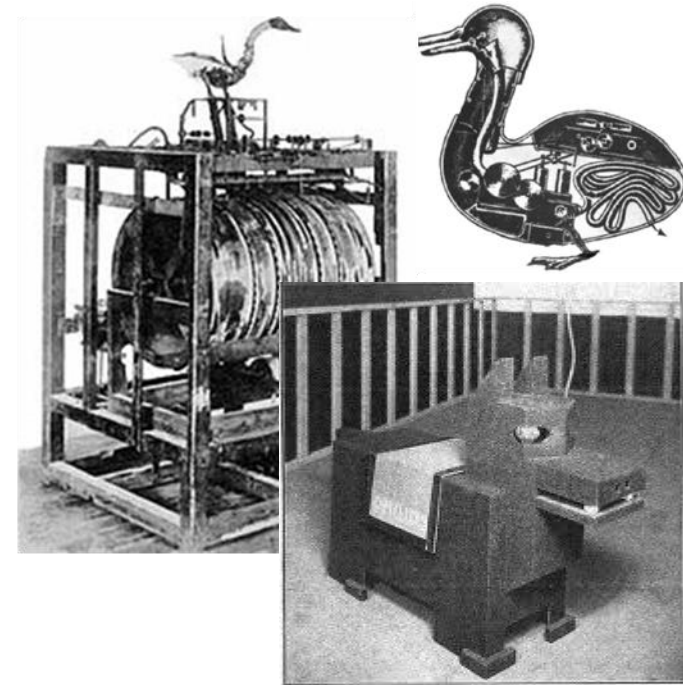
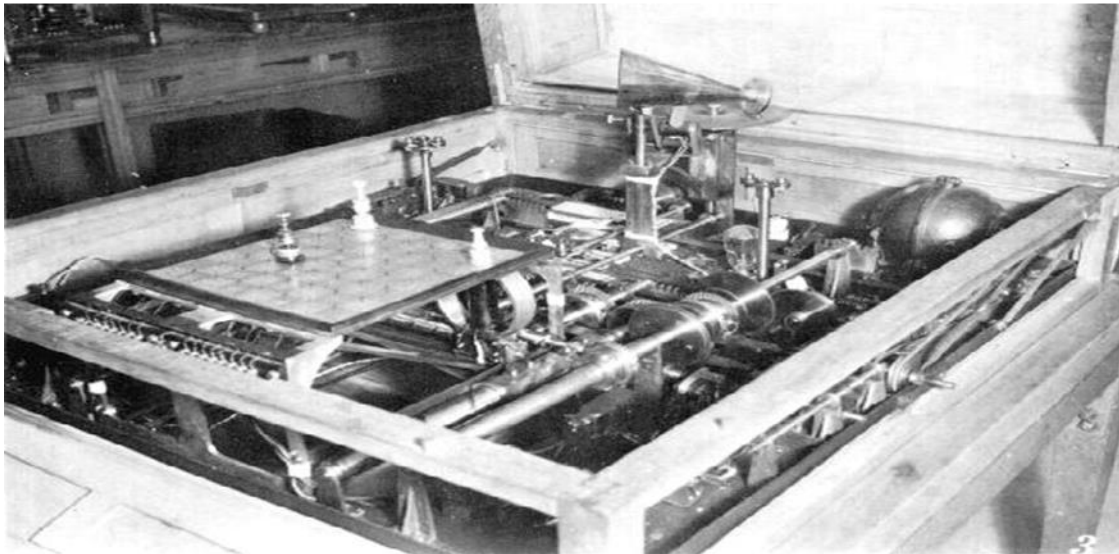
Un poco de (pre)historia

- Desde los primeros tiempos de nuestra civilización hasta mediados del siglo XX.
- **Construcción de autómatas** que simulaban desde el punto de vista externo el comportamiento humano o animal.
- Información y **automatización del razonamiento lógico** y matemático.



Pre-Historia. Autómatas

- Jacques de Vaucanson. El Flautista (1737), El pato (1738)
- 400 partes móviles, podía batir sus alas, beber agua, digerir grano)
- Torres Quevedo construyó dos versiones de "El jugador ajedrecista" (1912 y 1920)
- Kapek difunde en los años 20 la palabra "robot"



Pre-Historia. Automatización del Razonamiento

- Ramon Llull ("Ars Magna"), describe una gramática universal ("un método científico" de discusión).
- Leibnitz. Álgebra universal que permitiera deducir todas las verdades.
- Estudios matemáticos de Russell y Hilbert de comienzos de siglo. Reducir el razonamiento a la manipulación abstracta de cadenas de símbolos. (KBS y Representación de Conocimiento).
- Las teorías de la **computabilidad** y de los **autómatas** proporcionan el vínculo entre la formalización del razonamiento y las máquinas que estaban a punto de surgir tras la Segunda Guerra Mundial.



Prehistoria (WWII)

- Entre 1939 y 1943, en Bletchley Park, Gran Bretaña creó la escuela gubernamental de códigos y cifrado (GC&CS) en, para quebrar el tráfico de mensajes enemigos (Máquinas Enigma)
- Matemáticos y criptógrafos británicos, entre ellos [Alan Turing](#), jugadores de ajedrez y bridge y fanáticos de los crucigramas, se enfrentaron a los problemas presentados por las muchas variaciones alemanas del Enigma.



