INF8225

Intelligence artificielle : techniques probabilistes et d'apprentissage

Christopher Pal

École Polytechnique de Montréal

Introductions

Sondage / Discussion:

- Étudiant BAC vs maîtrise vs doctoral (superviseur et projet)
- Les cours similaires à INF8225 que vous avez déjà suivi

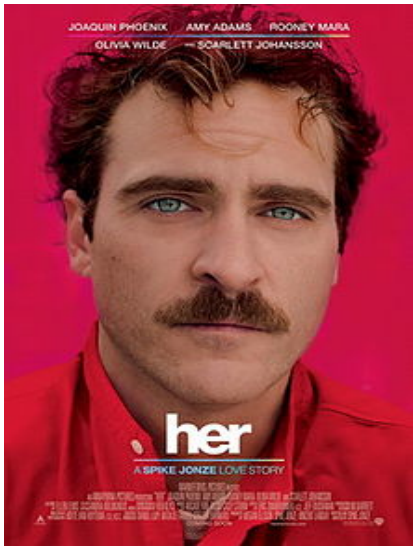
Quelques questions pour vous

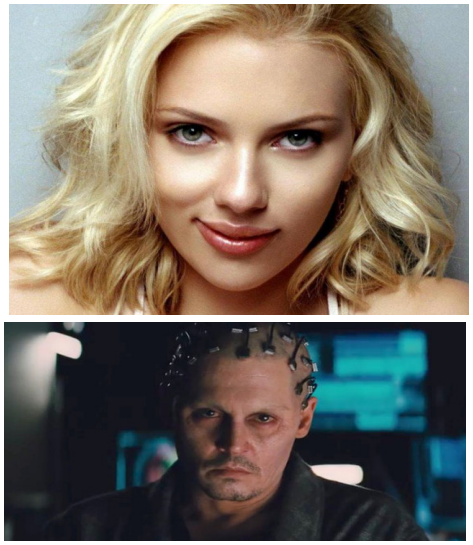
1) Quel est ce concept:
   l'intelligence artificielle?

2) Parmi les sujets et les applications d'IA,
   donnez-moi les sujets le plus intéressants
   pour vous.

There has been a lot of talk of AI & 'Deep Learning' in the major US media

- *Nov. 2012 – NYTimes*
  Scientists See Promise in Deep Learning Programs
- *Nov 2012 – The New Yorker*
  Is Deep Learning a Revolution in Artificial Intelligence?





- *2015 – NYTimes Article on the Open Letter signed by* Elon Musk, Stephen Hawking, Hinton, Bengio, LeCun, etc.

Commentaires

- En comparaison avec INF8215 (Intelligence artif.: méthodes et algorithmes), INF8225 se concentre sur les approches probabiliste du « deep learning » perspective du manuel « Data Mining » Chapitre 9 et 10 et sur
- **Des articles de recherche**.
- INF8225 se concentre sur les approches plus modernes en utilisant **Python, et/ou PyTorch et/ou Theano, et TensorFlow (C/ C++)** pour les TPs (pas Prolog)

Commentaires

- INF8225 doit être admissible comme un cours à option dans notre concentration multimédia de GL.
- Le contenu d'INF8225 est comme un mélange avec le cours d'intro IA célèbre en ligne à Stanford avec plus de 100,000 étudiants, un cours sur l'apprentissage automatique et un cours sur « deep learning »

En pratique

- Nous allons aussi lire quelques articles de recherche dans ce cours – beaucoup après la semaine de relâche
- Vous serez chargé de présenter un article de recherche et votre projet dans un groupe de pas plus de 4 personnes dans une présentation de 15 - 20 min
- Laboratoire pour les travaux pratiques Mardi 15h45-18:30, L-4712
- Chargés de labo B1 et 2: Alexandre Piché

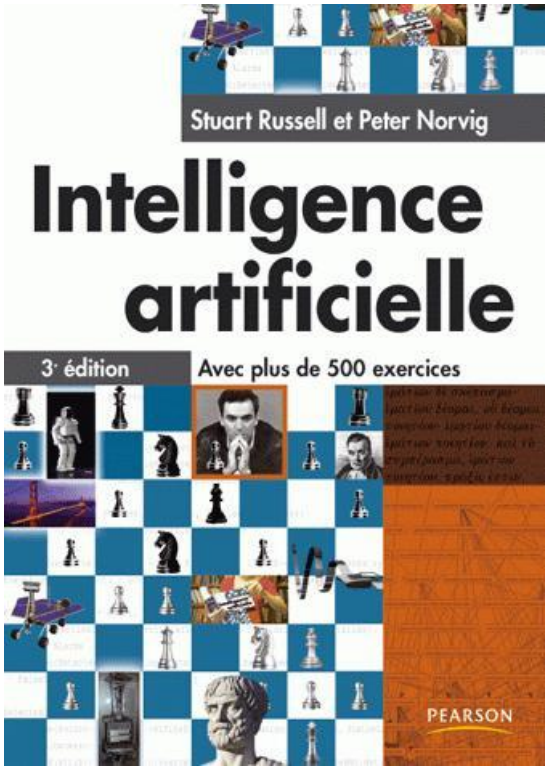
Évaluations

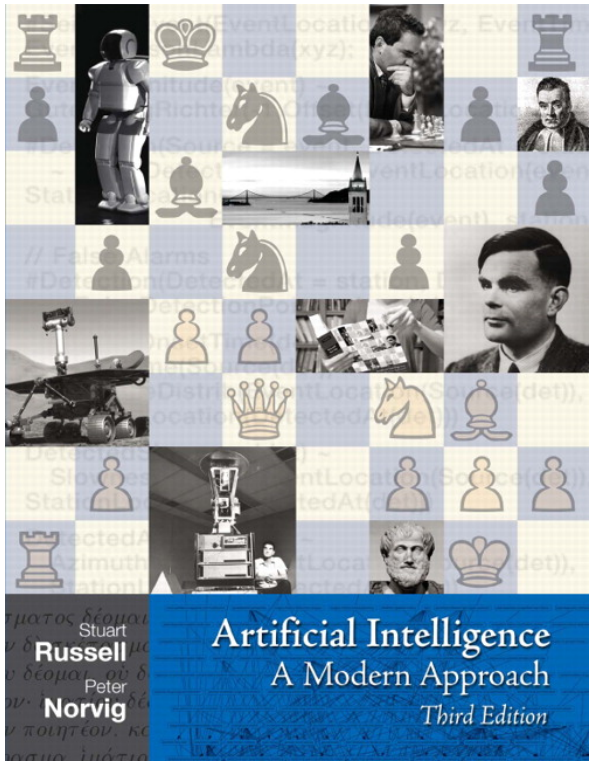
Composante	**%**
TP1 (individuel)	10%
TP2 (individuel)	10%
TP3 (individuel)	10%
Projet (en groupe, max 4), incluant la présentation d'un article de recherche	25%
Contrôle périodique	20%
Examen final	25%

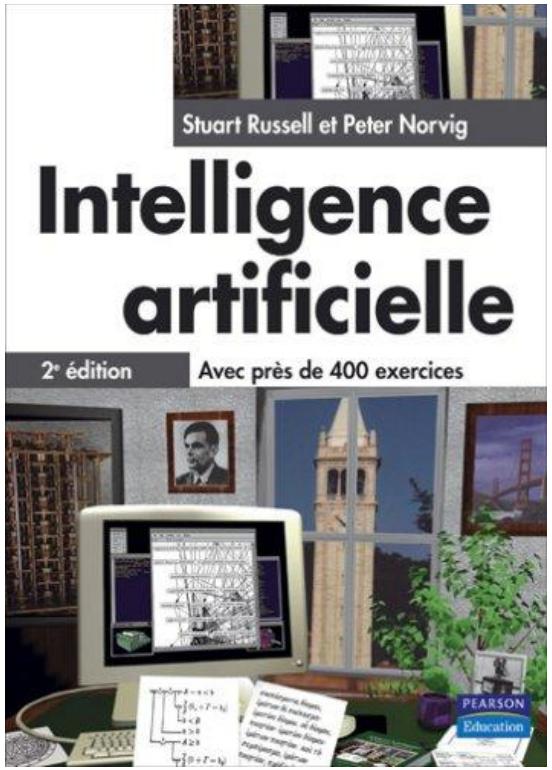
Voir le plan de cours !

Les manuels et des référencés

- Le vieux manuel primaire



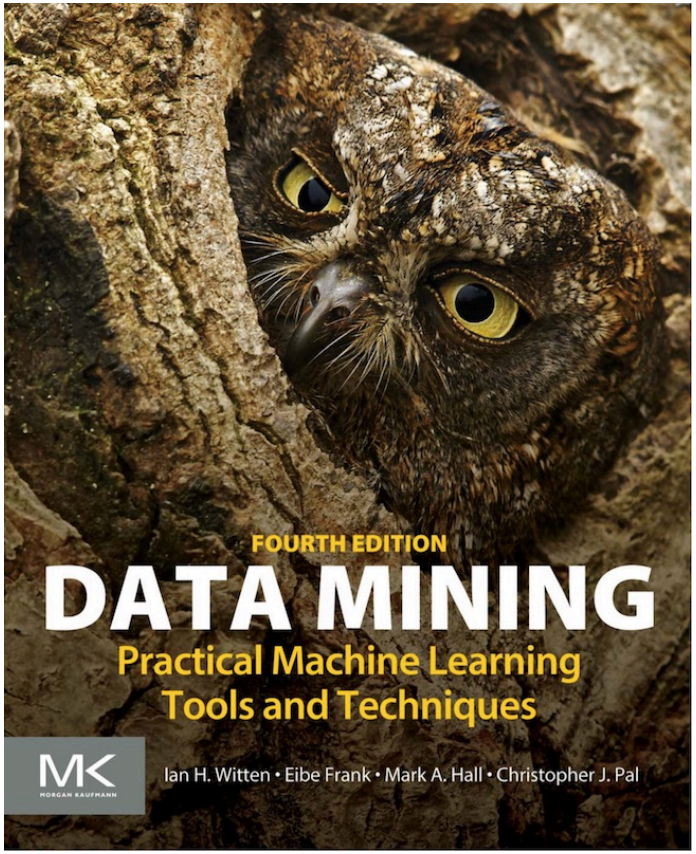




Intelligence artificielle 3e éd. Russel et Norvig, Pearson France, (disp. @ poly.)

Data Mining - Practical Machine Learning Tools and Techniques

- Witten, Frank, Hall and Pal, 4th Edition (2016)
- From the University of Waikato in NZ - where I was on sabbatical in 2015
- Will mainly use appendices & chapters 9,10
- Slides available online:

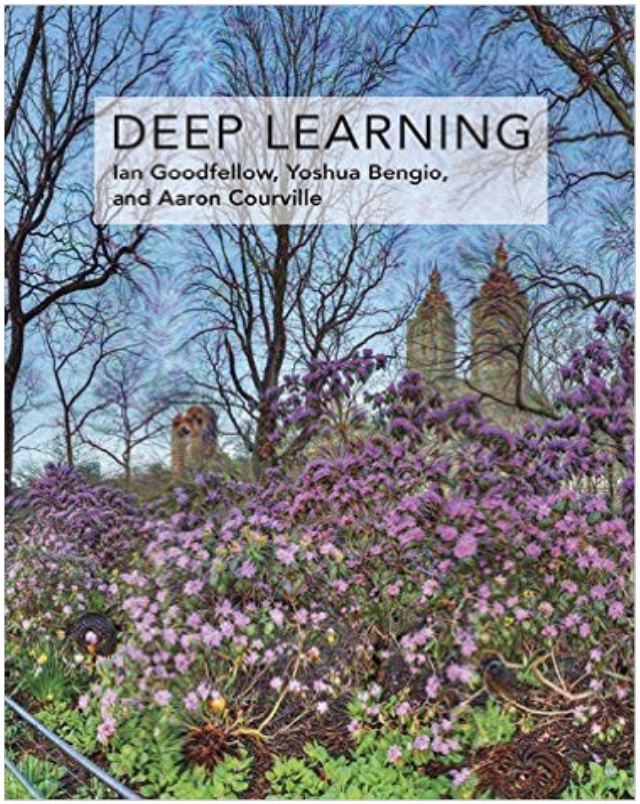


http://www.cs.waikato.ac.nz/ml/weka/book.html

Bien sûr

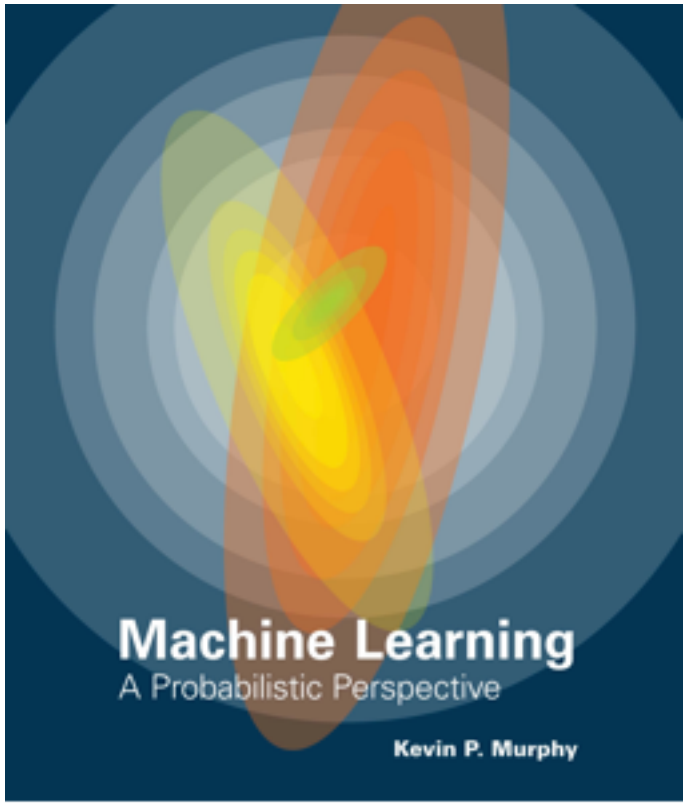
- Il y a un manuel récent focalisant sur Deep Learning
- Par: Ian Goodfellow Yoshua Bengio, et Aaron Courville
- Il est disponible en ligne:

http://www.deeplearningbook.org/



Autre manuel recommandé

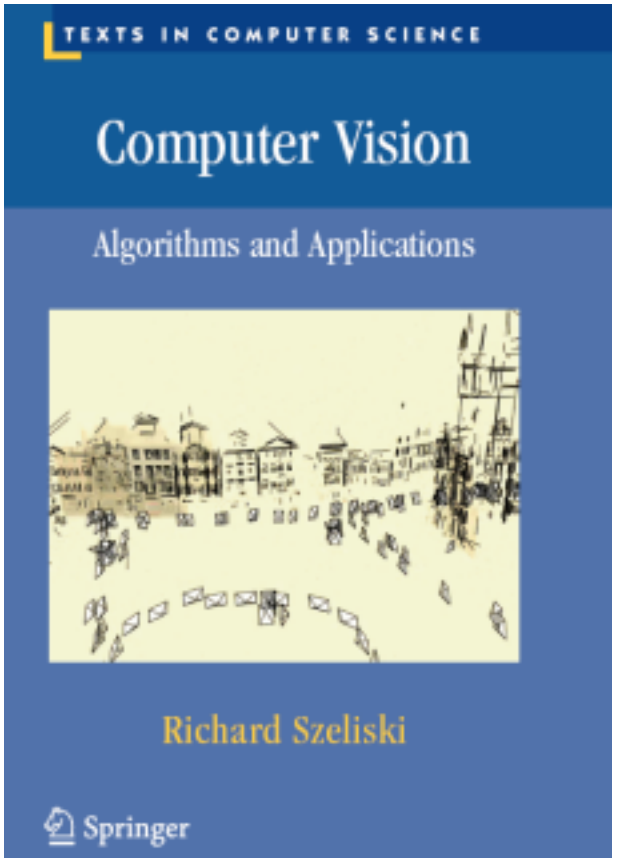
- Kevin Murphy
  Machine Learning: a Probabilistic Perspective
- Notez: Chapitre 1 disponible en ligne
- On va utiliser PMTK un peu -- pour TP1



http://www.cs.ubc.ca/~murphyk/MLbook/index.html

D’autres manuels utiles

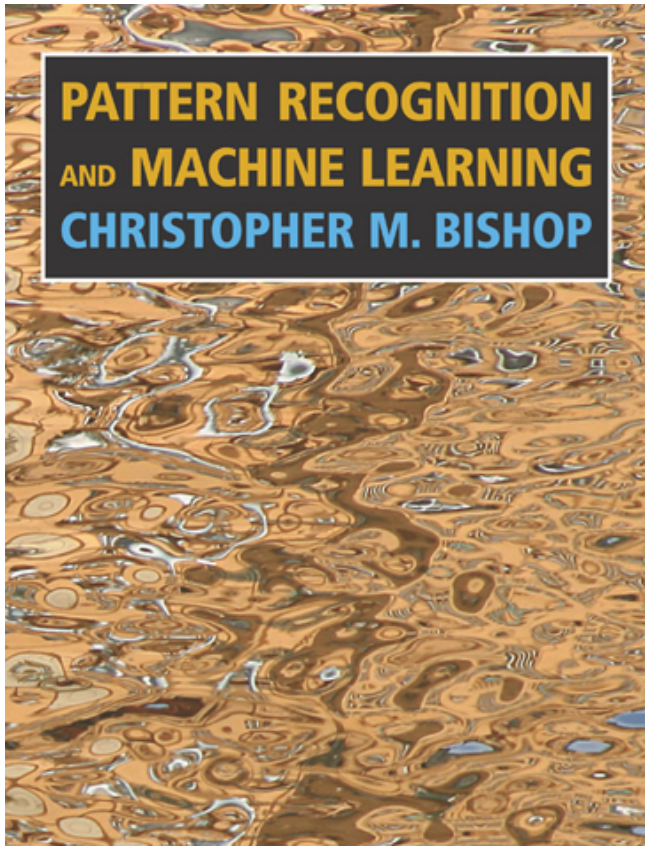
- Richard Szeliski Computer Vision Algorithms and Applications
- Notez: Des versions préliminaires disponibles en ligne



http://szeliski.org/Book/

D’autres manuels utiles

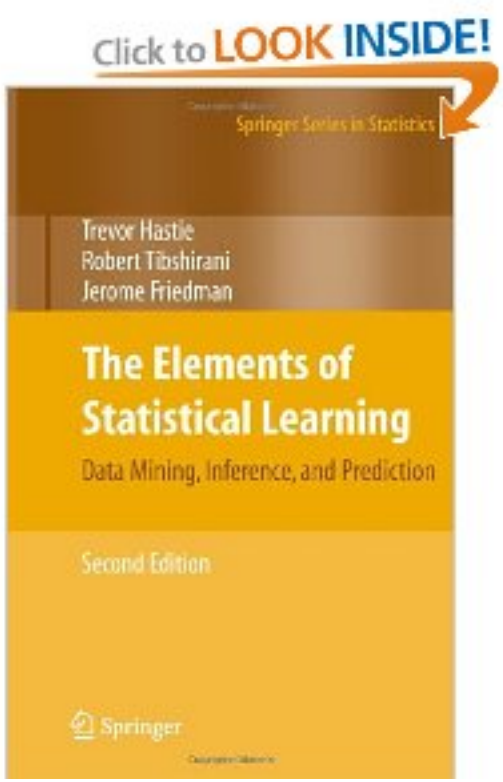
- Chris Bishop
  Pattern Recognition
  and Machine Learning



http://research.microsoft.com/en-us/um/people/cmbishop/prml/

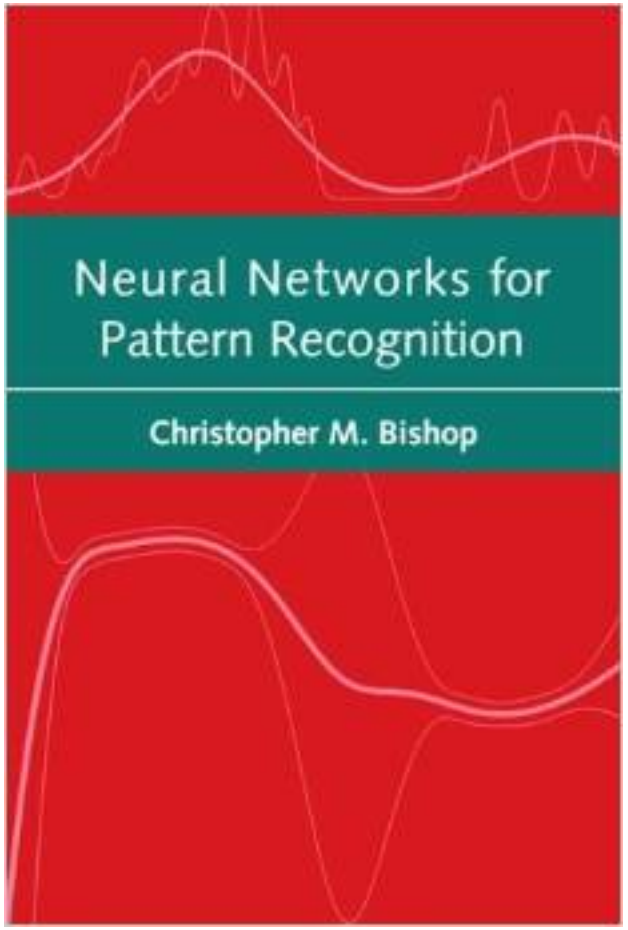
D’autres manuels utiles

- The Elements of Statistical Learning: Data Mining, Inference, and Prediction
- Trevor Hastie, Robert Tibshirani, & Jerome Friedman



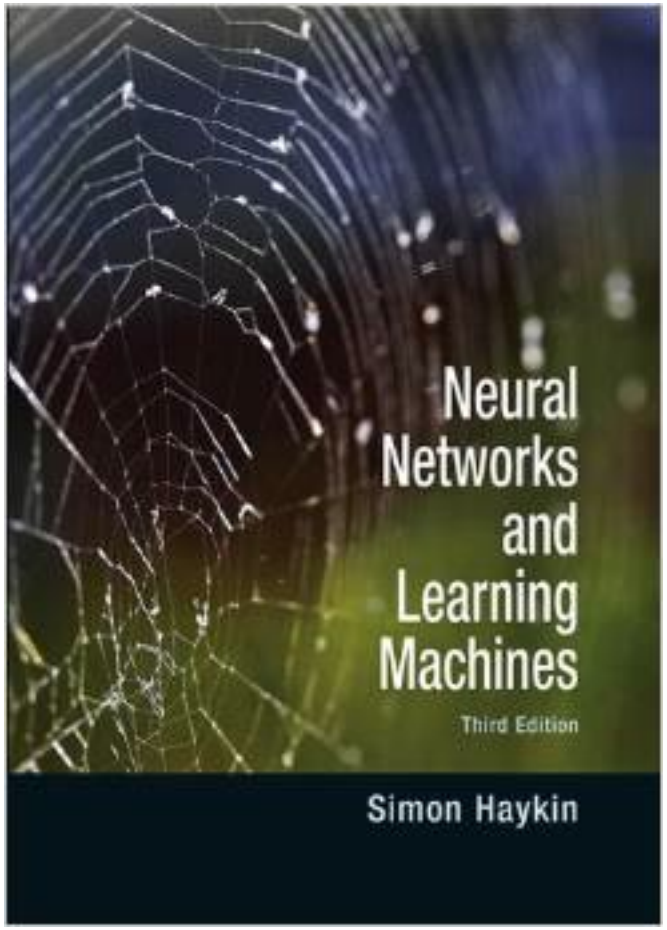
Older Textbook from Chris Bishop

- Neural Networks for Pattern Recognition (1996)
- Chris Bishop



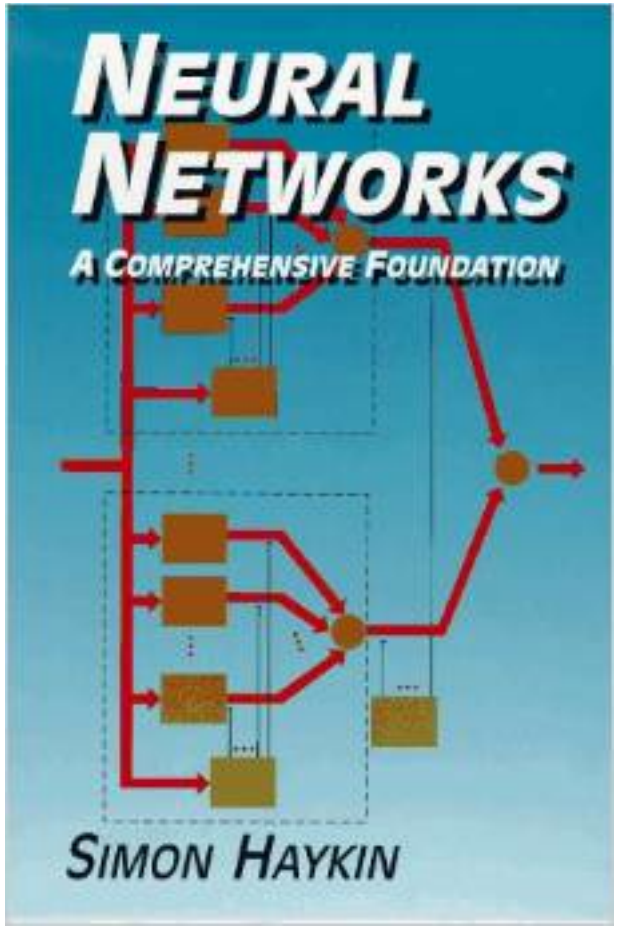
Classical Neural Networks Textbook

- Neural Networks and Learning Machines (2008)
- Simon Haykin McMaster University



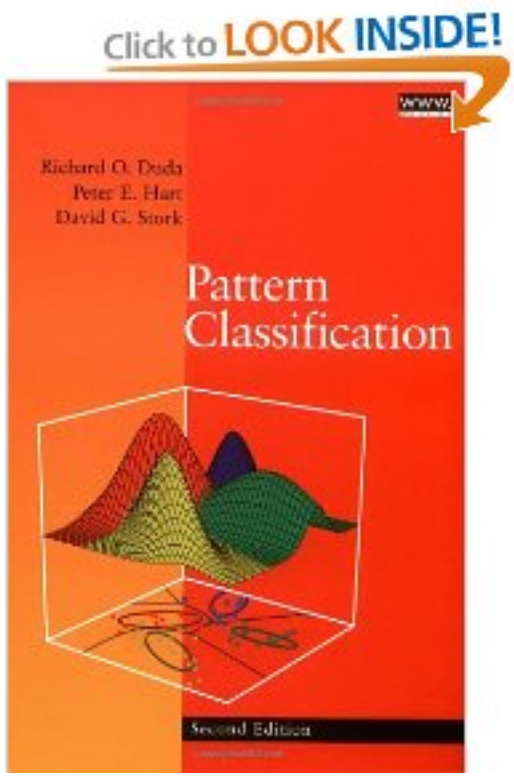
Older Edition (widely used)

- Neural Networks a Comprehensive Foundation (1994)
- Simon Haykin McMaster University



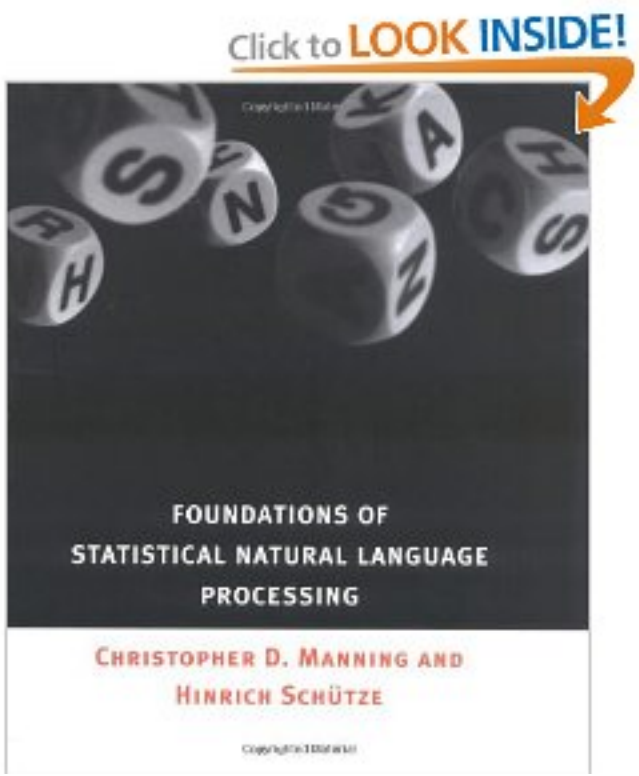
Manuel ‘Classique’

- Pattern Classification
- Duda, Hart & Stork



D’autres manuels utiles - NLP

- Foundations of Statistical Natural Language Processing
- Christopher Manning and Hinrich Schütze



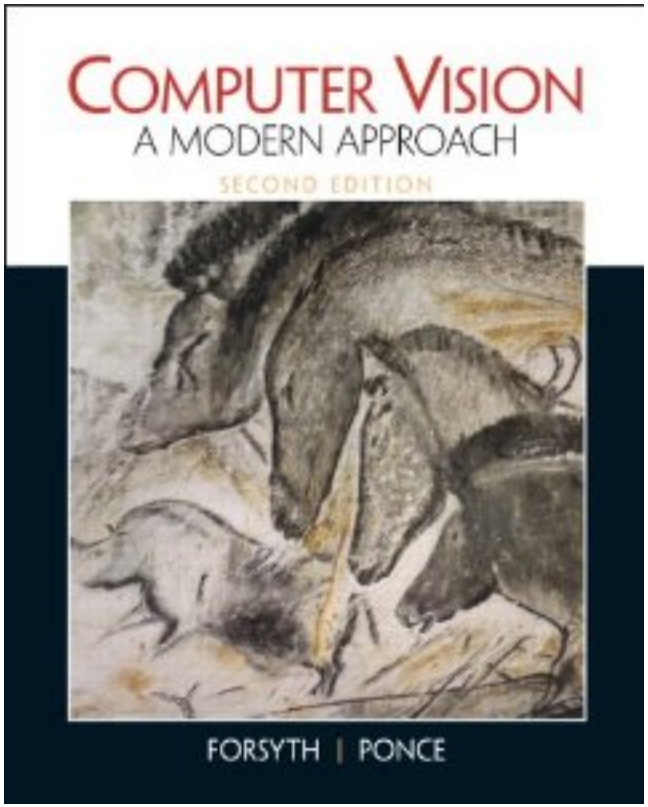
D’autres manuels utiles

- Simon J.D. Prince. Computer Vision: Models, Learning, and Inference.
- Notez: disponible en ligne (.pdf) le manuel au complet

http://www.computervisionmodels.com/

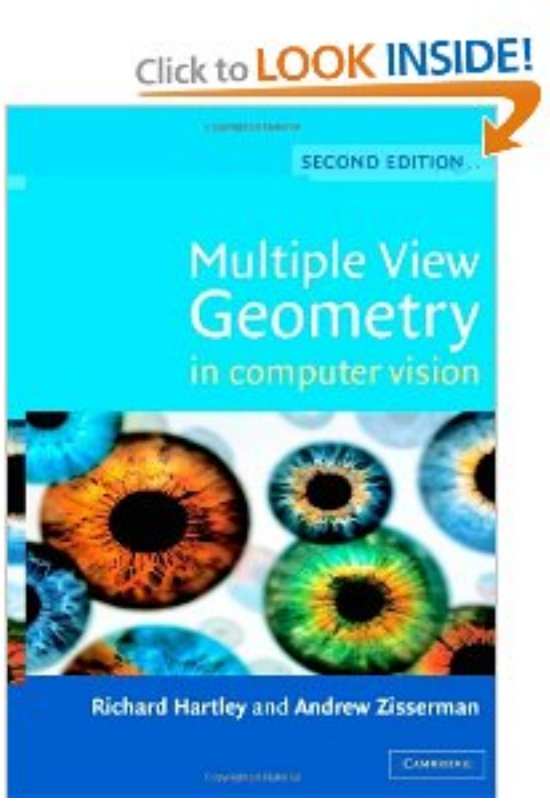
D’autres manuels utiles – Vision 3D

- Computer Vision a Modern Approach
- Forsyth & Ponce



D’autres manuels utiles – Vision 3D

- Multiple View Geometry
- Hartley & Zisserman



Un point de comparaison – Intro. IA – partie 1 (IA et logique)

- Prolog (programmation avec logique)
- Agents Intelligents
- Résolution de problèmes par l'exploration
- Exploration non informée vs. Informée (A*)
- Problèmes à satisfaction de contraintes
- Agents logiques
- Logique du premier ordre
- L'inférence en logique du premier ordre
- Représentation des connaissances
- Planification

IA dans les 1990s

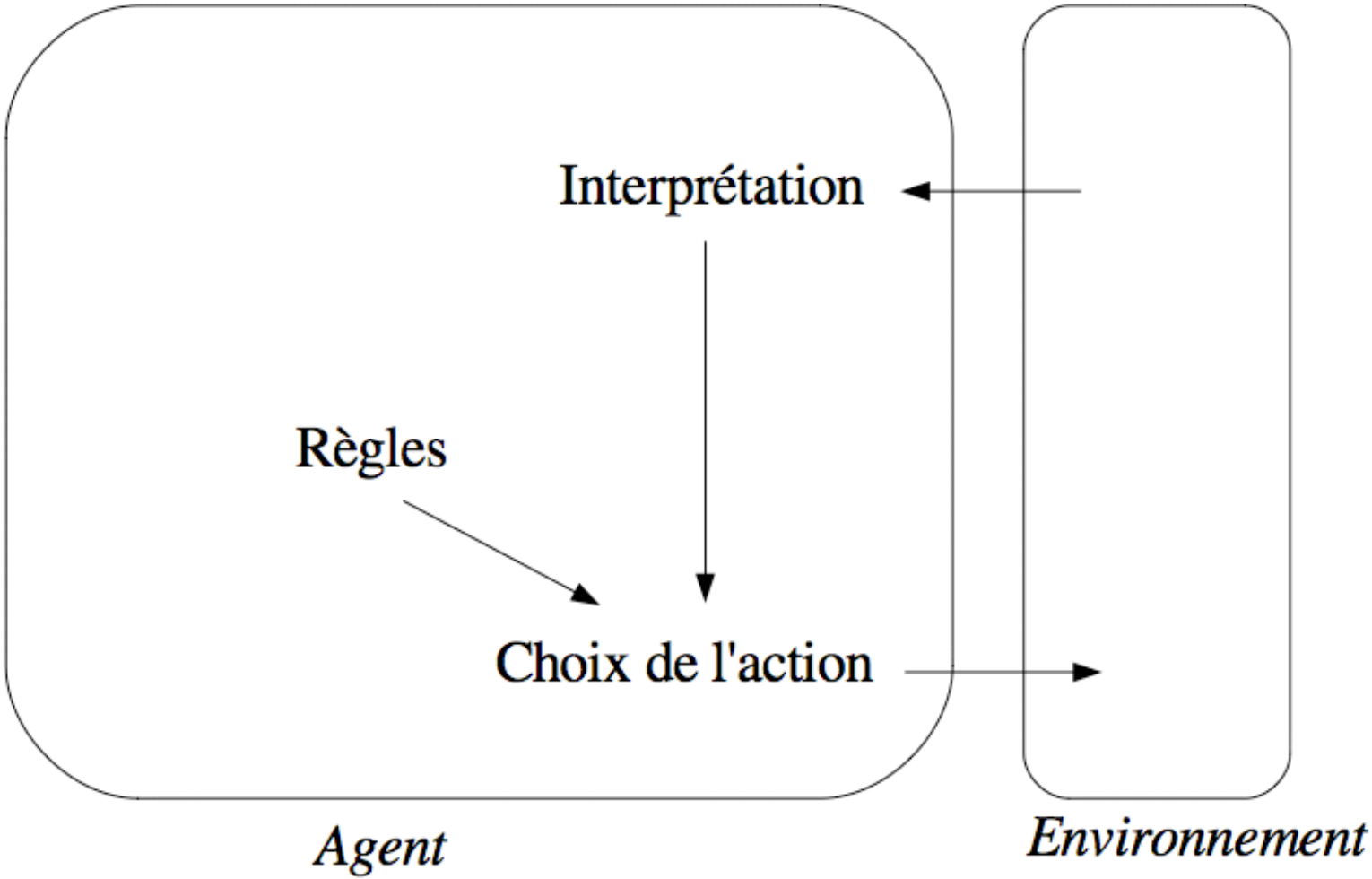
Kasparov vs Deep Blue:
a contrast in styles

Dissimilar processes yield similar conclusions

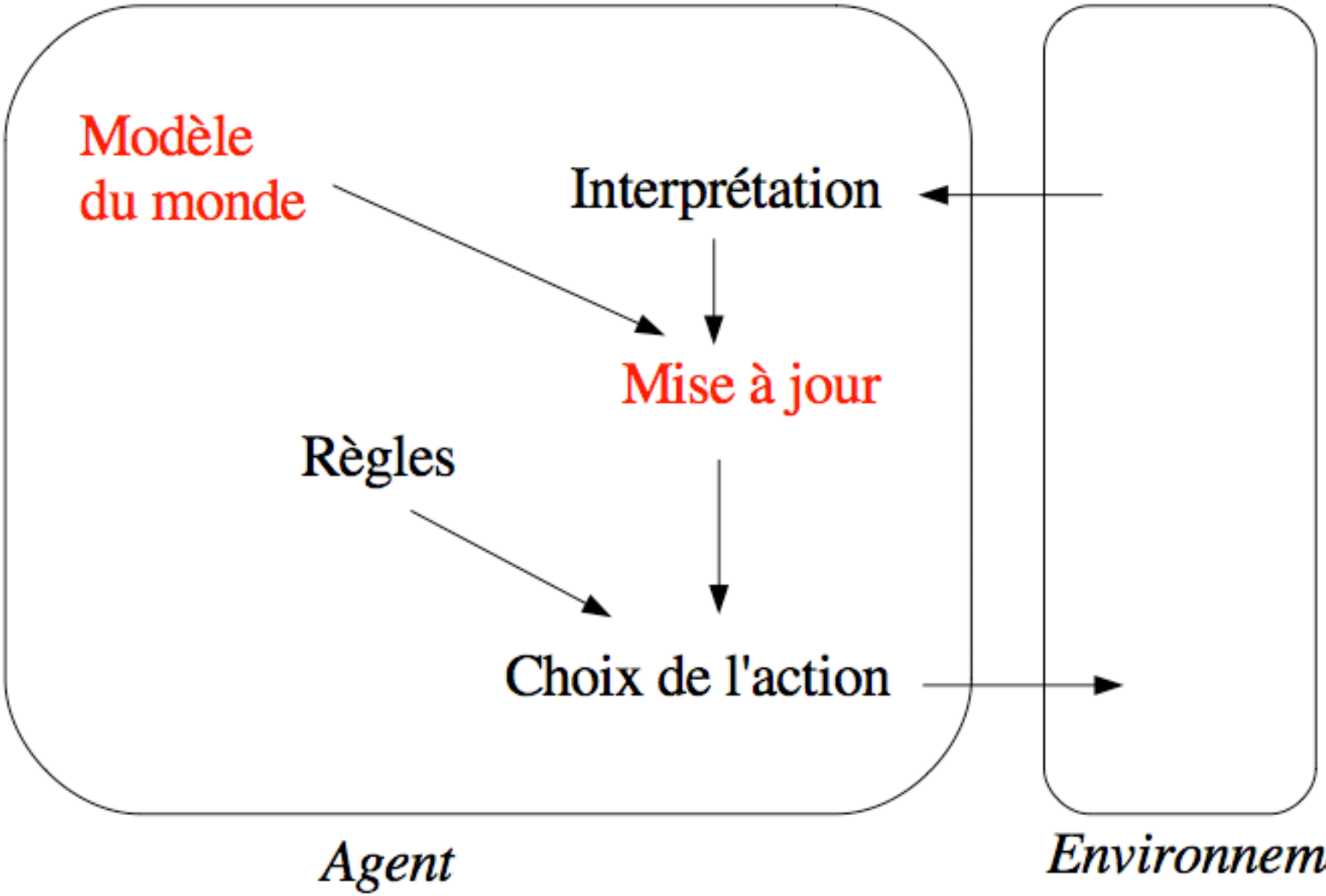
IBM®

In May 1997, IBM's Deep Blue Supercomputer played a fascinating match with the reigning World Chess Champion, Garry Kasparov. The event was captured live only on this Web site, where millions of chess and computing fans tuned in to witness the event in real-time. This Web site is an archive of that event, and information on this site has not been updated since the end of the match. Some content may no longer be relevant or up to date, and some links may not function. In particular, the audio and video clips are no longer available. Current information about IBM deep computing can be found at the IBM Research home page.