Gestación de la IA (1943-1955)



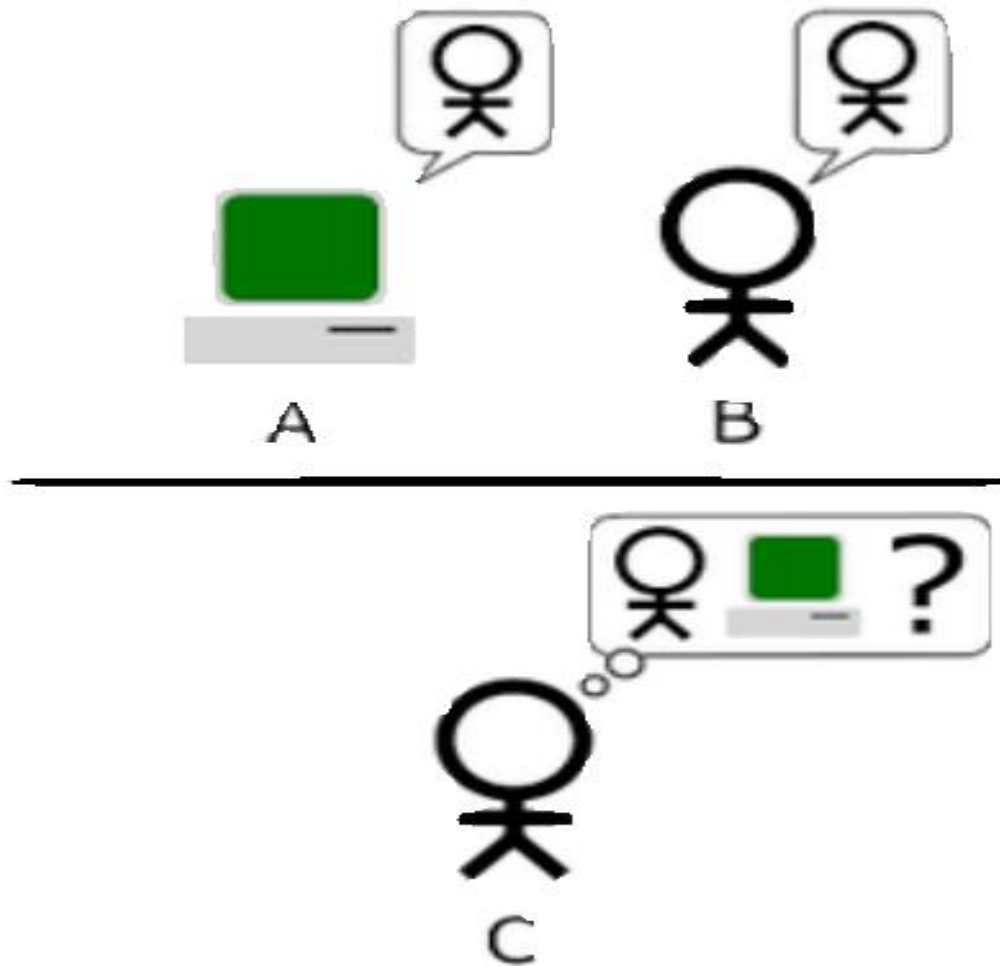
- “Cerebros electrónicos” (Von Neumann.)
- Nuevos conceptos como feedback, el control y los sistemas auto-organizados.
- La cibernética ligó la fisiología neuronal, la teoría de la información de Shannon, la lógica matemática y la naciente tecnología informática.
- El primer trabajo reconocido como IA fue realizado por Warren McCulloch y Walter Pitts (1943).
- **Hebbian learning.**
- Minsky construye la primera máquina neuronal durante su doctorado.

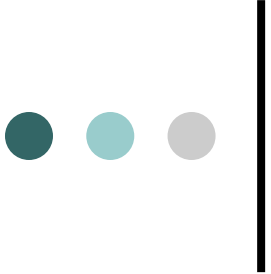
Gestación de la IA (1943-1955)



- En 1945 Vannevar Bush publicó “As We May Think” una visión del futuro en donde los ordenadores asistían a los humanos en muchas actividades.
- Alan Turing presentó un paper sobre la IA, titulado “**COMPUTING MACHINERY AND INTELLIGENCE**”. (*“I propose to consider the question, “Can machines think?”*). *Intelligence Machinery* (1948)
- Propone un “Test” para determinar el carácter inteligente o no de una máquina. *Imitation Game*. También propone la idea del “**programa niño**”, es decir, el lugar de intentar producir un programa que simule la mente de un adulto, por qué no producir un programa que imite la mente de un niño y que tenga capacidad de aprender?

The Turing Test





Nacimiento y expectativas (1956-1969)

- En 1956 fue creado el el IPL-II (búsqueda heurística) por Newell, Shaw y Simon. (Primer lenguaje especializado de la IA)
- En 1956 tuvo lugar la “Conferencia del Dalmouth” en Hanover (New Hampshire) sobre IA, organizada por J. McCarthy (Stanford), M. Minsky (MIT), A. Newell y H. Simon (CM) y personal de IBM.
- En Dartmouth, Newell y Simon ya disponían de un programa pro-IA, el “Logic Theorist” que resolvía problemas de búsqueda heurística, junto con los principios matemáticos de Whitehead y Rusell. (***El Journal of Symbolic Logic rechazó el paper***).
- A mediados de los años 50 John McCarthy y posteriormente el MIT, diseñaron el lenguaje LISP (List Processing).