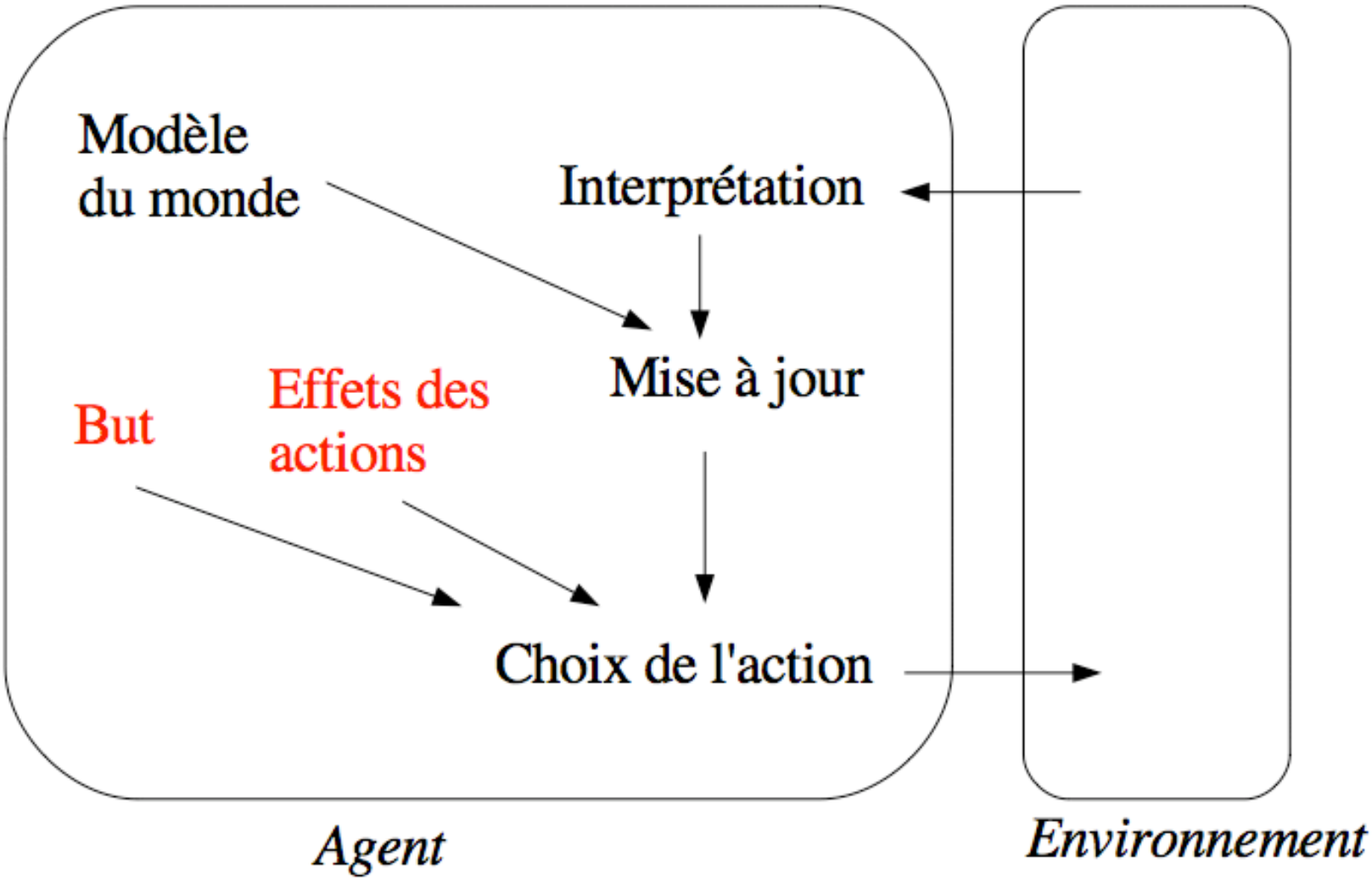
Agent réflexe



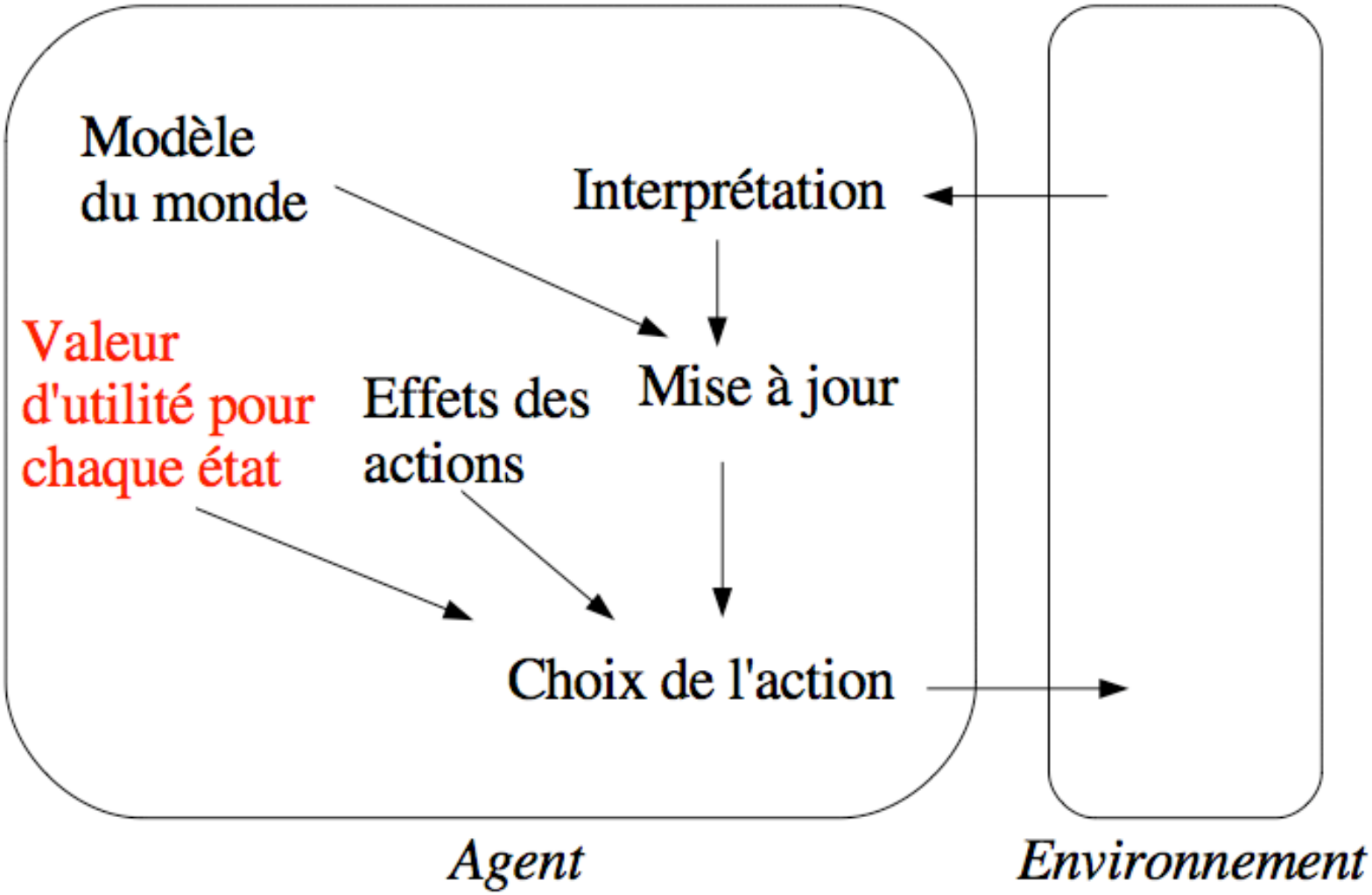
Agent intelligent avec mémoire



Agent intelligent avec buts



Agent intelligent avec théorie des décisions



L ’architectures des agents

- Agents réflexes simples
- Agents réflexes fondés sur des modèles
- Agents fondés sur des buts
- Agents fondés sur l’utilité
- **Agents capable d’apprentissage**

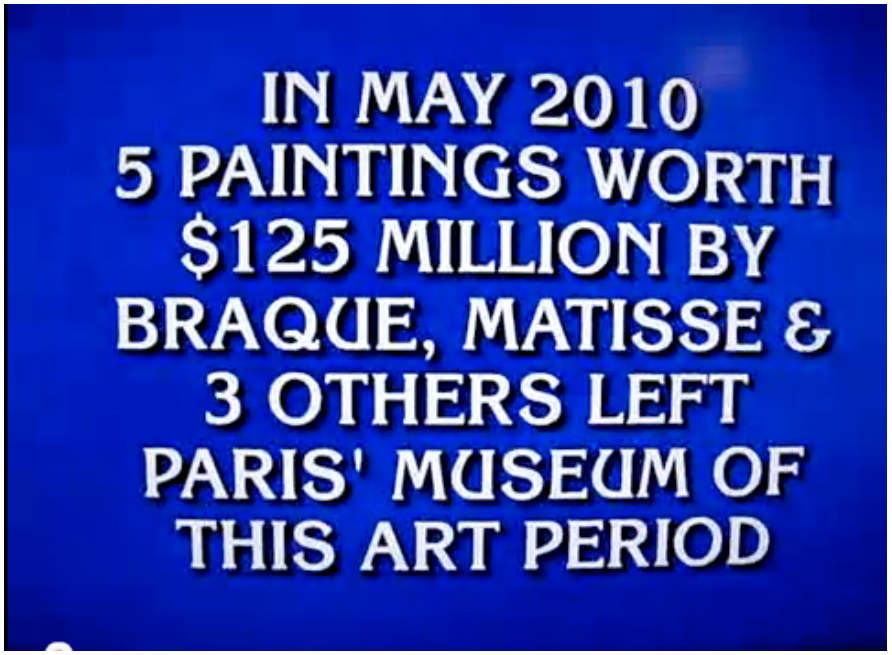
Ce cours

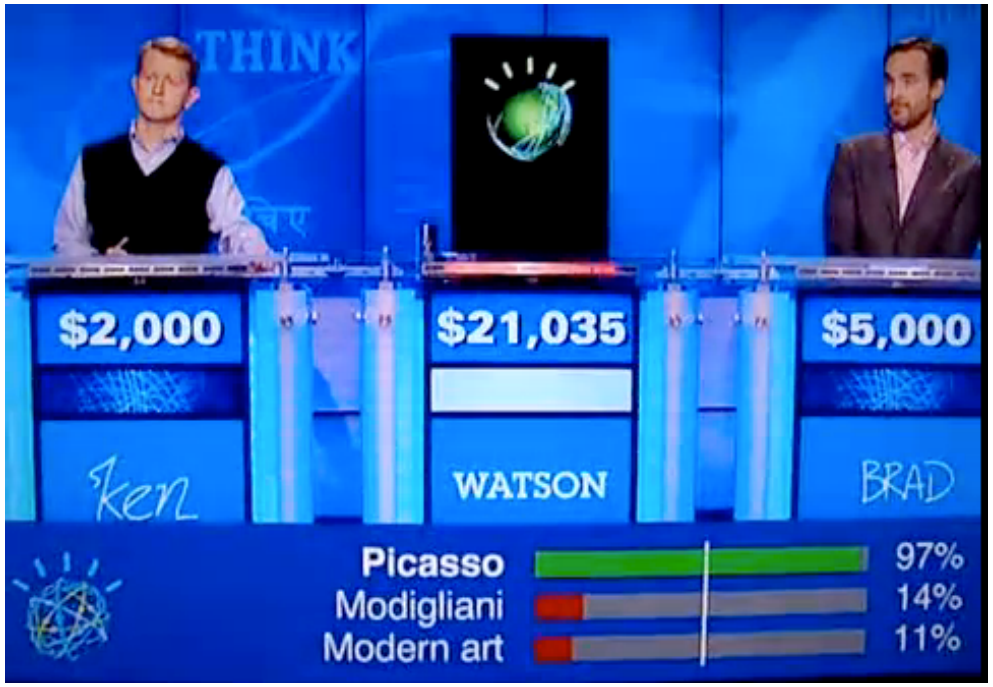
Intro. IA – partie 2 (IA et probabilité)

« Programmation avec la probabilité et des informations statistiques »

- Incertitude et probabilité de base
- Raisonnement probabiliste
- Raisonnement probabiliste temporel
- Prise de décisions simples
- Prise de décisions complexes
- Apprendre à partir d'observations
- Connaissances et apprentissage
- Méthodes d'apprentissage statistique

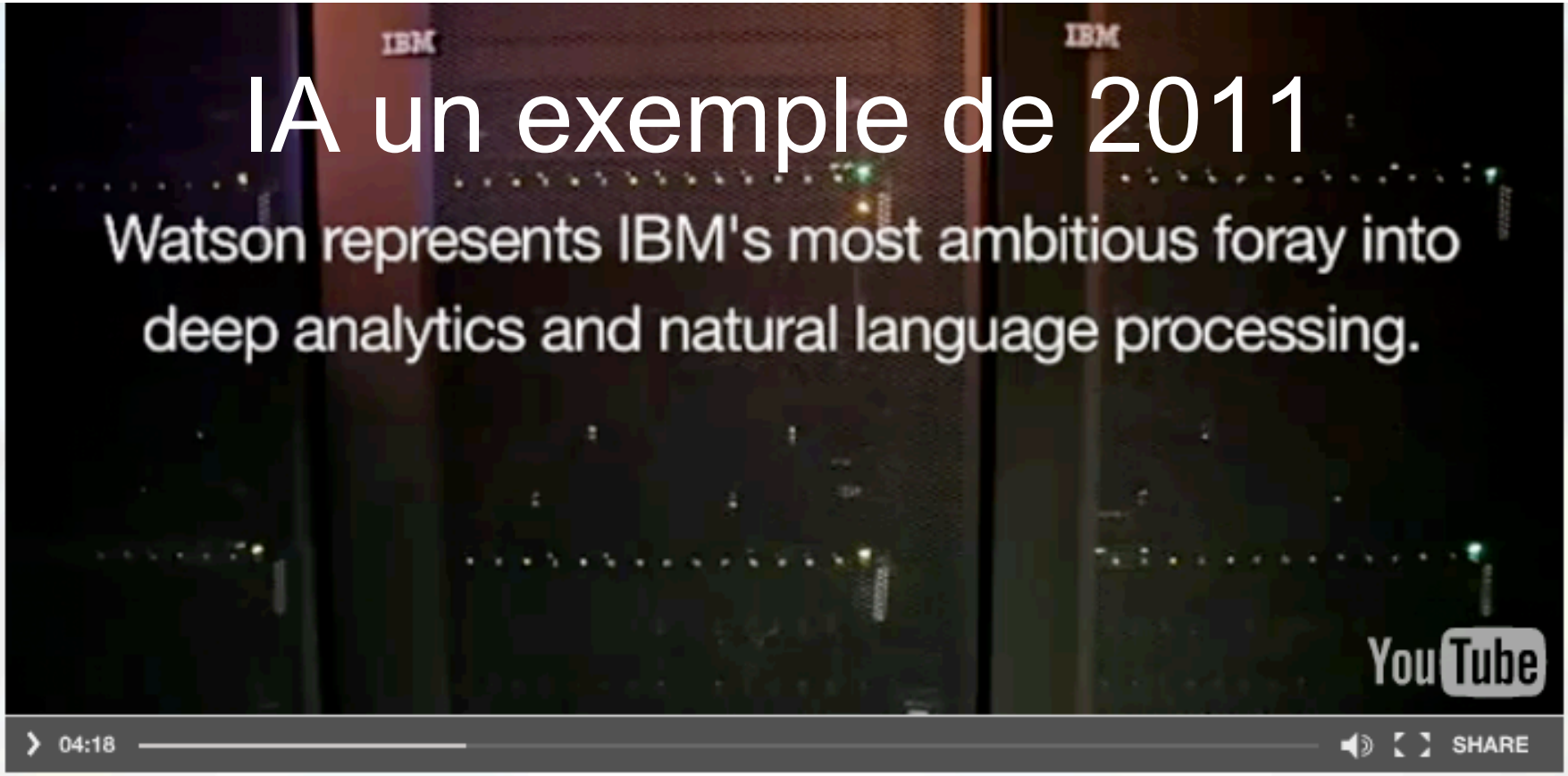
Intelligence Artificielle 2011





- À noter : réponses avec confidence et réponse correct

IA un exemple de 2011



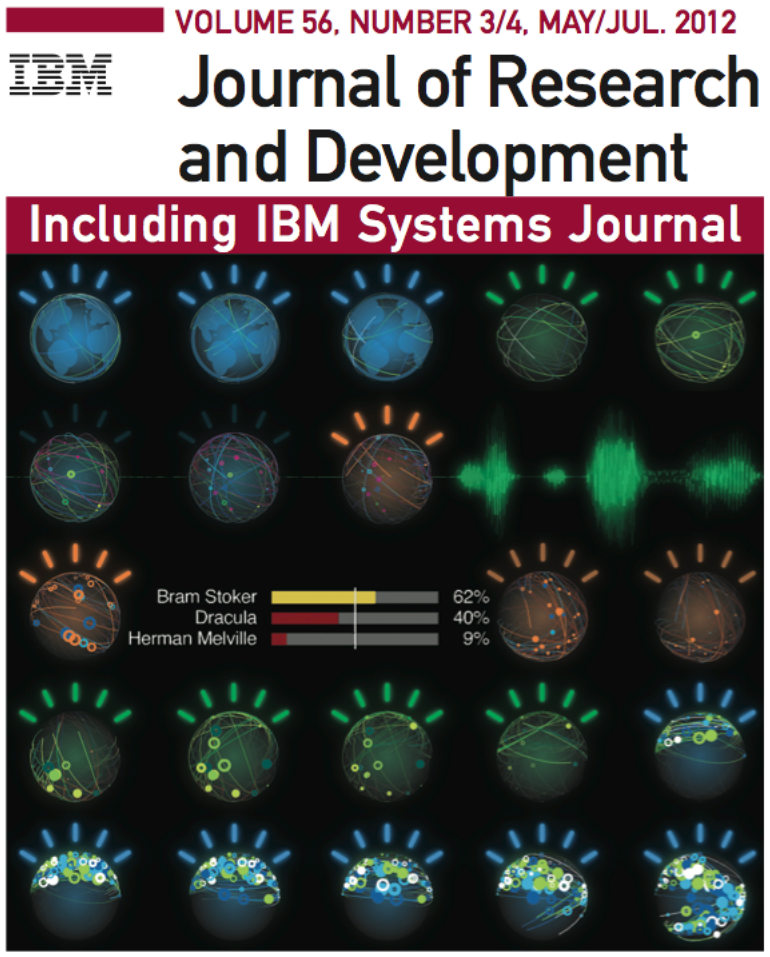
Why Jeopardy!?

The game of *Jeopardy!* makes great demands on its players – from the range of topical knowledge covered to the nuances in language employed in the clues. Can the analytical power of a computer system – normally accustomed to executing precise requests – overcome these obstacles? Can the troves of knowledge written in human terms become easily searchable by a machine in order to deliver a single, precise answer? Can a quiz show help advance science?