1956 - 1969



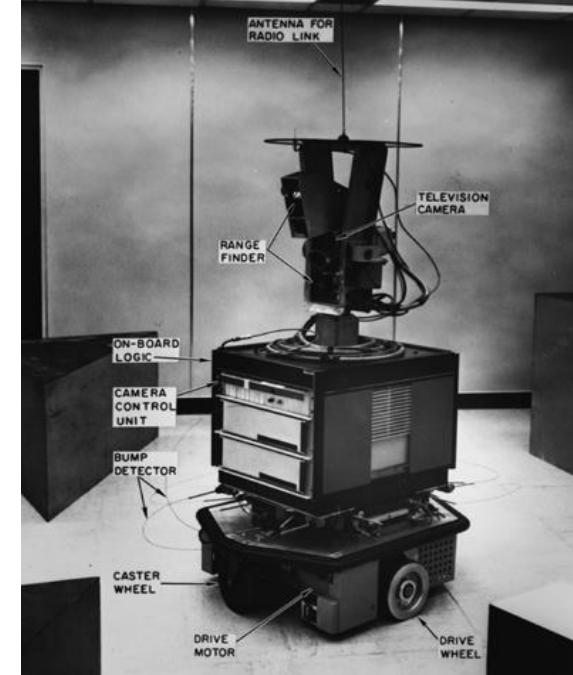
- **Logic Theorist y GPS.** 1956. Fueron programados por Newell, Simon y Shaw. Demostraciones de Teoremas Lógicos.
- **GPS** realiza ensayos hasta encontrar la manera de usar los medios para alcanzar el objetivo (análisis medios-fines). (Partial Order Planning, STRIPS, Situation Calculus)
- **Perceptron.** F. Rosenblatt. Imitaba el funcionamiento del ojo humano. Dispositivo de aprendizaje: en su configuración inicial no tenía la capacidad de distinguir patrones de entrada muy complejos, sin embargo mediante un proceso de aprendizaje era capaz de hacerlo.
- Incapacidad para solucionar problemas que no sean linealmente separables.
- A pesar de esta limitación, el Perceptrón es aún hoy una red de gran importancia, pues con base en su estructura se han desarrollado otros modelos de red neuronal.

1956 - 1969

- **LISP(LISt Procesing)**. Lenguaje declarativo funcional.
(McCarthy en MIT)

SAINT(Symbolic Automatic INTegrator).

GTP(Geometry Theorem Prover)



- **Programa para jugar a las damas**. Arthur Samuel auspiciado por IBM. El programa era capaz de aprender por si solo. Alex Bernstein, consiguió también un resultado similar para el ajedrez.
- **STUDENT y SIR**. (1964). Tesis Doctoral de Daniel Bobrow. STUDENT era capaz de entender enunciados de problemas algebraicos escritos en inglés y responder a sus preguntas. Bertram Raphael diseña SIR (Semantic Information Retrieval). **ELIZA (o DOCTOR)**. **DENDRAL**. El primer sistema experto (1965)
- **SHAKEY**. 1969. Espía mecánico, capaz de infiltrarse en las líneas enemigas para conseguir información (?) (<http://www.ai.sri.com/shakey/>).
- Minsky llega al MIT y se enfrenta a las ideas de McCarthy (1958)



1970 - 1979

- La IA se enfrenta a sus primeras dificultades: el problema del conocimiento de background, poco éxito en la traducción de texto científico, y la intratabilidad intrínseca de los problemas de gran escala (Teoría de la complejidad). Lighthill report (1973)
- Nacimiento de los **sistemas expertos**. **MYCIN** (Stanford, 1976) Incorpora cálculo de incertidumbre. **PROSPECTOR** (1978,1979).
- McCarthy's Advice Taker approach—Separar claramente el conocimiento en forma de reglas y el conocimiento.
- Primer Congreso Internacional de Inteligencia Artificial. Primer numero de la revista Artificial Intelligence.
- En 1972 se crea PROLOG en Marsella, Francia.
- **XCON**(eXpert CONfigurer, 1979) Primer **ES** con aplicaciones en el mundo real fuera de los ámbitos académicos y de los laboratorios de investigación. Creado por DEC, configura todos los VAX que salen de la empresa (Industria de los ES's)

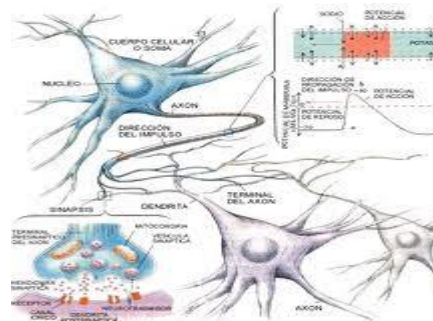
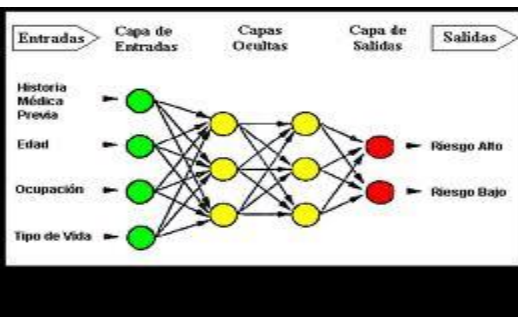


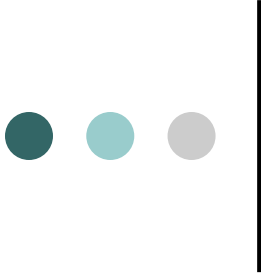
1980 – principios de los 90's

- Éxito de los sistemas expertos. Empresas especializadas en ofrecer soluciones comercializables basados en las técnicas de la IA.
- SHELLS (herramientas de desarrollo de sistemas expertos). DELTA (GE Company), Aldo y EURISKO.
- XCON llegó a tener 10.000 reglas, por lo que DEC tenía que gastar anualmente 2 millones de dólares en el mantenimiento de la Base de Conocimientos.
- Quinta generación de ordenadores (Japón) -> avances en procesamiento paralelo, procesamiento del lenguaje natural, programación lógica y en el tratamiento de la información, los computadores ópticos y la facilidad de uso de los ordenadores.

1980 – principios de los 90's

- En 1985 la NASA crea CLIPS. Generador de sistemas expertos que implementa un lenguaje propio.
- Cuenta con tres paradigmas de programación: orientado a reglas, procedural y orientado a objetos.
- A partir de 1986 se recuperan los viejos esquemas del Perceptron en la Redes Neuronales.

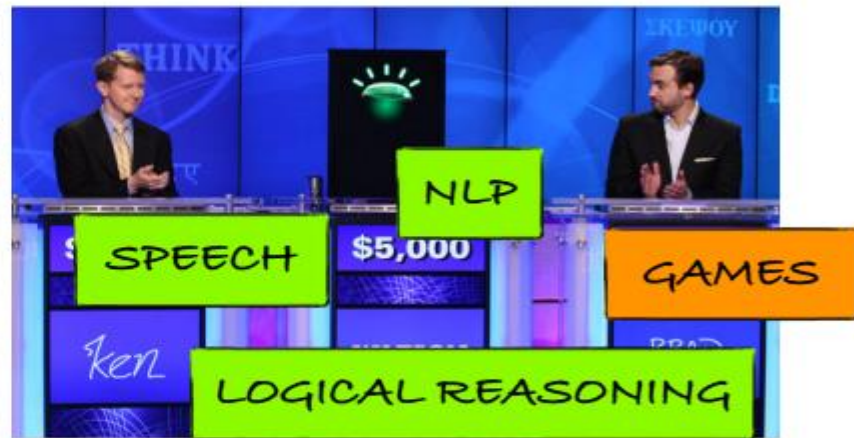




1995 – 2005 (Agentes Inteligentes)

- Newell, Laird y Rosembloom crean las bases de SOAR
- Descontento por parte de los fundadores de la IA respecto de las direcciones que toman los desarrollos recientes.
- *“machines that think, that learn and that create”*
- **Artificial General Intelligence: algoritmo universal para aprender y actuar en cualquier entorno.**

Actualidad



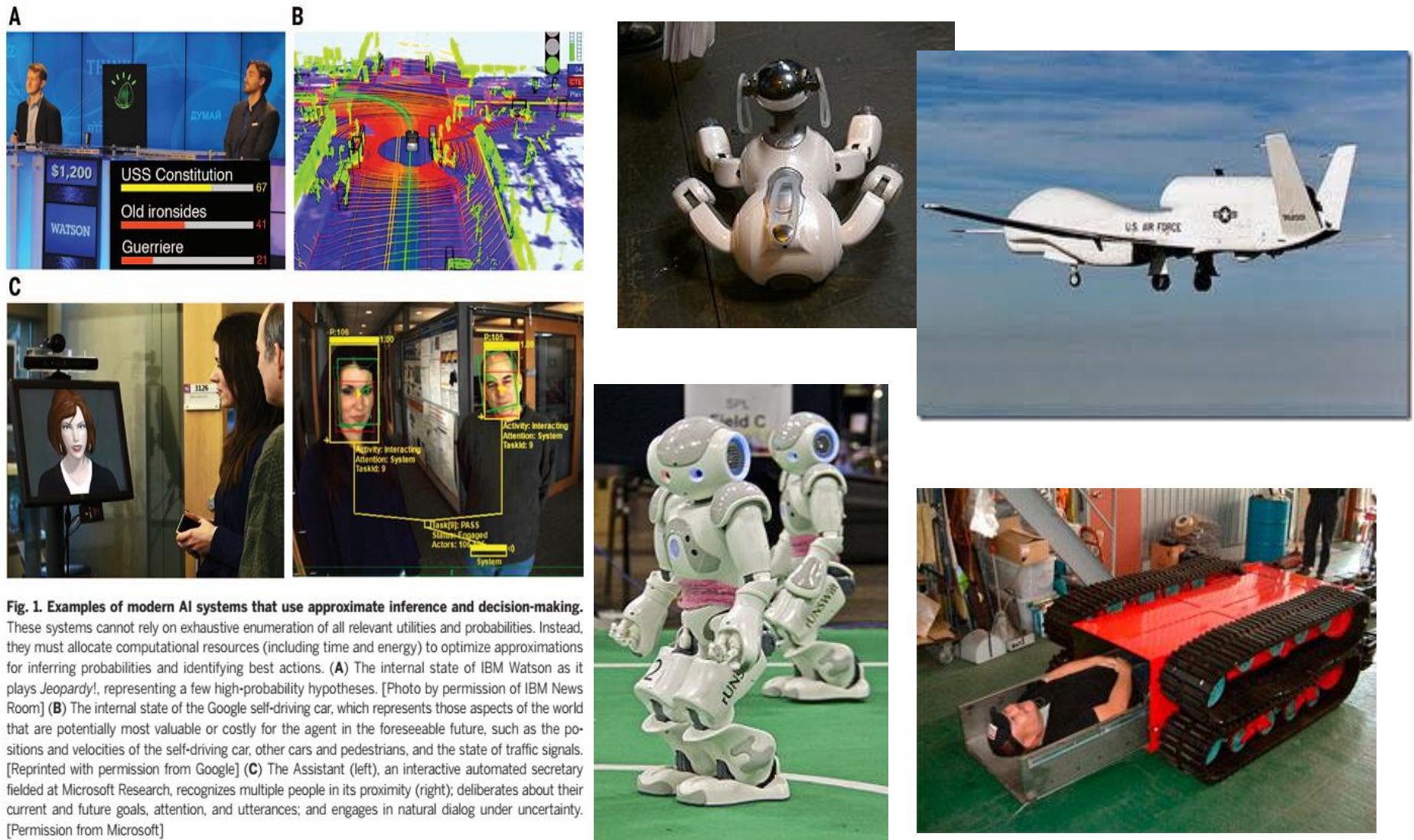
Actualidad



Java, C++, Prolog, Apache Hadoop, SUSE Linux

Millions of documents, including dictionaries, encyclopedias, and other reference material that it could use to build its knowledge. 200 million pages of structured and unstructured content consuming four terabytes of disk storage, including the full text of Wikipedia.