"IBM is not in the entertainment business. But we are in the business of technology and pushing frontiers."

David Shepler
IBM Research Program Manager

How Does Watson Work?

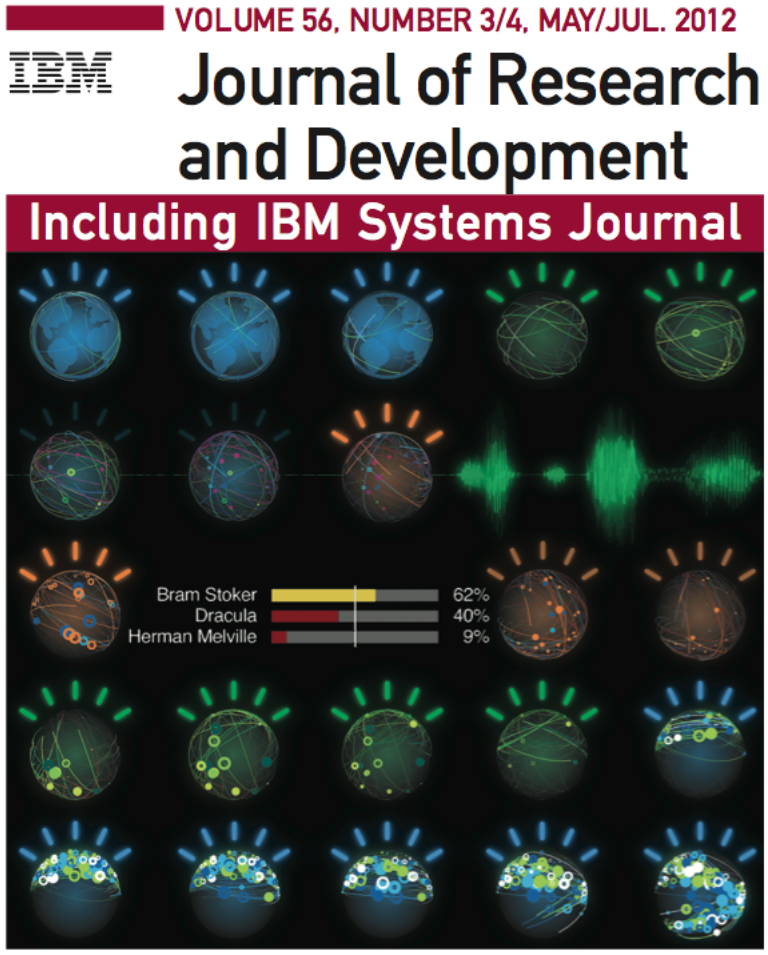


This Is Watson

Let's examine the Table of Contents

1. **Introduction to "This is Watson"** Ferrucci, D.A.
2. **Question analysis: How Watson reads a clue** Lally, A. et al.
3. **Deep parsing in Watson** McCord, M.C. et al.
4. **Textual resource acquisition and engineering** Chu-Carroll, J. et al.
5. **Automatic knowledge extraction from documents** Fan, J. et al.
6. **Finding needles in the haystack: Search and candidate generation** Chu-Carroll, J. et al.
7. **Typing candidate answers using type coercion** Murdock, J.W. et al.
8. **Textual evidence gathering and analysis** Murdock, J.W. et al.

How Does Watson Work?



This Is Watson

Let's examine the Table of Contents

9. **Relation extraction and scoring in DeepQA**
   Wang, C. ; et al.
10. **Structured data and inference in DeepQA**
    Kalyanpur, A. ; et al.
11. **Special Questions and techniques**
    Prager, J.M. et al.
12. **Identifying implicit relationships**
    Chu-Carroll, J. et al.
13. **Fact-based question decomposition in DeepQA** Kalyanpur, A. et al.
14. **A framework for merging and ranking of answers in DeepQA** Gondek, D.C. et al.
15. **Making Watson fast** Epstein, E.A. et al.
16. **Simulation, learning, and optimization techniques in Watson's game strategies**
    Tesauro, G. et al.
17. **In the game: The interface between Watson and Jeopardy!** Lewis, B.L.

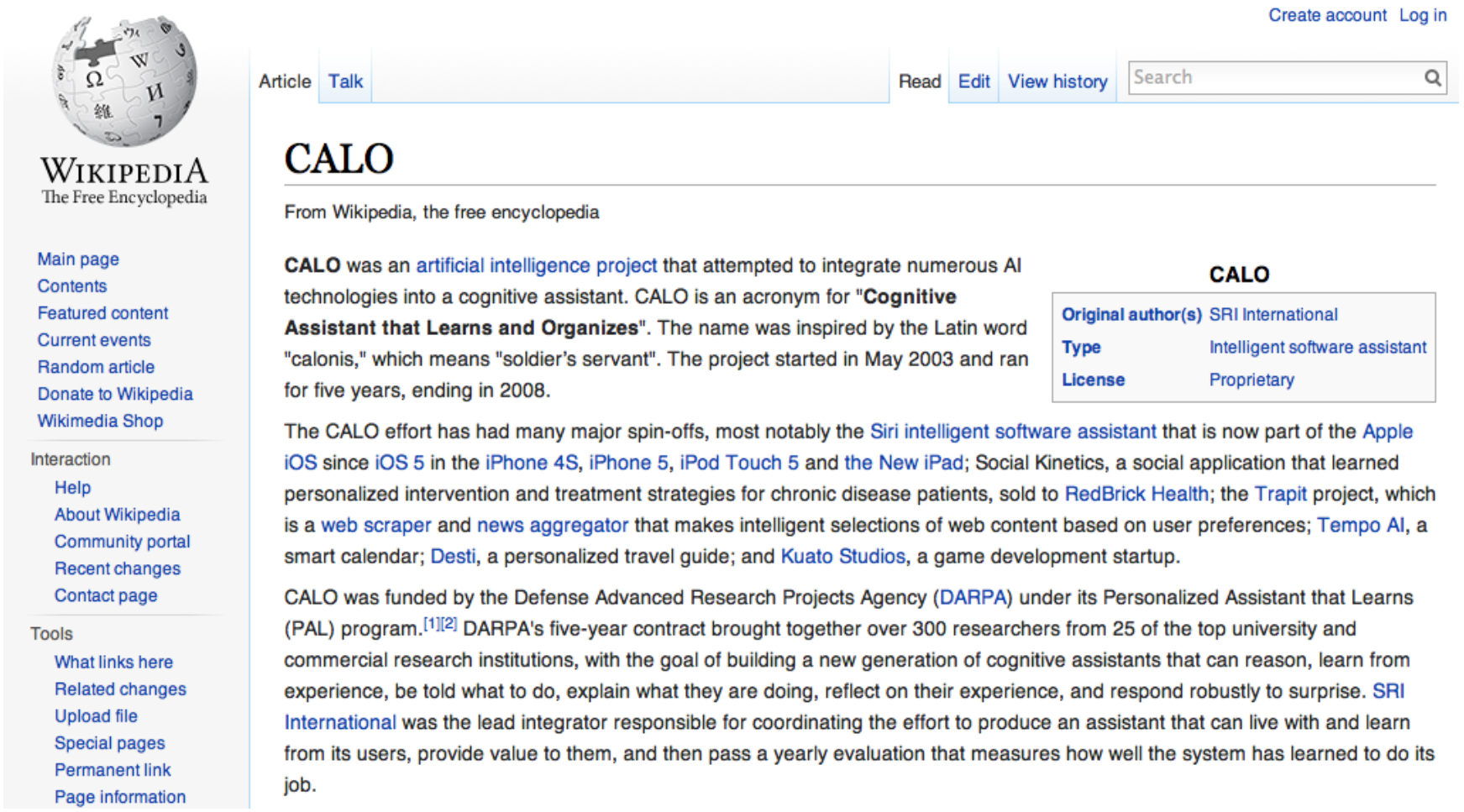
A practical application

Example from Apple

Exemple: Apple Siri

What is this history of Siri?

- Origin: CALO – “A cognitive assistant that learns and organizes”



Create account Log in

Article Talk Read Edit View history Search

WIKIPEDIA
The Free Encyclopedia

Main page
Contents
Featured content
Current events
Random article
Donate to Wikipedia
Wikimedia Shop

Interaction
Help
About Wikipedia
Community portal
Recent changes
Contact page

Tools
What links here
Related changes
Upload file
Special pages
Permanent link
Page information

CALO

From Wikipedia, the free encyclopedia

CALO was an artificial intelligence project that attempted to integrate numerous AI technologies into a cognitive assistant. CALO is an acronym for "**Cognitive Assistant that Learns and Organizes**". The name was inspired by the Latin word "calonis," which means "soldier's servant". The project started in May 2003 and ran for five years, ending in 2008.

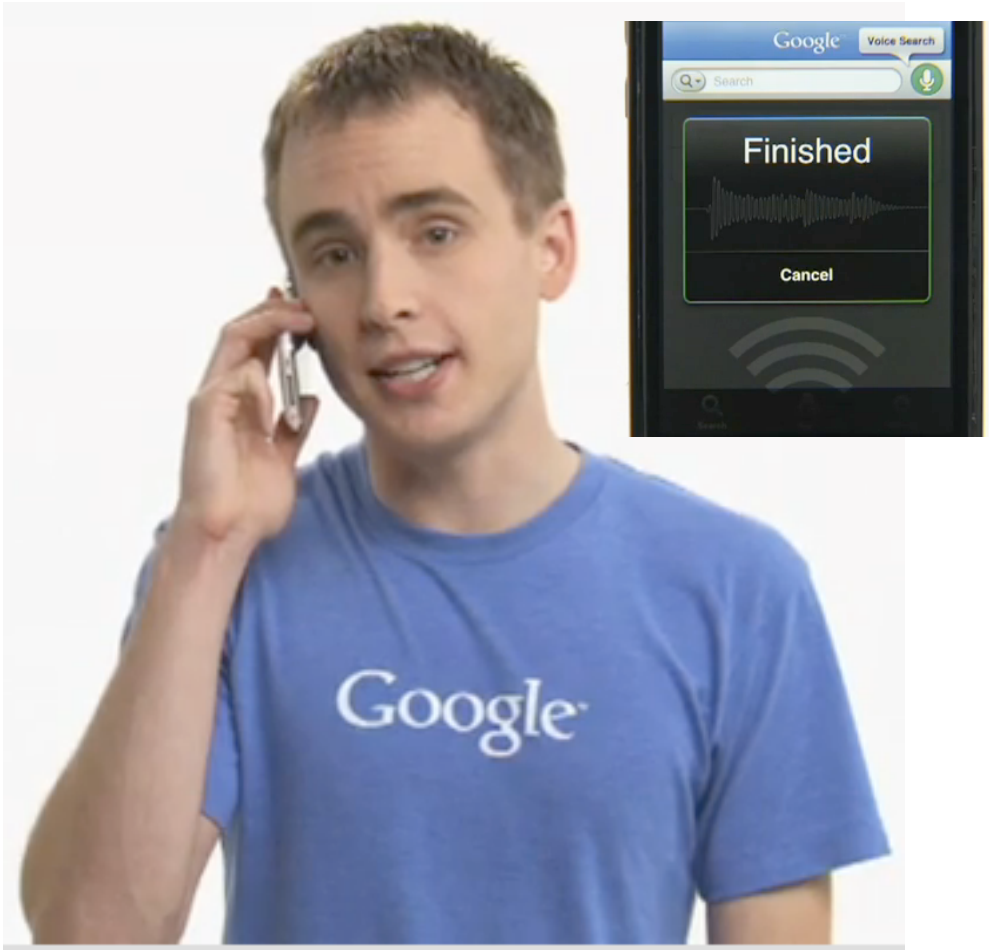
CALO

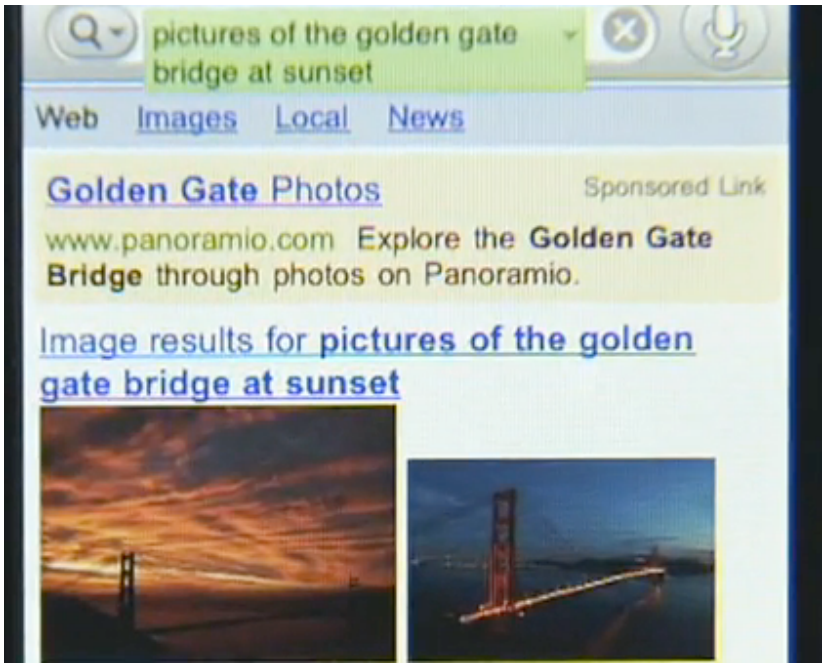
Original author(s)	SRI International
Type	Intelligent software assistant
License	Proprietary

The CALO effort has had many major spin-offs, most notably the Siri intelligent software assistant that is now part of the Apple iOS since iOS 5 in the iPhone 4S, iPhone 5, iPod Touch 5 and the New iPad; Social Kinetics, a social application that learned personalized intervention and treatment strategies for chronic disease patients, sold to RedBrick Health; the Trapit project, which is a web scraper and news aggregator that makes intelligent selections of web content based on user preferences; Tempo AI, a smart calendar; Desti, a personalized travel guide; and Kuato Studios, a game development startup.

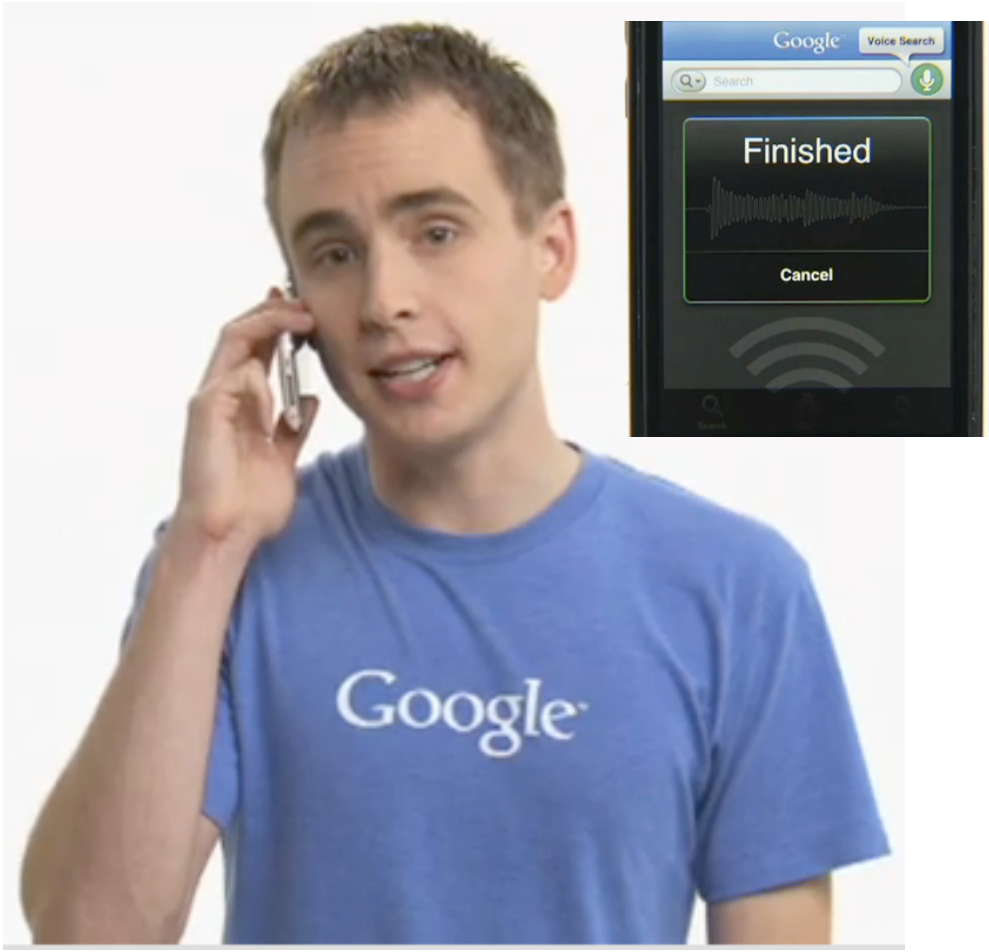
CALO was funded by the Defense Advanced Research Projects Agency (DARPA) under its Personalized Assistant that Learns (PAL) program.[1][2] DARPA's five-year contract brought together over 300 researchers from 25 of the top university and commercial research institutions, with the goal of building a new generation of cognitive assistants that can reason, learn from experience, be told what to do, explain what they are doing, reflect on their experience, and respond robustly to surprise. SRI International was the lead integrator responsible for coordinating the effort to produce an assistant that can live with and learn from its users, provide value to them, and then pass a yearly evaluation that measures how well the system has learned to do its job.