Actualidad



Actualidad

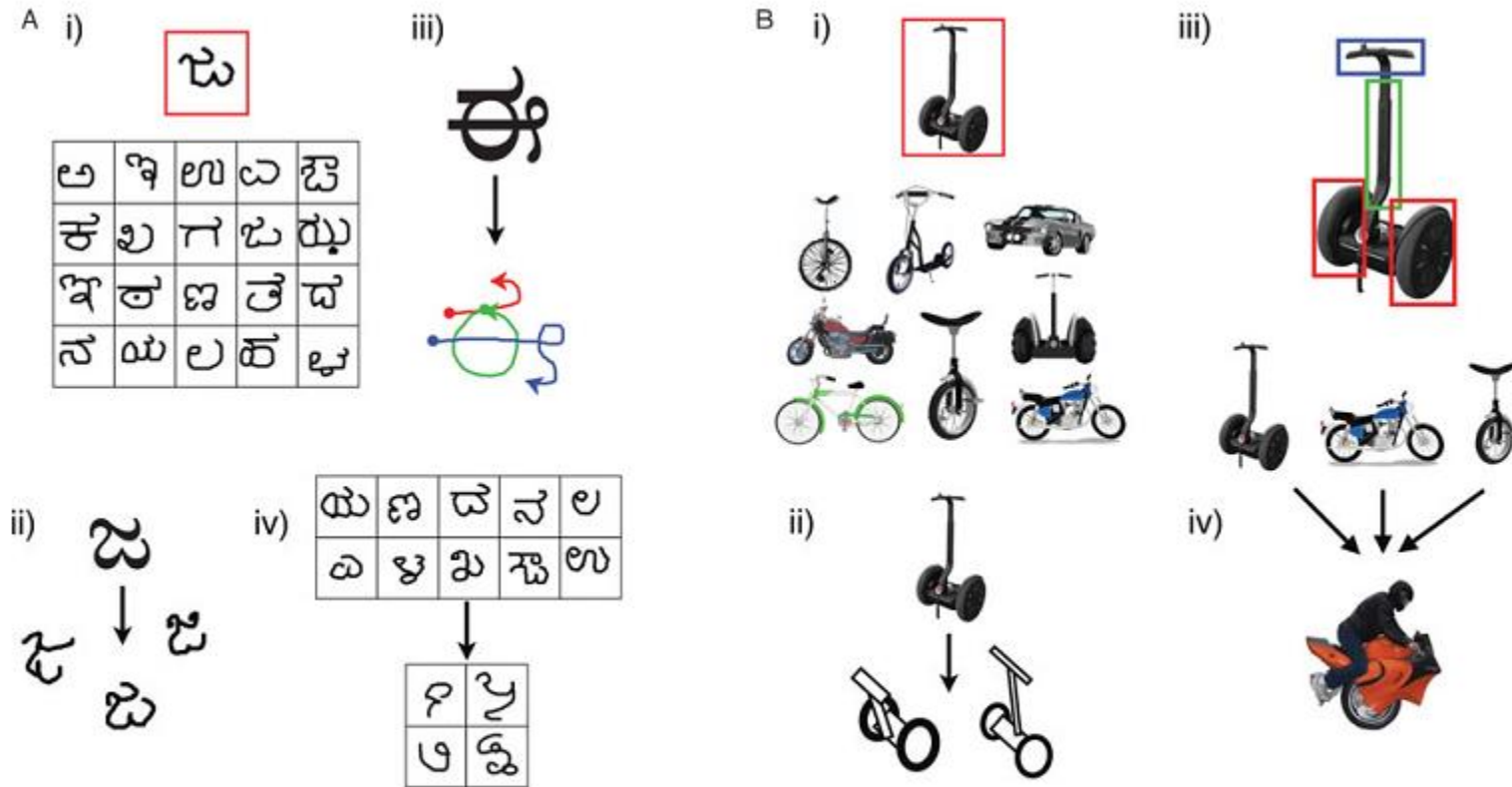
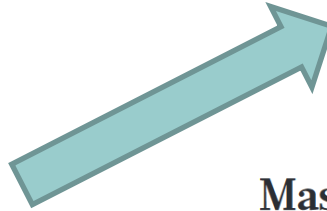


Figure 1. The Characters Challenge: Human-level learning of novel handwritten characters (A), with the same abilities also illustrated for a novel two-wheeled vehicle (B). A single example of a new visual concept (*red box*) can be enough information to support the (i) classification of new examples, (ii) generation of new examples, (iii) parsing an object into parts and relations, and (iv) generation of new concepts from related concepts. Adapted from Lake et al. (2015a).

Actualidad



LETTER

doi:10.1038/nature14236

Human-level control through deep reinforcement learning

Volodymyr Mnih^{1*}, Koray Kavukcuoglu^{1*}, David Silver^{1*}, Andrei A. Rusu¹, Joel Veness¹, Marc G. Bellemare¹, Alex Graves¹, Martin Riedmiller¹, Andreas K. Fiedelnd¹, Georg Ostrovski¹, Stig Petersen¹, Charles Beattie¹, Amir Sadik¹, Ioannis Antonoglou¹, Helen King¹, Dharshan Kumaran¹, Daan Wierstra², Shane Legg² & Demis Hassabis¹

The theory of reinforcement learning provides a normative account¹, deeply rooted in psychological² and neuroscientific³ perspectives on animal behaviour, of how agents may optimize their control of an environment. To use reinforcement learning successfully in situations approaching real-world complexity, however, agents are confronted with a difficult task: they must derive efficient representations of the environment from high-dimensional sensory inputs, and use these to generalize past experience to new situations. Remarkably, humans and other animals seem to solve this problem through a harmonious combination of reinforcement learning and hierarchical sensory processing systems^{4,5}, the former evidenced by a wealth of neural data revealing notable parallels between the phasic signals emitted by dopa-

agent is to select actions in a fashion that maximizes cumulative future reward. More formally, we use a deep convolutional neural network to approximate the optimal action-value function

$$Q^{\pi}(s,a) = \max_{\pi} \mathbb{E}[r_1 + \gamma r_{t+1} + \gamma^2 r_{t+2} + \dots | s_t = s, a_t = a, \pi],$$

which is the maximum sum of rewards r_t discounted by γ at each time-step t , achievable by a behaviour policy $\pi \approx P(a|s)$, after making an observation (s) and taking an action (a) (see Methods)¹⁶.

Reinforcement learning is known to be unstable or even to diverge when a nonlinear function approximator such as a neural network¹⁷ is used to represent the action-value (also known as Q) function¹⁸. This instability has several causes: the correlations present in the sequence

Mastering the game of Go with deep neural networks and tree search

David Silver^{1*}, Aja Huang^{1*}, Chris J. Maddison¹, Arthur Guez¹, Laurent Sifre¹, George van den Driessche¹, Julian Schrittwieser¹, Ioannis Antonoglou¹, Veda Panneershelvam¹, Marc Lanctot¹, Sander Dieleman¹, Dominik Grewe¹, John Nham¹, Nal Kalchbrenner¹, Ilya Sutskever², Timothy Lillicrap¹, Madeleine Leach¹, Koray Kavukcuoglu¹, Thore Graepel¹ & Demis Hassabis¹

The game of Go has long been viewed as the most challenging of classic games for artificial intelligence owing to its enormous search space and the difficulty of evaluating board positions and moves. Here we introduce a new approach to computer Go that uses 'value networks' to evaluate board positions and 'policy networks' to select moves. These deep neural networks are trained by a novel combination of supervised learning from human expert games, and reinforcement learning from games of self-play. Without any lookahead search, the neural networks play Go at the level of state-of-the-art Monte Carlo tree search programs that simulate thousands of random games of self-play. We also introduce a new search algorithm that combines Monte Carlo simulation with value and policy networks. Using this search algorithm, our program AlphaGo achieved a 99.8% winning rate against other Go programs, and defeated the human European Go champion by 5 games to 0. This is the first time that a computer program has defeated a human professional player in the full-sized game of Go, a feat previously thought to be at least a decade away.

ARTICLE

doi:10.1038/nature24270

Mastering the game of Go without human knowledge

David Silver^{1*}, Julian Schrittwieser^{1*}, Karen Simonyan^{1*}, Ioannis Antonoglou¹, Aja Huang¹, Arthur Guez¹, Thomas Hubert¹, Lucas Baker¹, Matthew Lai¹, Adrian Bolton¹, Yutian Chen¹, Timothy Lillicrap¹, Fan Hui¹, Laurent Sifre¹, George van den Driessche¹, Thore Graepel¹ & Demis Hassabis¹

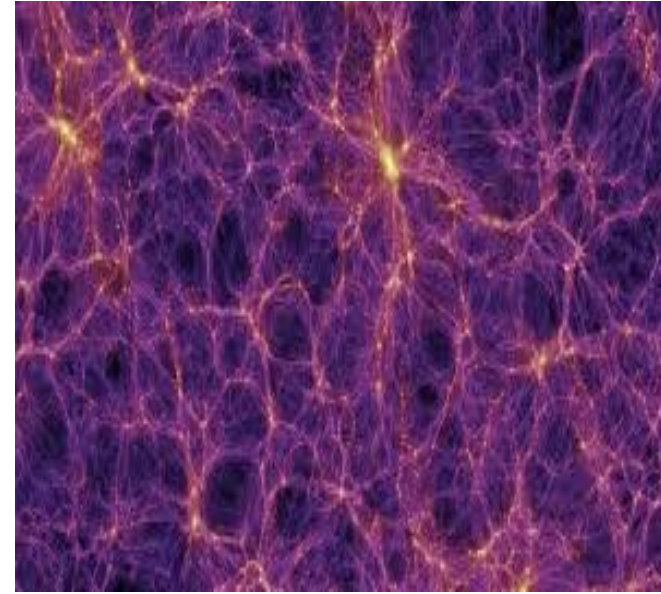
A long-standing goal of artificial intelligence is an algorithm that learns, *tabula rasa*, superhuman proficiency in challenging domains. Recently, AlphaGo became the first program to defeat a world champion in the game of Go. The tree search in AlphaGo evaluated positions and selected moves using deep neural networks. These neural networks were trained by supervised learning from human expert moves, and by reinforcement learning from self-play. Here we introduce an algorithm based solely on reinforcement learning, without human data, guidance or domain knowledge beyond game rules. AlphaGo becomes its own teacher: a neural network is trained to predict AlphaGo's own move selections and also the winner of AlphaGo's games. This neural network improves the strength of the tree search, resulting in higher quality move selection and stronger self-play in the next iteration. Starting *tabula rasa*, our new program AlphaGo Zero achieved superhuman performance, winning 100–0 against the previously published, champion-defeating AlphaGo.

facebook

amazon

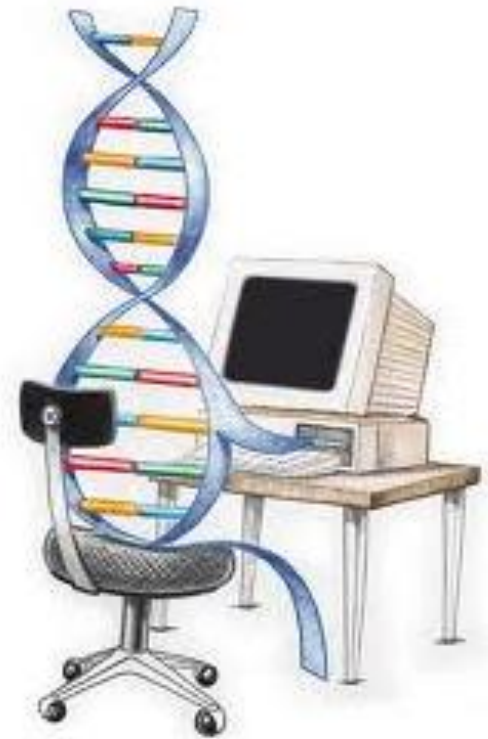
Técnicas

- Redes Neuronales
- Deep learning
 - Backpropagation
 - Convolutional neural network (ConvNet)
 - Model-free and model-based reinforcement learning
 - Deep Q-learning
 - Generative models
 - Program induction