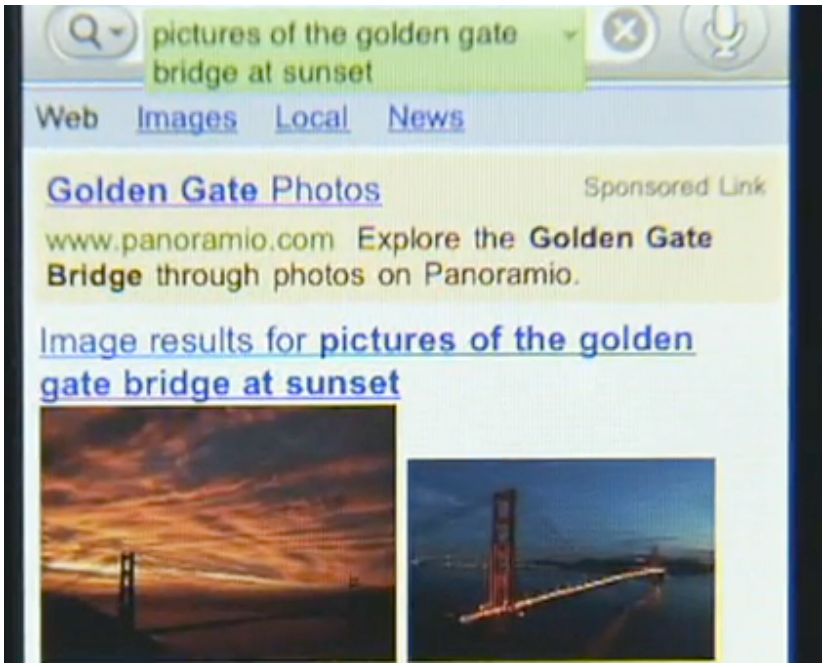
Exemple:
La recherche web par la parole





Another example that (almost) everyone now knows about





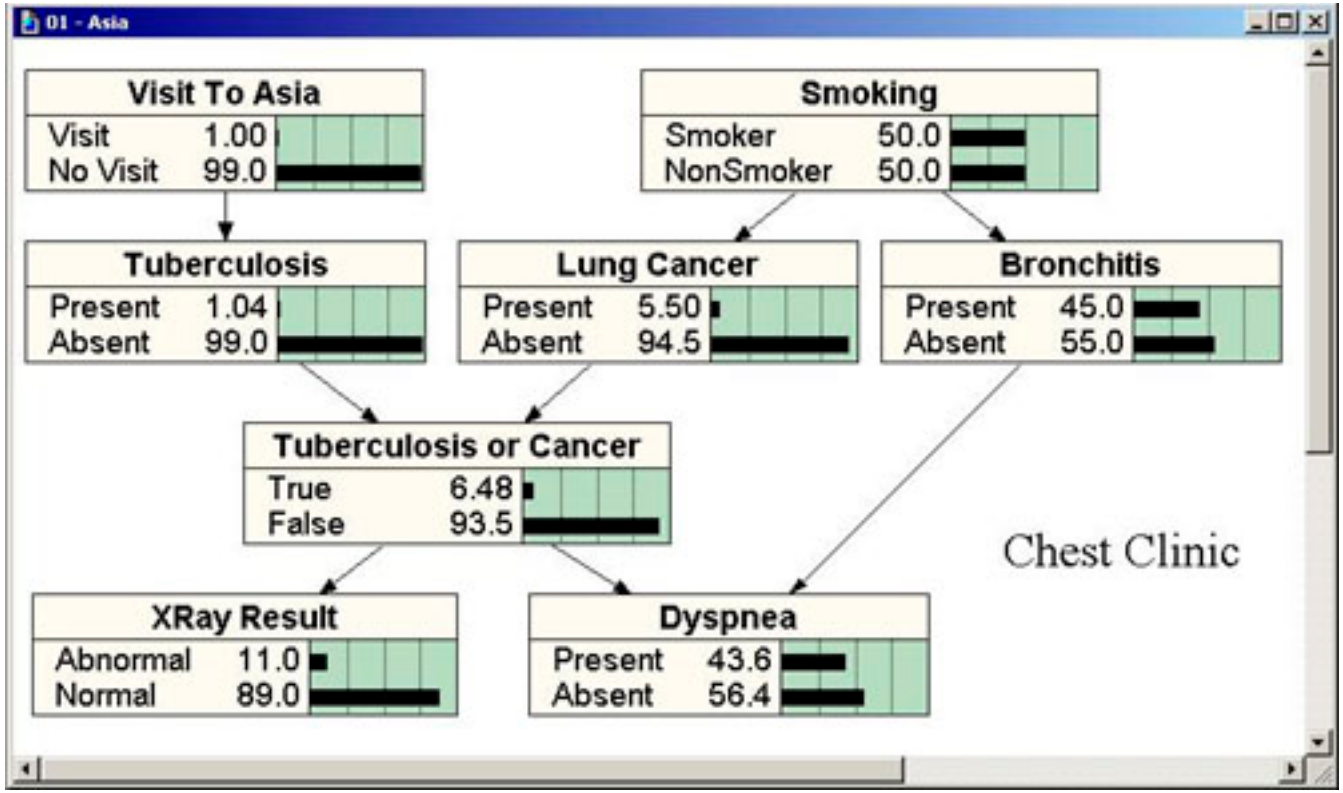
Example from Google

INF8225 - Thème 1
Connaître et penser l'incertain

- Quantification de l'incertitude
- Raisonnement probabiliste, simple et structurée

… et les applications

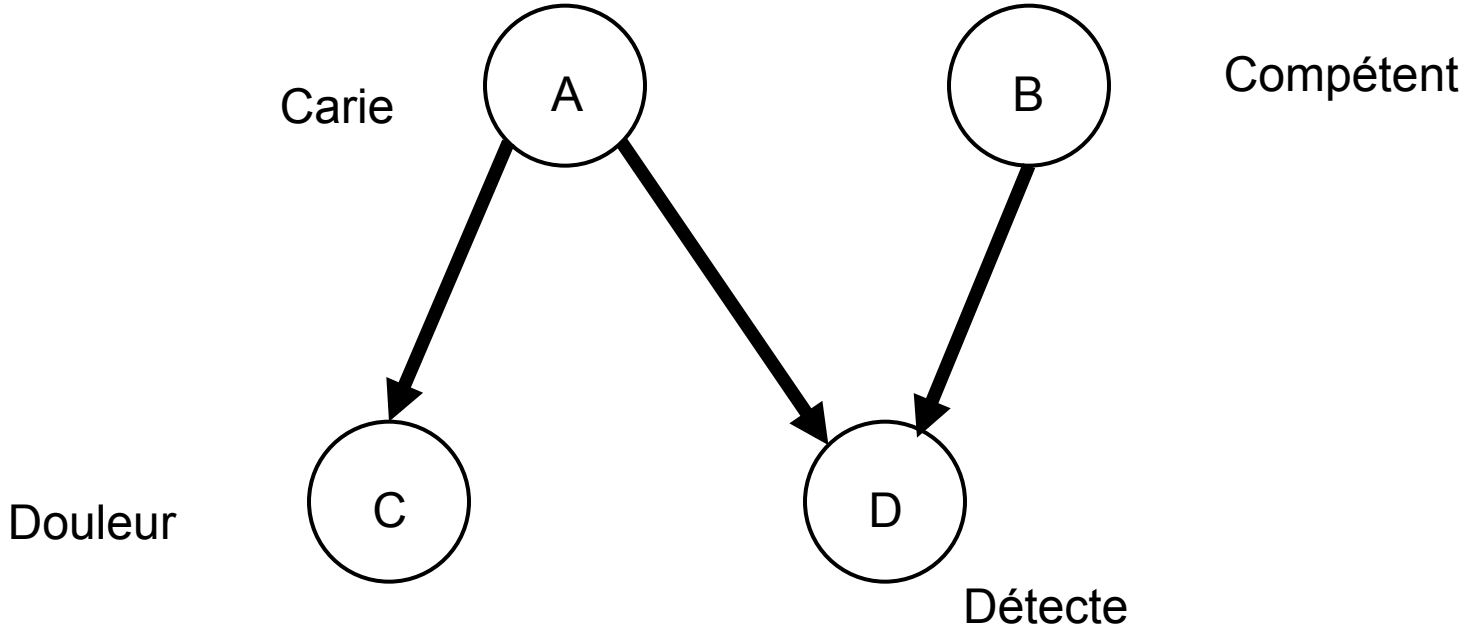
Exemple classique : le diagnostic médical



- Le logiciel « Netica » de Norsys (Vancouver)

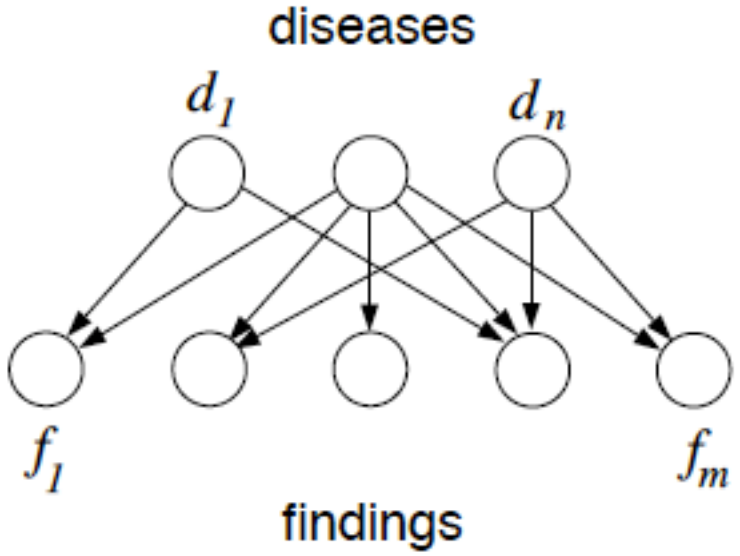
Réseaux bayésiens

- Il y a une correspondance entre la factorisation de la probabilité jointe et un graphe.
- Par exemple, avec le modèle P(A,C,B,D)=P(A)P(B)P(C|A)P(D|A,B)
- Nous avons un réseau deBayes avec la structure suivante :



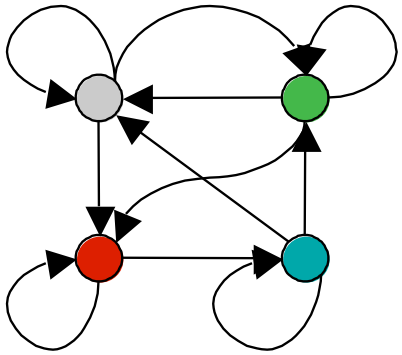
Exemple classique : le diagnostic médical

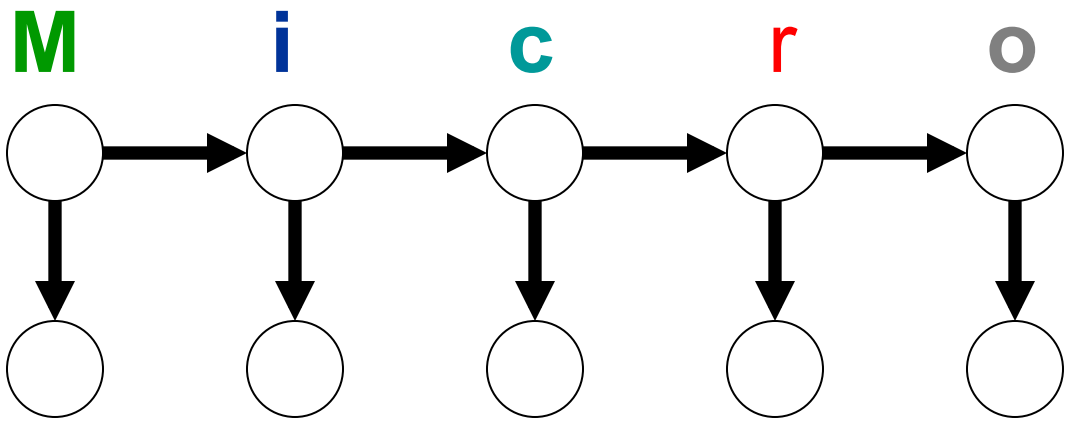
- QMR : « Quick Medical Reference »
- 600 « diseases »
- 4000 « findings »
- QMR-DT : « formulated as a bipartite graph »



Modèles probabiliste séquentiel

- Les modèles de Markov cachés (Hidden Markov Models HMMs)

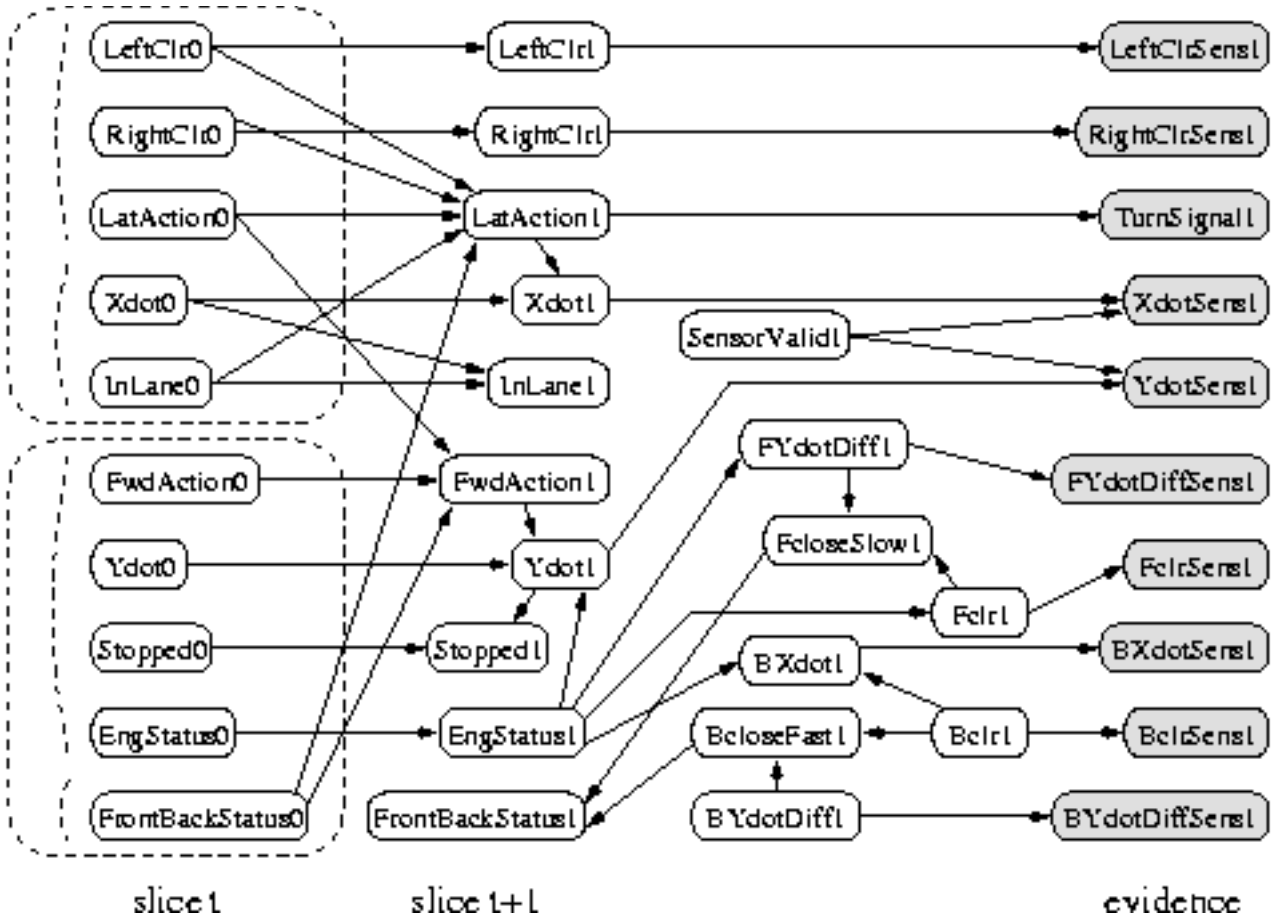




Today in Redmond, **Microsoft** Chair Bill Gates…

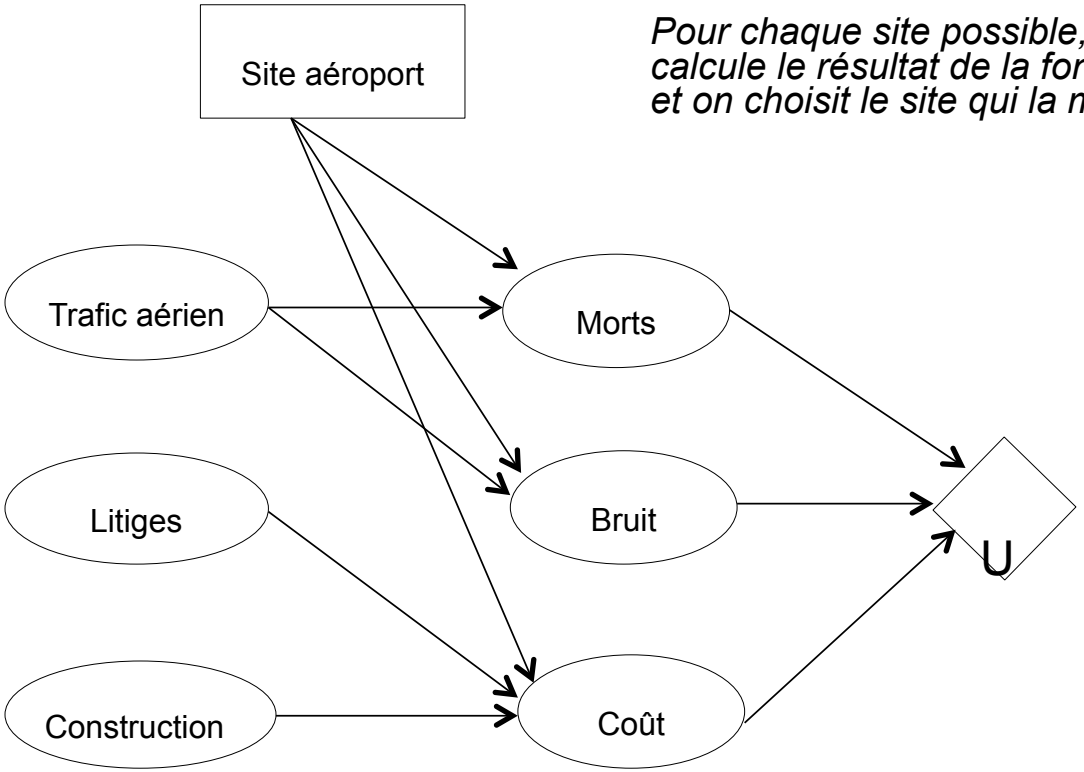
- La modélisation des séquences et des données temporels : ex. texte, parole, la reconnaissance des mot et l'extraction dinformation

Bayesian Automated Taxi (BAT) network



- Image de Daphne Koller (Stanford)

Exemple de réseau de décision



Pour chaque site possible, on calcule le résultat de la fonction d'utilité et on choisit le site qui la maximise.

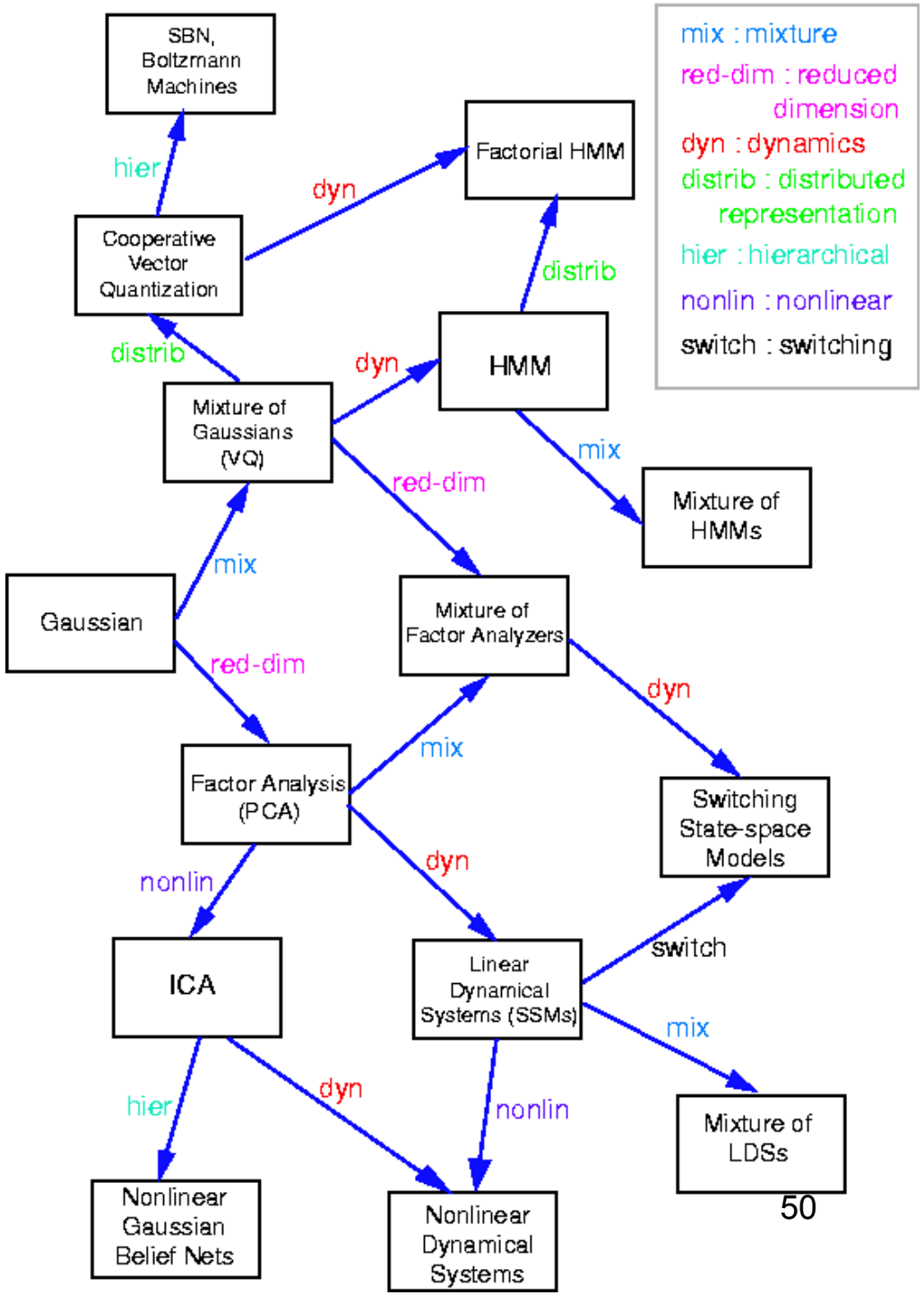
INF8225 - Thème 2
Apprentissage

- Apprendre à partir d'exemples
- Apprentissage de modèles probabilistes
- Apprentissage par renforcement
- « **Deep Learning** »

… et les applications

Les modèles probabilistes

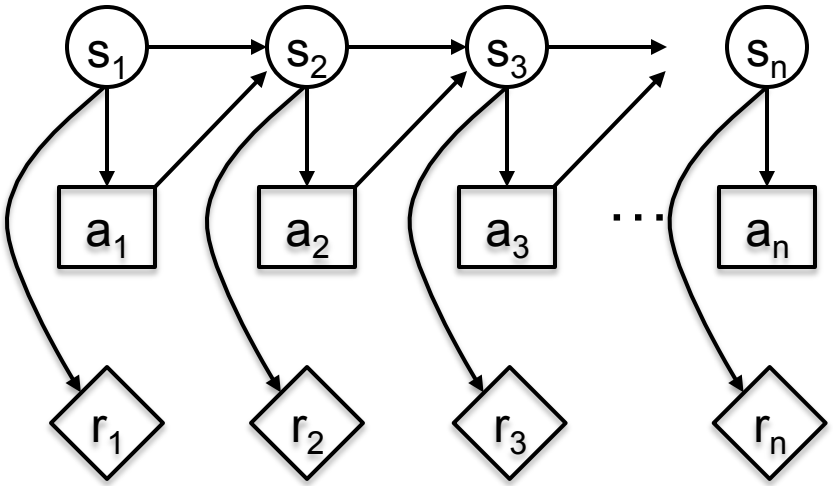
- Quelques relations entre des différentes types des modèles probabilistes



D’autres méthodes d’apprentissage statistique

- Les machines à vecteurs de support ou séparateurs à vaste marge (en anglais Support Vector Machines, SVMs)
- Les arbres de décision
- Apprentissage par renforcement (PDMs)

Un PDM comme un réseau de décision



- But: étant donné paramètres
  $R(s)$, $P(s_1)=s_o$, $P(s_{i+1}|s_i, a_i)=T(s',s,a)$, γ
- Calculez la politique $\pi(s)$ qui maximise

$$E\left[\sum_{t=1}^{\infty}\gamma^t R(s_t)\,|\,\pi(s)\right] = V_\pi(s) = R(s) + \gamma\sum_{s'}P(s'\,|\,s,\pi(s))V_\pi(s')$$

2005 DARPA Grand Challenge

robohub.org

- Won by Stanford Team (Stanley)
- Sebastien Thrun (CMU -> Stanford -> Google -> Udacity

Supposed Event from Recent History

- Anonymous Prof. 1: “They should just shut down NIPS” (the main neural networks conference).
  - Q: Why say that?
    In 2005 neural networks were very unpopular
  - Even the main neural networks conference had very few papers on neural networks
  - NIPS looked a lot like ICML, the International Conference on Machine Learning

2005-2013

- ~2011: “ AI is ‘dead’ ”,
  (the opinion of many professors)

2013 - 2017
AI Became (un) Dead

The success of IBM's Watson had already started to shift the perspectives of the research community, but AI really started capturing headlines as a result of the success of Deep Learning techniques for AI.

But even before the most recent explosion of interest there was…

THE MNIST DATABASE

of handwritten digits

- A very widely used dataset in the deep learning community with a well maintained list of results by:
  - Yann LeCun, Courant Institute, NYU
  - Corinna Cortes, Google Labs, New York
  - Christopher J.C. Burges, Microsoft Research, Redmond
- 60,000 training instances, 10,000 test instances, 28x28 pixel images

http://yann.lecun.com/exdb/mnist/

Image from LeCun et al. 1998

And we knew that…

- Convolutional neural networks '**work**' (on small images) (LeCun et al 1998)
- Synthetic transformations of visual input **work** (Simard et al., 2003)
- Plain old neural networks **work** (better) when they are deep (Ciresan et al. 2010)

CLASSIFIER	PRE-PROCESSING	TEST ERROR RATE (%)	REFERENCE
linear classifier (1-layer NN)	none	12.0	LeCun et al. 1998
K-nearest-neighbors, Euclidean (L2)	none	5.0	LeCun et al. 1998
2-layer NN, 300 hidden units, mean square error	none	4.7	LeCun et al. 1998
SVM, Gaussian Kernel	none	1.4	MNIST Website
Convolutional net LeNet-5, [no distortions]	none	**0.95**	LeCun et al. 1998
NOW ADD DISTORTIONS			
Virtual SVM, deg-9 poly, 2-pixel jittered	deskewing	0.56	DeCoste and Scholkopf, 2002
Convolutional net, cross-entropy [elastic distortions]	elastic distortions	0.4	Simard et al. 2003
6-layer NN (on GPU) [elastic distortions]	elastic distortions	**0.35**	Ciresan et al. 2010
Large/dee ConvNet [elastic distortions]	elastic distortions	**0.35**	Ciresan et al. 2011

- But what about *real* images and more complex

Why are Deep Neural Networks so Hot?

- **Large-Vocabulary Speech Recognition**
  Significant increase in performance (Dahl, Yu, Deng & Acero, 2012)
  - Deep Neural Network : 16-23% relative error rate reduction over the previous state-of-the-art

- **Visual Recognition of 1000 Classes**
  Top results on the ImageNet contest (Krizhevsky, Sutskever & Hinton, 2012), in 1.2 million images
  - Deep Neural Network : 15% top-5 error rate
  - Second best entry: 26% top-5 error rate

- **Face Verification**
  Near human performance, 97.35% accuracy on the Labeled Faces in the Wild (LFW) evaluation, reducing the error of the state of the art at the time by more than 27% (Facebook Research, 2014)

A Key Advance in Speech Recognition

- Notice that Watson did not use speech recognition
- However, IBM is well known for their world class speech recognition research
- Geoff Hinton lead a paper in 2012 on speech recognition entitled: “Deep Neural Networks for Acoustic Modeling in Speech Recognition – The shared views of four research groups”,
- Geoffrey Hinton, Li Deng, Dong Yu, George E. Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N. Sainath, and Brian Kingsbury
- The four groups being: University of Toronto, Microsoft Research, Google and IBM

2012 Article in the New York Times

WORLD-CLASS ENGINEERING

Scientists See Promise in Deep-Learning Programs

Hao Zhang/The New York Times

A voice recognition program translated a speech given by Richard F. Rashid, Microsoft's top scientist, into Mandarin Chinese.

By JOHN MARKOFF
Published: November 23, 2012

Using an artificial intelligence technique inspired by theories about how the brain recognizes patterns, technology companies are reporting startling gains in fields as diverse as computer vision, speech recognition and the identification of promising new molecules for designing drugs.

FACEBOOK
TWITTER
GOOGLE+
SAVE
EMAIL

- Live demo of speech recognition, translation and synthesis system
- Rick Rachid speaks in English, it is translated into Chinese and said by the computer
- Delay of a few seconds after each phrase

Example from NYTimes

IMAGENET

14,197,122 images, 21841 synsets indexed

Explore Download Challenges Publications CoolStuff About

Not logged in. Login I Signup

ImageNet is an image database organized according to the WordNet hierarchy (currently only the nouns), in which each node of the hierarchy is depicted by hundreds and thousands of images. Currently we have an average of over five hundred images per node. We hope ImageNet will become a useful resource for researchers, educators, students and all of you who share our passion for pictures.

Click here to learn more about ImageNet, Click here to join the ImageNet mailing list.

What do these images have in common? *Find out!*

Check out the ImageNet Challenge 2015!

In the 2012 Challenge, Deep ConvNets Significantly Outperformed the Competition (Task 1 and 2)

Team name	Error (5 guesses)	Description (Note Task 1 is classification)
SuperVision (U. Toronto)	**0.15315**	Using extra training data from ImageNet Fall 2011 release
SuperVision	**0.16422**	Using only supplied training data
ISI (U. Tokyo)	**0.26172**	Weighted sum of scores from each classifier with SIFT+FV, LBP+FV, GIST+FV, and CSIFT+FV, respectively.
OXFORD_VGG	**0.26979**	Mixed selection from High-Level SVM scores and Baseline Scores, decision is performed by looking at the validation performance
XRCE/INRIA	**0.27058**	
OXFORD_VGG	**0.27079**	High-Level SVM over Fine Level Classification score, DPM score and Baseline Classification scores (Fisher Vectors over Dense SIFT and Color Statistics)
OXFORD_VGG	**0.27302**	Baseline: SVM trained on Fisher Vectors over Dense SIFT and Color Statistics
University of Amsterdam	**0.29576**	See text above

http://www.image-net.org/challenges/LSVRC/2012/results.html

In the 2012 Challenge, Deep ConvNets Significantly Outperformed the Competition (Task 1 and 2)

Team name	Error (5 guesses)	Description (Note Task 2 is localization)
SuperVision	0.335463	Using extra training data for classification from ImageNet Fall 2011 release
SuperVision	**0.341905**	Using only supplied training data
OXFORD_VGG	0.500342	Re-ranked DPM detection over Mixed selection from High-Level SVM scores and Baseline Scores, decision is performed by looking at the validation performance
OXFORD_VGG	0.50139	Re-ranked DPM detection over High-Level SVM Scores
OXFORD_VGG	0.522189	Re-ranked DPM detection over High-Level SVM Scores - First bbox selection heuristic
OXFORD_VGG	0.529482	DPM detection over baseline classification scores

http://www.image-net.org/challenges/LSVRC/2012/results.html

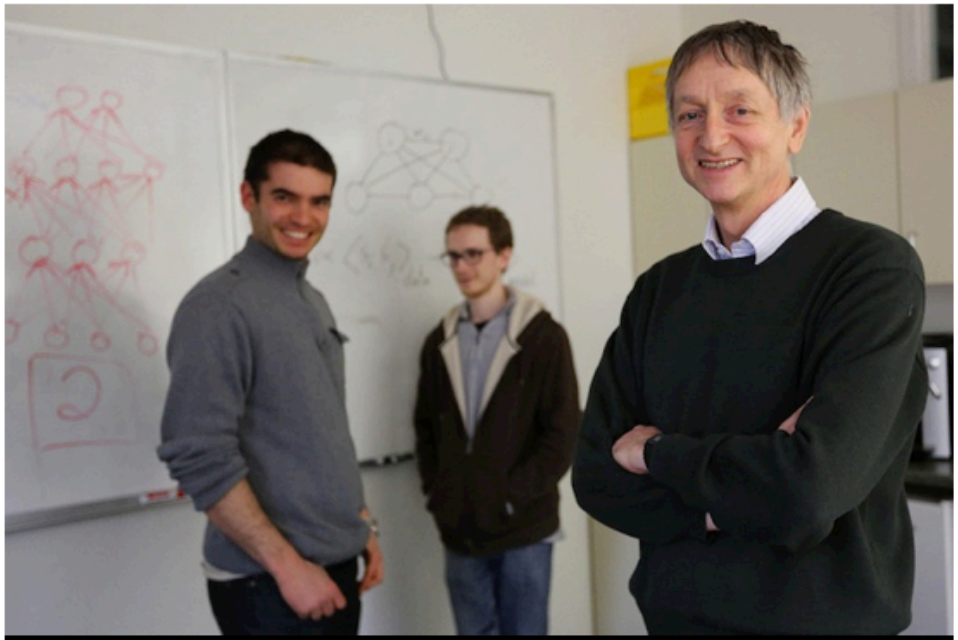
2013 article in Wired & New developments at Facebo

WORLD-CLASS ENGINEERING

ogle Hires Brains that Helped Supercharge | Share 719

Google Hires Brains that Helped Supercharge Machine Learning

BY ROBERT MCMILLAN 03.13.13 6:30 AM



Geoffrey Hinton (right) Alex Krizhevsky, and Ilya Sutskever (left) will do machine learning work at Google. *Photo: U of T*

Google has hired the man who showed how to make computers learn much like the human brain.

- After winning ImageNet challenge
- Google buys Geoff Hinton's company

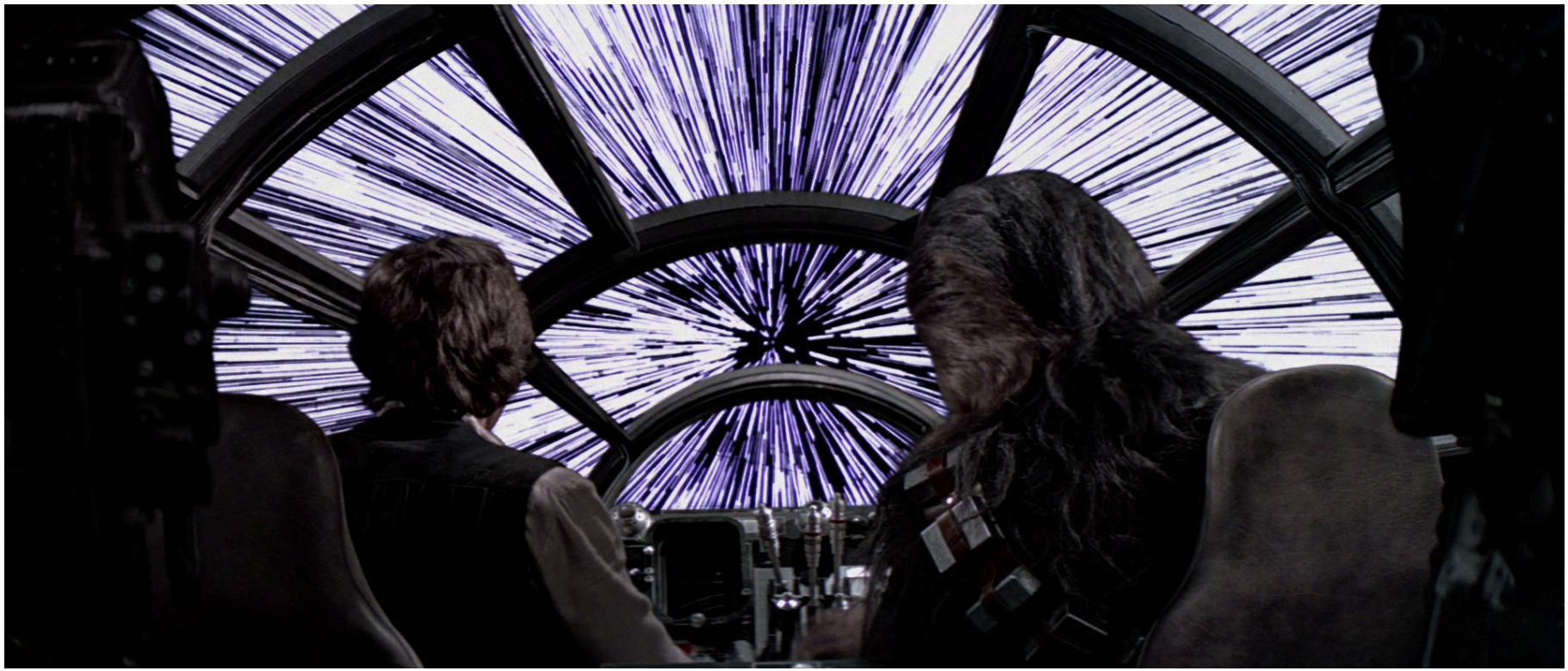
facebook

Yann LeCun

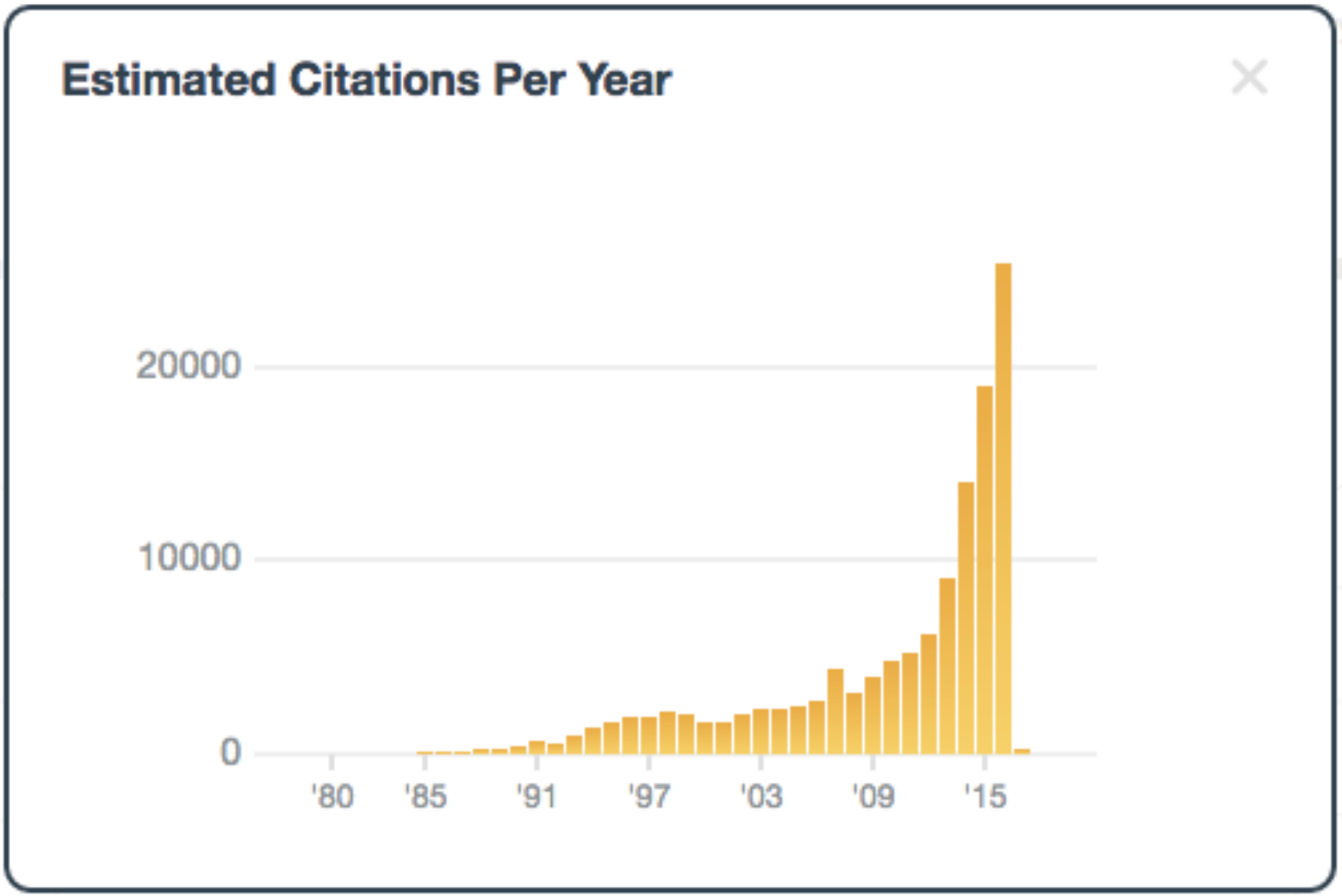
December 9, 2013 ·

- Facebook has created a new research lab with the ambitious, long-term goal of bringing about major advances in Artificial Intelligence.
- Simultaneously, Facebook and New York University's Center for Data Science are entering a partnership to carry out research in data science, machine learning, and AI.