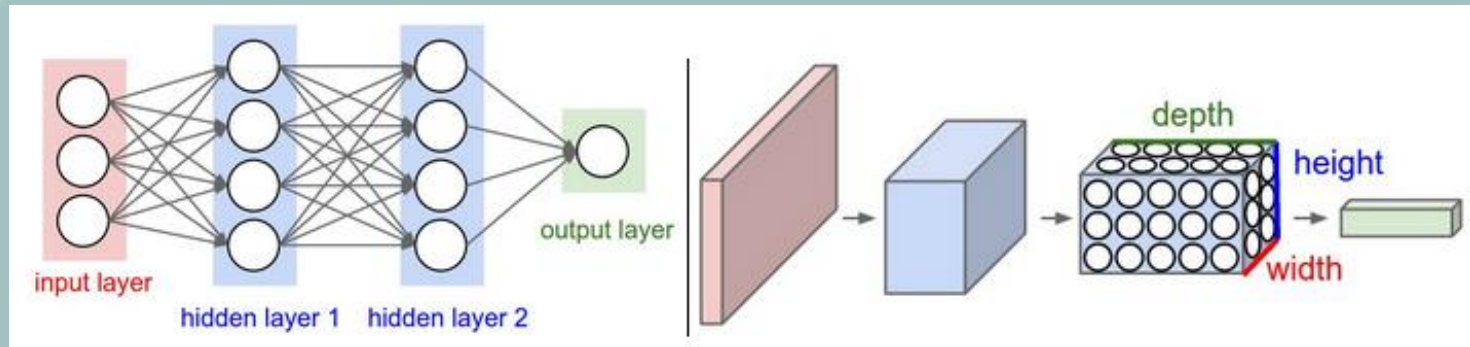


Técnicas (II)

- Evolutionary Computation
- Case-based reasoning

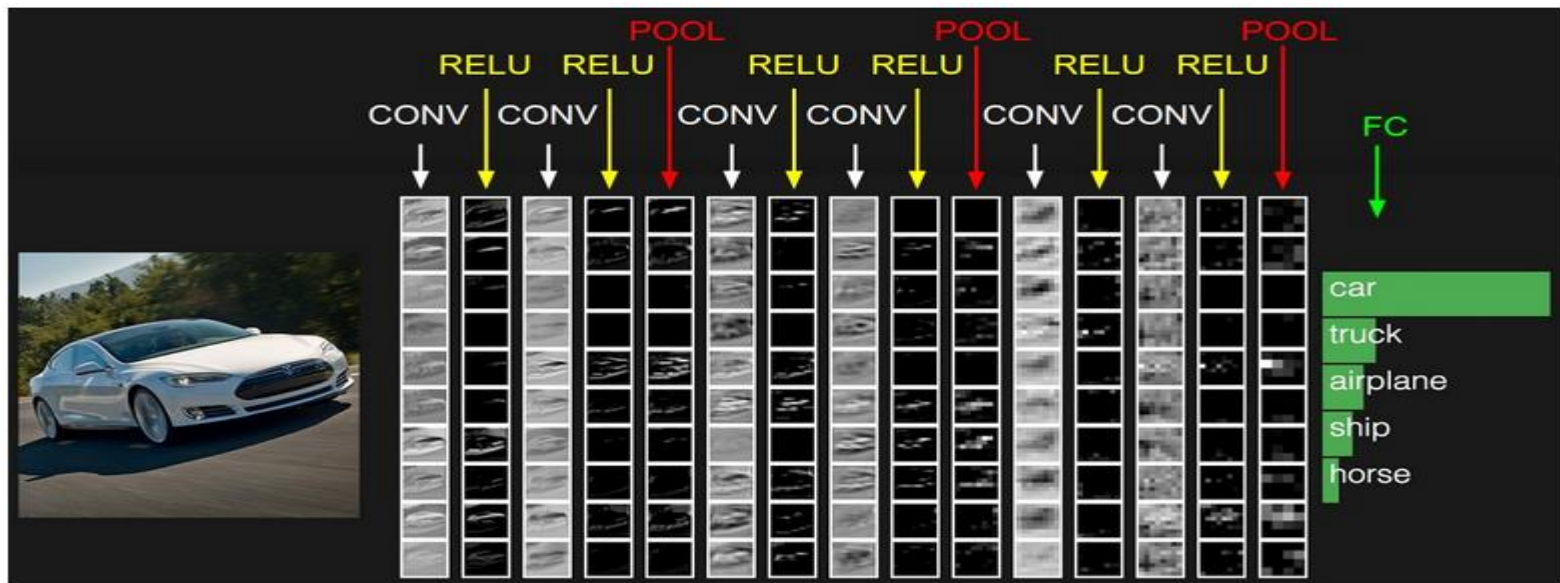


Deep Neural Networks



Red Neuronal Clásica

Red Neuronal Profunda



Deep Neural Networks



1	1	1	0	0
0	1	1	1	0
0	0	1 _{x1}	1 _{x0}	1 _{x1}
0	0	1 _{x0}	1 _{x1}	0
0	1	1 _{x1}	0 _{x0}	0 _{x1}

Image

4	3	4
2	4	3
2	3	4

Convolved
Feature

1	0	1
0	1	0
1	0	1

Filter

IMAGENET Large Scale Visual Recognition Challenge 2014 (ILSVRC2014)

ALEXNET

Open AI: Dota 2 Challenge



Open AI Five: Equipo de 5 redes neuronales (LSTM) desarrolladas por Open AI para enfrentar jugadores humanos (250 GPUs, 128,000 CPU cores para entrenar!!!)