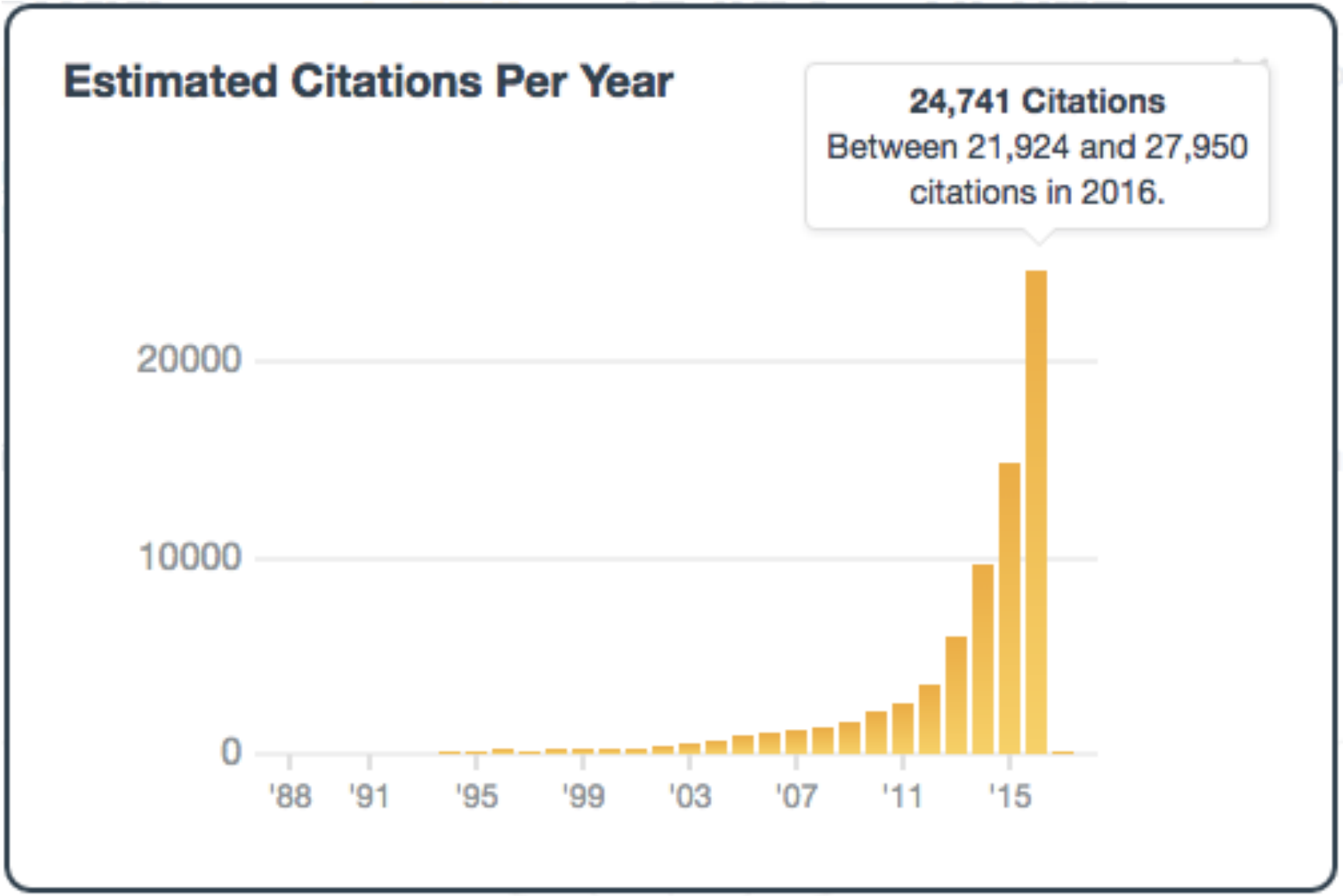
The level of interest in AI and Deep Learning has jumped to hyperintense



Geoff Hinton's citations



Prof. Bengio's Citations



Some Recent Headlines about Montréal

September 2016: Minister Bibeau Announces $213,187,000 to Transform Research at Université de Montreal, HEC Montreal, Polytechnique Montreal and McGill University

Google opens new AI lab in Montreal

January 13, 2017

Maluuba + Microsoft

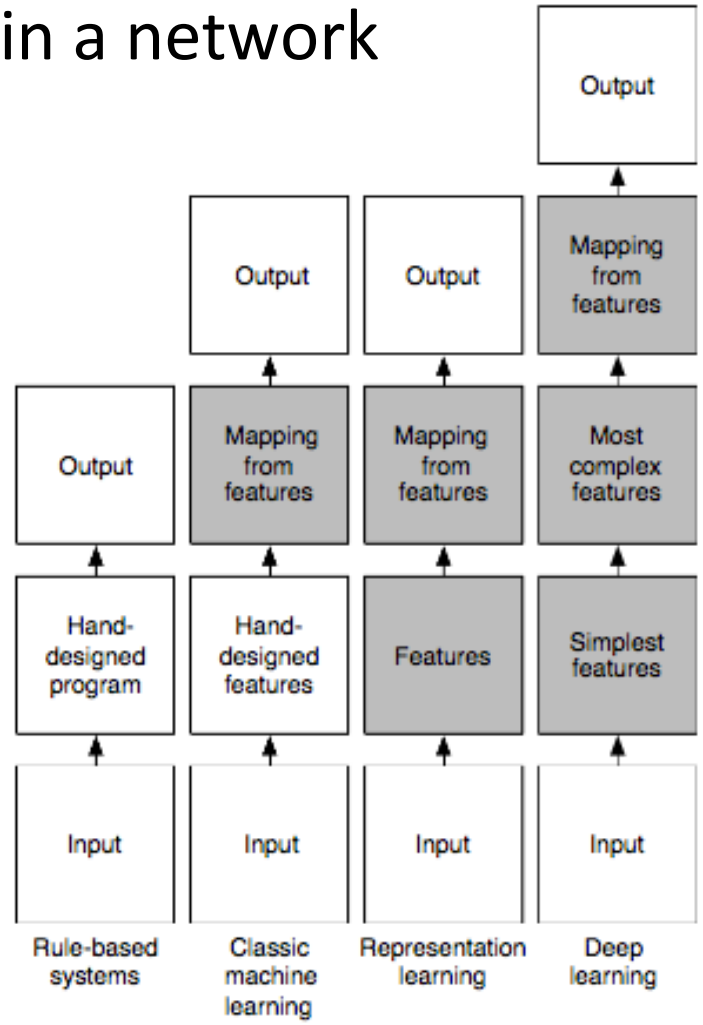
What are the key factors leading to this Scientific Revolution?

- Increased high quality labelled data for training, validation and testing
- The use of Graphics Processing Units or GPU technology to accelerate learning
- Proper, open scientific evaluations where all teams have the same training, valid and test data
- Some key mathematical insights for training deep neural networks
- Initially one had to use specialized low level code (NVIDIA CUDA) now we have higher level tools Theano, Tensorflow, Torch, etc.

Why does Deep Learning 'Work' so well

- Deep Learning methods seek to learn representations through a sequence of transformations of the data, typically though of as layers in a network

- Flow charts from Bengio et al. (2014) "showing how the different parts of an AI system relate to each other within different AI disciplines.
- Shaded boxes indicate components that are able to learn from data"



Visualizing the filters learned by a CNN

- Learned edge-like filters and texture-like filters are frequently observed in the early layers of CNNs trained using natural images
- Since each layer in a CNN involves filtering the feature map below, so as one moves up the receptive fields become larger
- Higher- level layers learn to detect larger features, which often correspond to textures, then small pieces of objects

First Layer

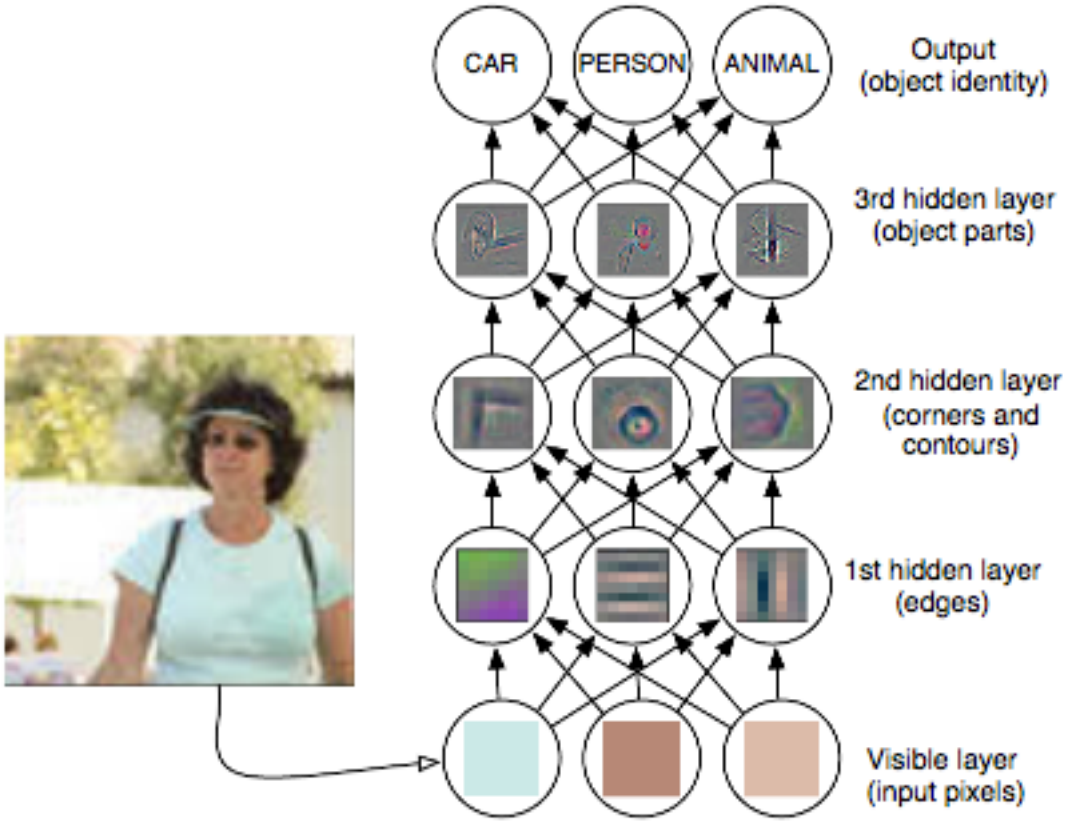
Second Layer

Third Layer

(Imagery kindly provided by Matthew Zeiler)

- Above are the strongest activations of random neurons projecting the activation back into image space using the deconvolution approach of Zeiler and Fergus (2013).

Intuitions about Deep Learning



- Deep Learning methods seek to learn representations through a sequence of transformations of the data, typically though of as layers in a network

Image from Bengio, GoodFellow & Courville (2014) /
Originally from From Zeiler and Fergus (2014)

GPU computing is basically essential for training and experiment with models

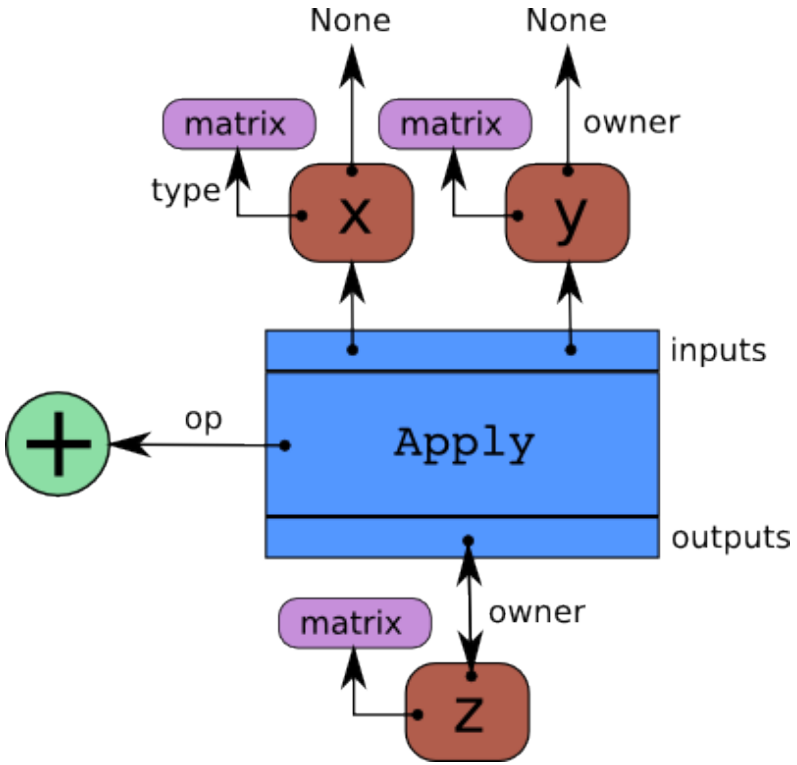
- Convolution operations and matrix multiplies are parallelize very well using GPUs
- Model training can be 30-60x faster using GPUs
- Example: Experiment using the famous Oxford VGG-convolutional neural network
- The model was trained on 1.28 million images

CPU: 2x Xeon e5-2699v4 server takes **55 days**

GPU: NVIDIA DGX-1 Trains in 23 hours or **1 day**

Theano (made by MILA)

```
import theano.tensor as T
x = T.dmatrix('x')
y = T.dmatrix('y')
z = x + y
```



- Create a computation graph, encoding interactions between instances of:
- Apply (blue),
- Variable (red),
- Ops (green), and
- Types (purple)
- Allows you to compute derivatives symbolically
- Compile large graphs and create deep networks easily
- Heavily influenced/inspired Google's Tensor Flow

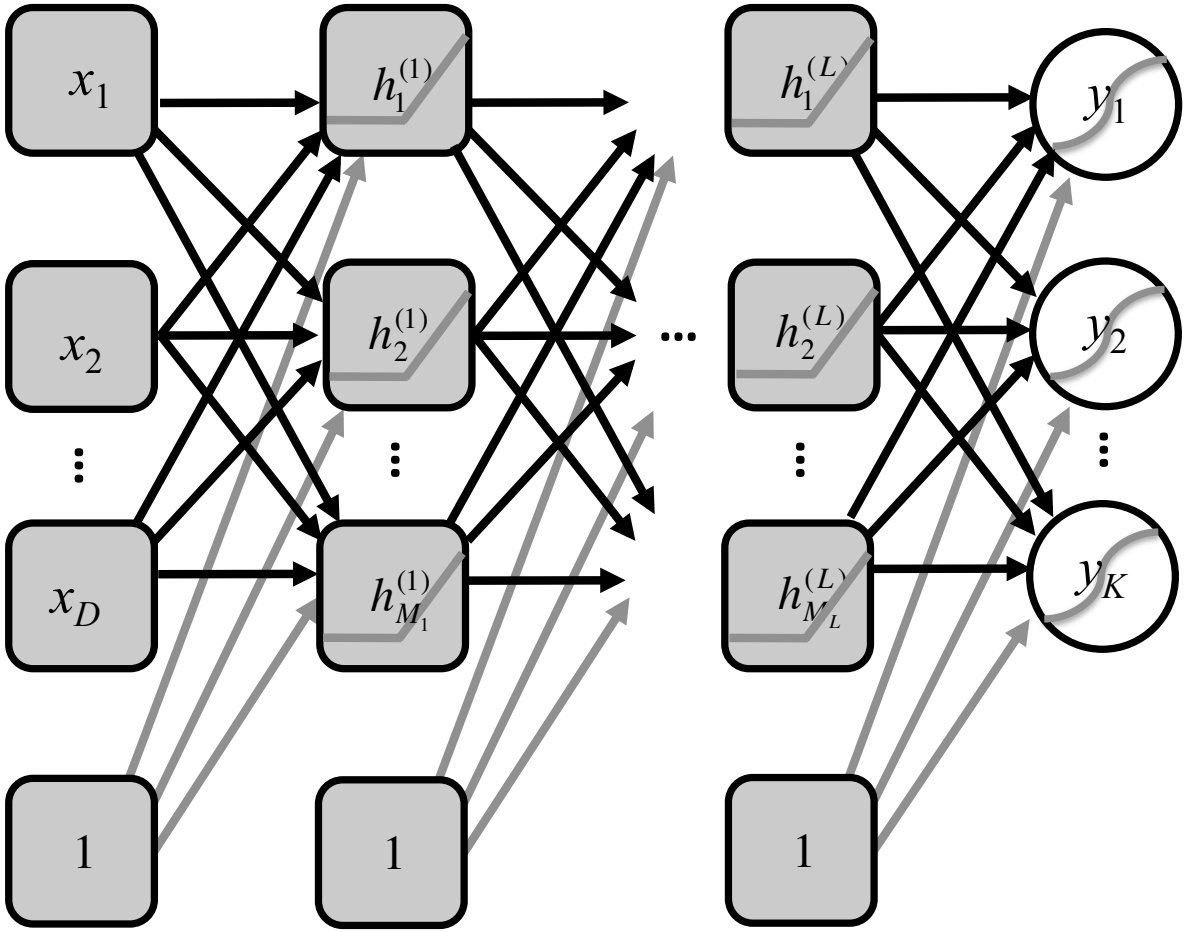
http://deeplearning.net/software/theano/tutorial/symbolic_graphs.html

A Brief Overview of the Main Deep Learning Architectures

Classic Feedforward Neural Networks

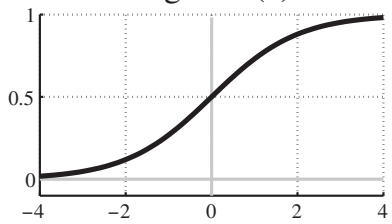
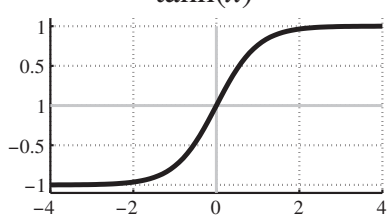
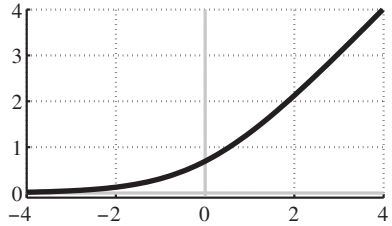
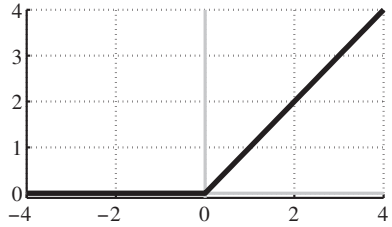
Multilayer Perceptrons

Deep feedforward networks



- Unlike Bayesian networks the hidden units here are *intermediate deterministic computations* not random variables, which is why they are not represented as circles
- However, the output variables y_k are drawn as circles because they can be formulated probabilistically

Activation functions

Name and Graph	Function	Derivative
sigmoid(x) 	$h(x) = \frac{1}{1 + \exp(-x)}$	$h'(x) = h(x)[1 - h(x)]$
tanh(x) 	$h(x) = \frac{\exp(x) - \exp(-x)}{\exp(x) + \exp(-x)}$	$h'(x) = 1 - h(x)^2$
softplus(x) 	$h(x) = \log(1 + \exp(x))$	$h'(x) = \frac{1}{1 + \exp(-x)}$
rectify(x) 	$h(x) = \max(0, x)$	$h'(x) = \begin{cases} 1 \text{ if } x \geq 0 \\ 0 \text{ if } x < 0 \end{cases}$

Deep neural network architectures

- Compose computations performed by many layers
- Denoting the output of hidden layers by $\mathbf{h}^{(l)}(\mathbf{x})$, the computation for a network with *L* hidden layers is:

$$\mathbf{f}(\mathbf{x}) = \mathbf{f}\left[\mathbf{a}^{(L+1)}\left(\mathbf{h}^{(L)}\left(\mathbf{a}^{(L)}\left(\ldots\left(\mathbf{h}^{(2)}\left(\mathbf{a}^{(2)}\left(\mathbf{h}^{(1)}\left(\mathbf{a}^{(1)}(\mathbf{x})\right)\right)\right)\right)\right)\right)\right]$$

- Where *pre-activation functions* $\mathbf{a}^{(l)}(\mathbf{x})$ are typically linear, of the form $\mathbf{a}^{(l)}(\mathbf{x}) = \mathbf{W}^{(l)}\mathbf{x} + \mathbf{b}^{(l)}$ with matrix $\mathbf{W}^{(l)}$ and bias $\mathbf{b}^{(l)}$
- This formulation can be expressed using a single parameter matrix θ with the trick of defining $\hat{\mathbf{x}}$ as x with a 1 appended to the end of the vector; we then have