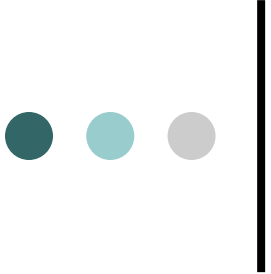
Alpha Star: Mastering the Real-Time Strategy Game StarCraft II





Alpha Star

- **Game theory:** StarCraft is a game where, just like rock-paper-scissors, there is no single best strategy. As such, an AI training process needs to continually explore and expand the frontiers of strategic knowledge.
- **Imperfect information:** Unlike games like chess or Go where players see everything, crucial information is hidden from a StarCraft player and must be actively discovered by “scouting”.
- **Long term planning:** Like many real-world problems cause-and-effect is not instantaneous. Games can also take anywhere up to one hour to complete, meaning actions taken early in the game may not pay off for a long time.
- **Real time:** Unlike traditional board games where players alternate turns between subsequent moves, StarCraft players must perform actions continually as the game clock progresses.
- **Large action space:** Hundreds of different units and buildings must be controlled at once, in real-time, resulting in a combinatorial space of possibilities. On top of this, actions are hierarchical and can be modified and augmented. Our parameterization of the game has an average of approximately 10 to the 26 legal actions at every time-step.

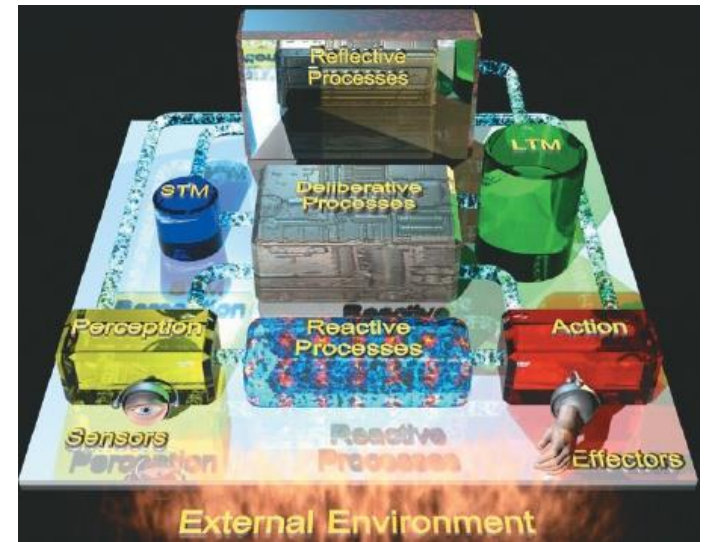


Video juegos como benchmarks (Starcraft, Dota)


- Acciones cuyo efecto se observa o bien a largo plazo, o bien a nivel estratégico.
- Estado observable de manera parcial.
- Altas dimensiones, gran cantidad de acciones y estados, acciones continuas.
- Necesidad de planear a largo plazo.
- <https://blog.openai.com/openai-five/>

Cognitive Systems and Arch's

- “We need to change our perspective on computational systems and get off the current trajectory.” (Brachman, 2002)
- **Systems That Know What They're Doing.**
- “A truly cognitive system would be able to learn from its experience—as well as by being instructed—and perform better on day two than it did on day one. It would be able to explain what it was doing and why it was” (Brachman, 2002)



Cognitive Architectures

- **Cognitive architectures:** Research issues and challenges. Pat Langley, John E. Laird, Seth Rogers. *Cognitive Systems Research*, 2009, Elsevier.
- ACT-R, SOAR, Prodigy, Icarus.  Numenta

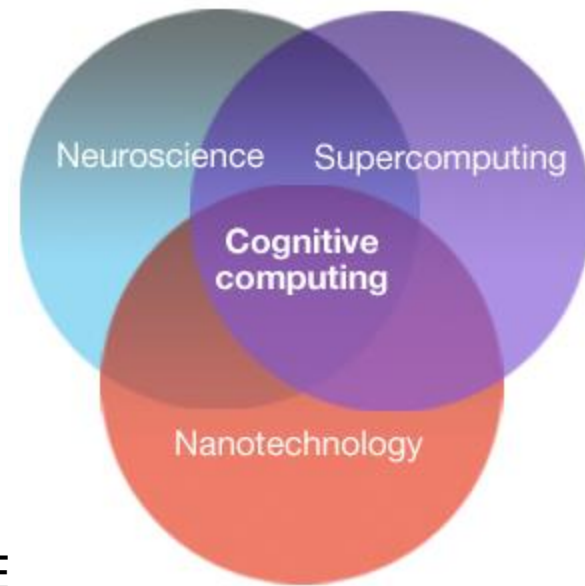
A cognitive architecture specifies the underlying infrastructure for an intelligent system. Briefly, an architecture includes those aspects of a cognitive agent that are constant over time and across different application domains. These typically include:

- the short-term and long-term memories that store content about the agent's beliefs, goals, and knowledge;
- the representation of elements that are contained in these memories and their organization into larger-scale mental structures; and
- the functional processes that operate on these structures, including the performance mechanisms that utilize them and the learning mechanisms that alter them.





Cognitive Computing



- IBM SyNAPSE
- Systems of Neuromorphic Adaptive Plastic Scalable E
- “By reproducing the structure and architecture of the brain—the way its elements receive sensory input, connect to each other, adapt these connections, and transmit motor output—the SyNAPSE project models computing systems that emulate the brain's computing efficiency, size and power usage without being programmed.”
- http://www.ibm.com/smarterplanet/us/en/business_analytics/article/cognitive_computing3.html

Tipos de Aprendizaje