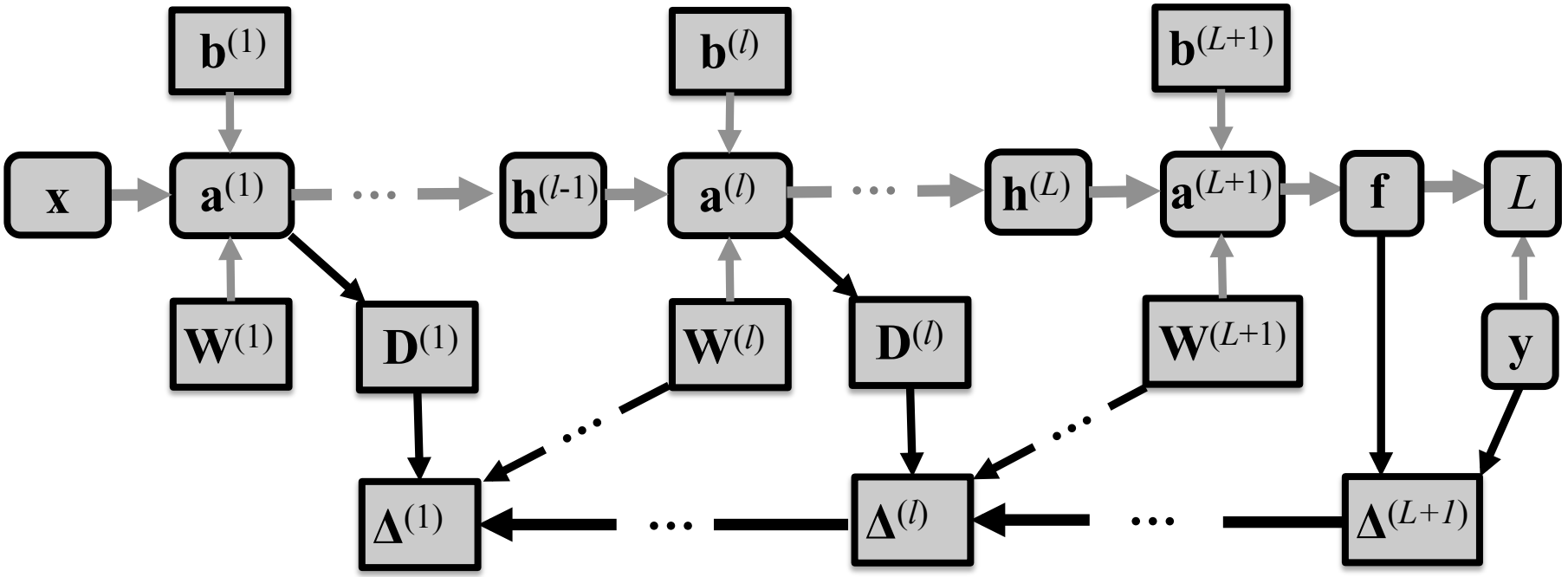
$$\mathbf{a}^{(l)}(\hat{\mathbf{x}}) = \boldsymbol{\theta}^{(l)}\hat{\mathbf{x}} \qquad , l = 1$$

$$\mathbf{a}^{(l)}(\hat{\mathbf{h}}^{(l-1)}) = \boldsymbol{\theta}^{(l)}\hat{\mathbf{h}}^{(l-1)} \qquad , l > 1$$

Backpropagation revisited in vector matrix form

Visualizing backpropagation

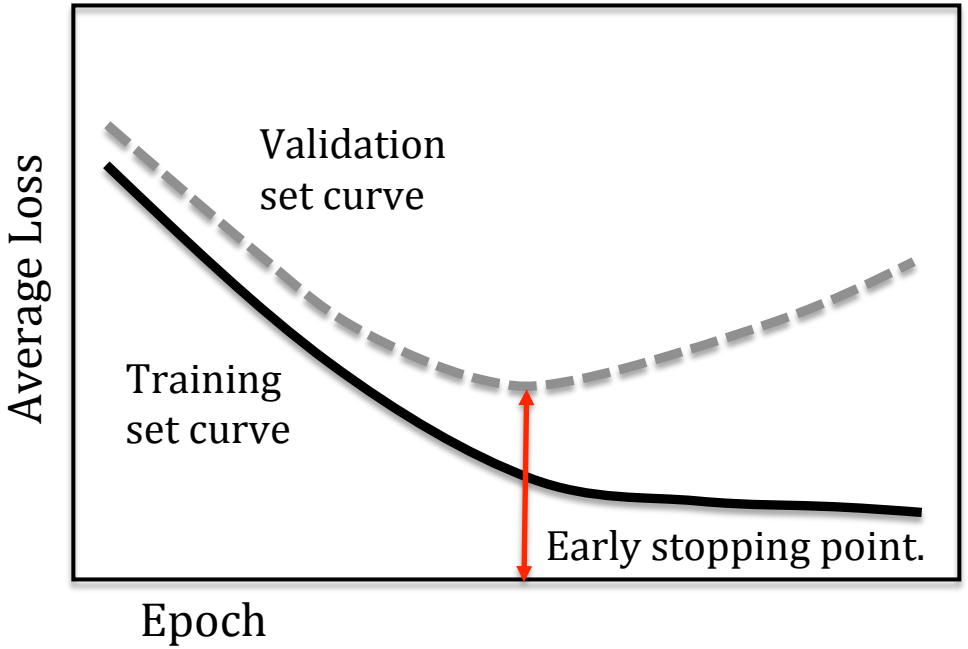
- In the backward propagation phase we compute terms of the form below



$$\Delta^{(l)} = \mathbf{D}^{(l)}\mathbf{W}^{\mathrm{T}(l+1)}\Delta^{(l+1)} \qquad \frac{\partial\,\mathbf{h}^{(l)}}{\partial\,\mathbf{a}^{(l)}} = \mathbf{D}^{(l)} \qquad \frac{\partial\,\mathbf{a}^{(l+1)}}{\partial\,\mathbf{h}^{(l)}} = \mathbf{W}^{\mathrm{T}(l+1)} \qquad \frac{\partial\,L}{\partial\,\mathbf{a}^{(L+1)}} = -\left[\mathbf{y} - \mathbf{f}(\mathbf{x})\right] = -\Delta^{(L+1)}$$

Training and evaluating deep networks

Early stopping

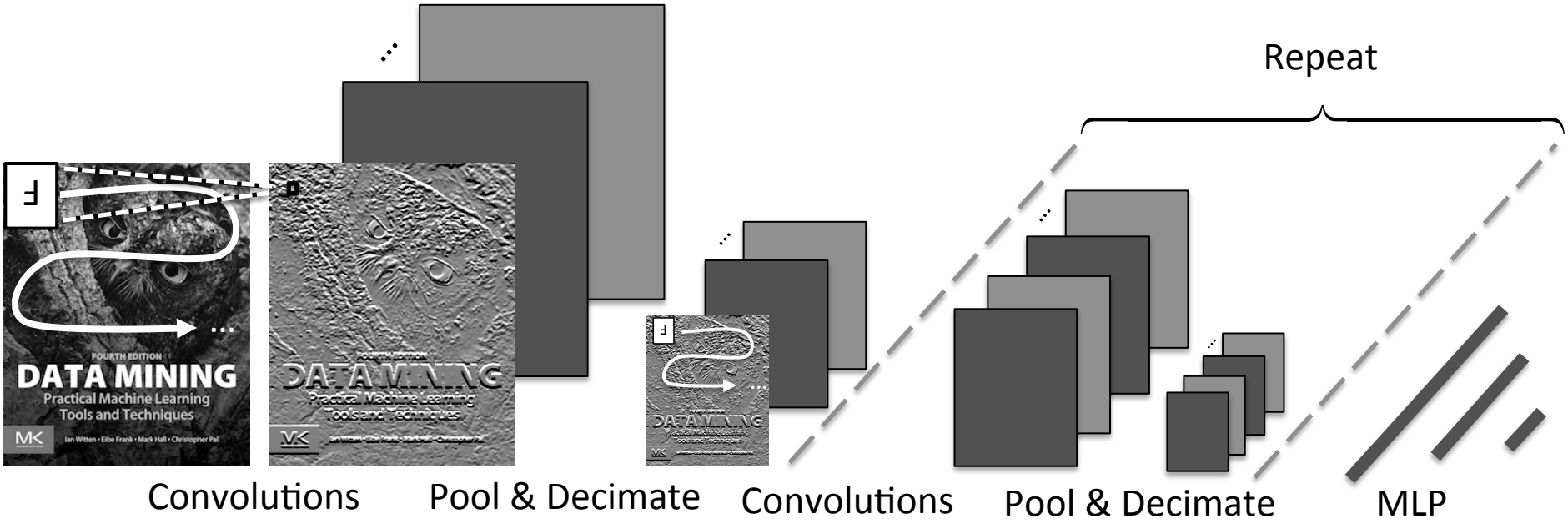


- In practice the curves above can be more noisy due to the use of stochastic gradient descent
- As such, it is common to keep the history of the validation set curve when looking for the minimum – even if it goes back up it might come back down

Convolutional neural networks

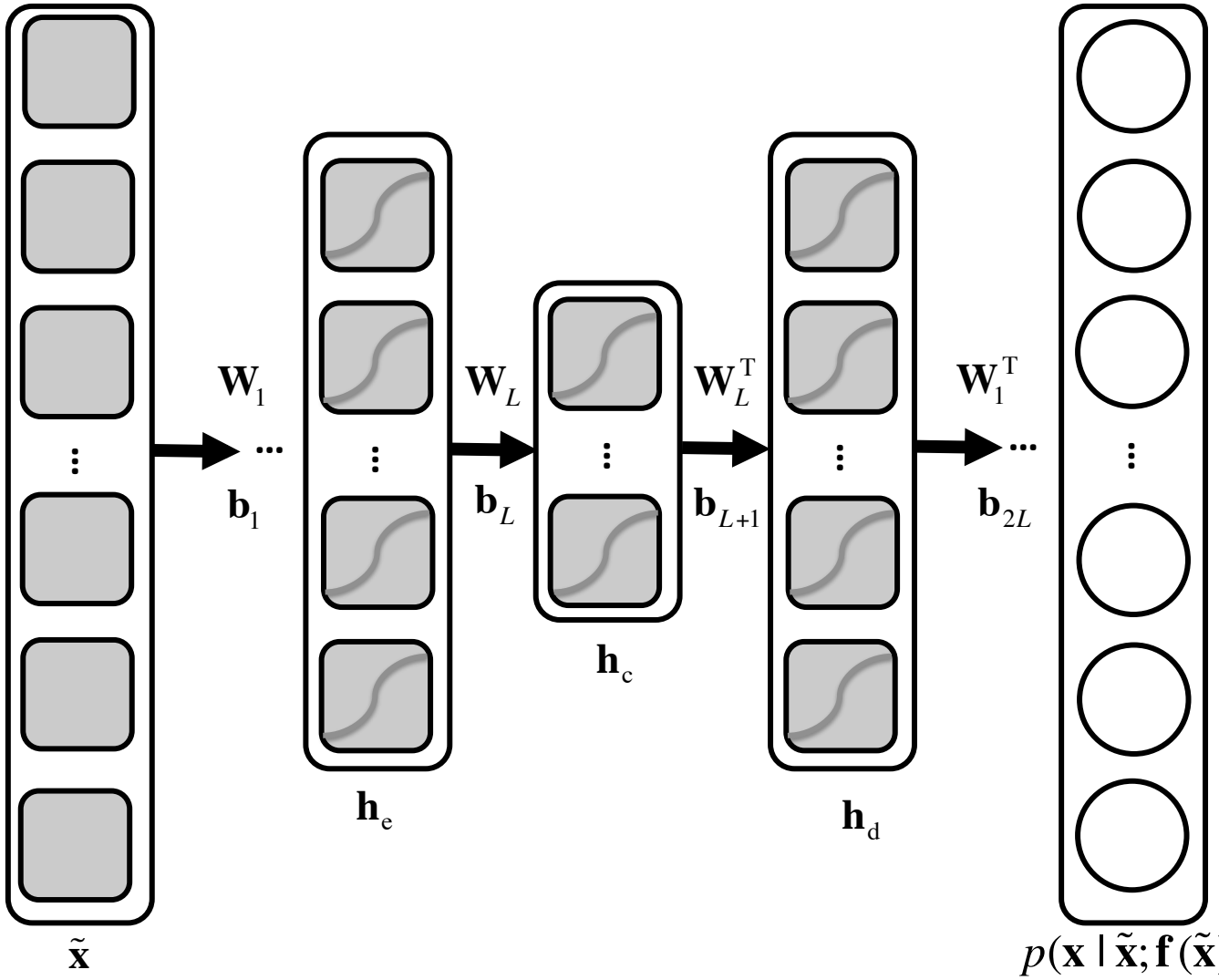
A typical CNN architecture

- Many feature maps are obtained from convolving learnable filters across an image
- Results are aggregated or pooled & decimated
- Process repeats until last set of feature maps are given to an MLP for final prediction



Autoencoders

A deep autoencoder

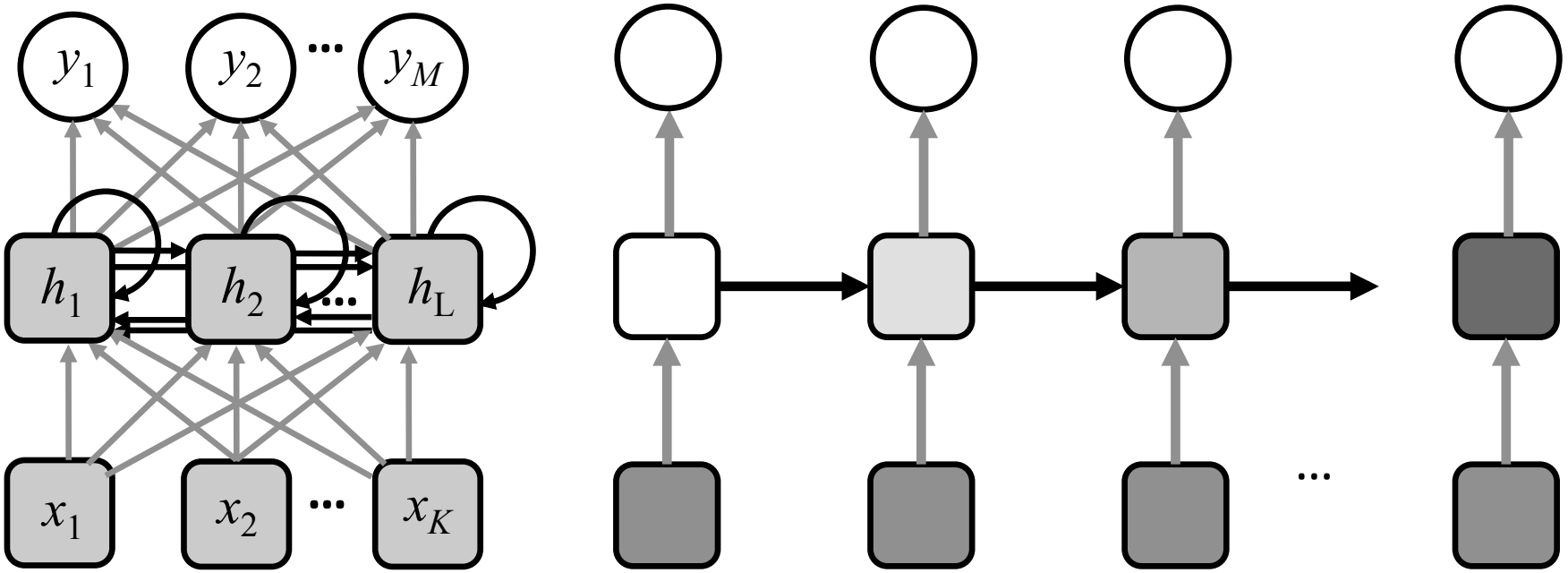


$$\mathbf{f}(\mathbf{x}) = \mathbf{f}_{\mathbf{d}}(\mathbf{a}_{\mathbf{d}}^{(2L)}(\ldots\mathbf{h}_{\mathbf{d}}^{(L+1)}(\mathbf{a}_{\mathbf{d}}^{(L+1)}(\mathbf{h}_{\mathbf{c}}^{(L)}(\mathbf{a}_{\mathrm{e}}^{(L)}(\ldots\mathbf{h}_{\mathrm{e}}^{(1)}(\mathbf{a}_{\mathrm{e}}^{(1)}(\mathbf{x}))))))))).$$

Recurrent neural networks

Recurrent neural networks (RNNs)

- An RNN can be unwrapped in time and implemented using the same weights and biases at each step to link units over time
- The resulting unwrapped RNN is similar to a hidden Markov model, but keep in mind that the hidden units in RNNs are not stochastic



Deep learning software

Theano

- A library in Python which has been developed with the specific goal of facilitating research in deep learning (Bergstra et al., 2010; Theano Development Team, 2016)
- It is also a powerful general purpose tool for general mathematical programming
- Theano extends NumPy (the main Python package for scientific computing) by adding symbolic differentiation and GPU support, among various other functions
- It provides a high-level language for creating the mathematical expressions that underlie deep learning models, and a compiler that takes advantage of deep learning techniques, including calls to GPU libraries, to produce code that executes quickly
- Theano supports execution on multiple GPUs

Deep learning software

Theano

- Allows the user to declare symbolic variables for inputs and targets, and supply numerical values only when they are used
- Shared variables such as weights and biases are associated with numerical values stored in NumPy arrays
- Theano creates symbolic graphs as a result of defining mathematical expressions involving the application of operations to variables
  - These graphs consist of *variable*, *constant*, *apply* and *operation* nodes
  - Constants, and constant nodes, are a subclass of variables, and variable nodes, which hold data that will remain constant and can therefore be subjected to various optimizations by the compiler
- Theano is an open-source project using a BSD license

Deep learning software

Tensor Flow

- C++ and Python based software library for the types of numerical computation typically associated with deep learning (Abadi et al., 2016)
- It is heavily inspired by Theano, and, like it, uses dataflow graphs to represent the ways in which multidimensional data arrays communicate between one another
- These multidimensional arrays are referred to as “tensors.”
- Tensor Flow also supports symbolic differentiation and execution on multiple GPUs
- It was released in 2015 and is available under the Apache 2.0 license

Deep learning software

Torch / PyTorch

- Torch: An open-source machine learning library built using C and a high-level scripting language known as Lua (Collobert et al., 2011)
- PyTorch – a Python interface to Torch
- It uses multidimensional array data structures, and supports various basic numerical linear algebra manipulations
- It has a neural network package with modules that permit the typical forward and backward methods needed for training neural networks
- It also supports automatic differentiation

Deep learning software

Computational Network Toolkit (CNTK)

- C++ library for manipulating computational networks (Yu et al., 2014)
- It was produced by Microsoft Research, but has been released under a permissive license
- It has been popular for speech and language processing, but also supports convolutional networks of the type used for images
- It supports execution on multiple machines and using multiple GPUs

Deep learning software

Caffe

- C++ and Python based BSD-licensed convolutional neural network library (Jia et al., 2014).
- Has a clean and extensible design which makes it a popular alternative to the original open-source implementation of Krizhevsky et al. (2012)'s famous AlexNet that won the 2012 ImageNet challenge.

Deep learning software

Deeplearning4j

- Java-based open-source deep learning library available under the Apache 2.0 license
- Uses an multidimensional array class and provides linear algebra and matrix manipulation support similar to that provided by Numpy

Deep learning software

Lasagne, Keras and cuDNN

- Lasagne is a lightweight Python library built on top of Theano that simplifies the creation of neural network layers
- Similarly, Keras is a Python library that runs on top of either Theano or TensorFlow (Chollet, 2015) that allows one to quickly define a network architecture in terms of layers and also includes functionality for image and text preprocessing
- cuDNN is a highly optimized GPU library for NVIDIA units that allows deep learning networks to be trained more quickly
  - It can dramatically accelerate the performance of a deep network and is often called by the other packages above.