Etat des lieux des recherches en Deep Learning

Thierry Artières

QARMA (Machine Learning) team
Head of Data Science Department - LIS lab
Professor at Ecole Centrale Marseille

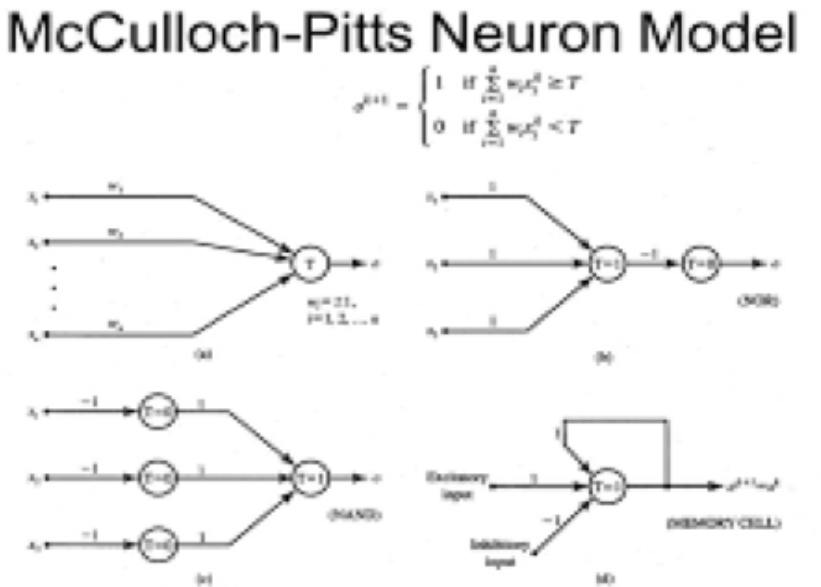
June 26, 2018

Outline

History

Key dates

- 1943 : Formal neuron [McCuloch-Pitts]
- 1950 : Organization of neurons and learning rules [Hebb]
- 1960 : Perceptron [Rosenblatt]
- 1960 : Update rule [Widrow Hoff]
- 1969 : Limitations of the Perceptron [Minsky]
- 1980s : Back-propagation [Rumelhart and Hinton]
- 1990s : Convolutional Networks [LeCun and al.]
- 1990s: Long Short Term Memory networks [Hochreiter and Schmidhuber]
- 2006 : Paper on Deep Learning in Nature [Hinton and al.]
- 2012 : Imagenet Challenge Win [Krizhevsky, Sutskever, and Hinton]
- 2013 : First edition of ICLR
- 2013 : Memory networks [Weston and al.]
- 2014 : Adversarial Networks [Goodfelow and al.]
- 2014 : Google Net [Szegedy and al.]
- 2015 : Residual Networks [He et al.]



History

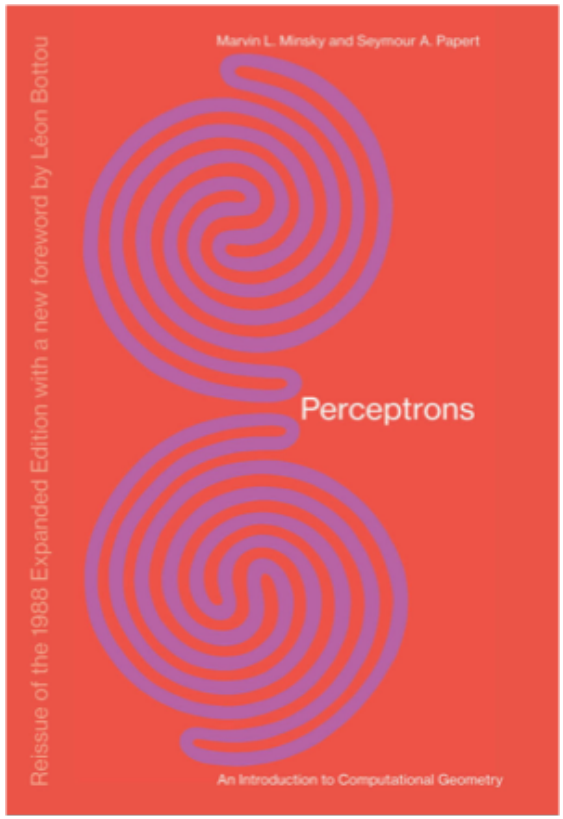
Key dates

- 1943 : Formal neuron [McCuloch-Pitts]
- 1950 : Organization of neurons and learning rules [Hebb]
- 1960 : Perceptron [Rosenblatt]
- 1960 : Update rule [Widrow Hoff]
- 1969 : Limitations of the Perceptron [Minsky]
- 1980s : Back-propagation [Rumelhart and Hinton]
- 1990s : Convolutional Networks [LeCun and al.]
- 1990s: Long Short Term Memory networks [Hochreiter and Schmidhuber]
- 2006 : Paper on Deep Learning in Nature [Hinton and al.]
- 2012 : Imagenet Challenge Win [Krizhevsky, Sutskever, and Hinton]
- 2013 : First edition of ICLR
- 2013 : Memory networks [Weston and al.]
- 2014 : Adversarial Networks [Goodfelow and al.]
- 2014 : Google Net [Szegedy and al.]
- 2015 : Residual Networks [He et al.]

History

Key dates

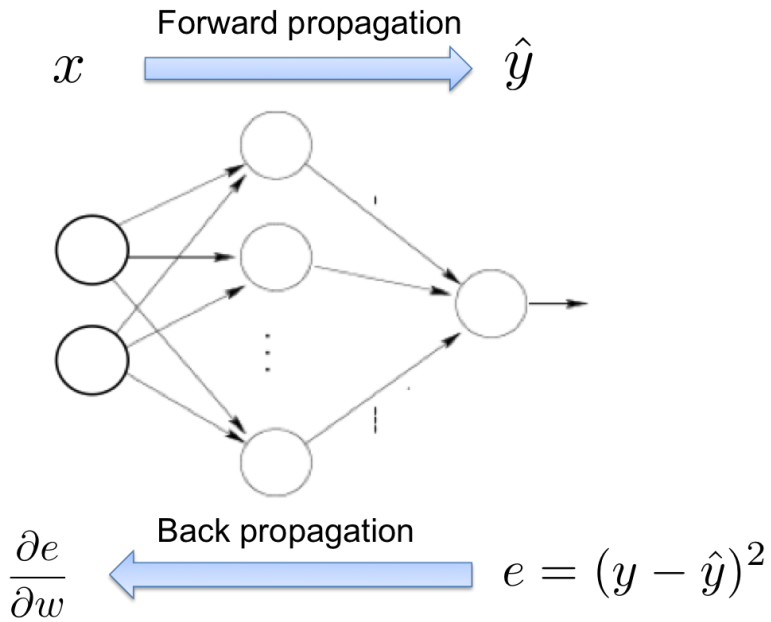
- 1943 : Formal neuron [McCuloch-Pitts]
- 1950 : Organization of neurons and learning rules [Hebb]
- 1960 : Perceptron [Rosenblatt]
- 1960 : Update rule [Widrow Hoff]
- 1969 : Limitations of the Perceptron [Minsky]
- 1980s : Back-propagation [Rumelhart and Hinton]
- 1990s : Convolutional Networks [LeCun and al.]
- 1990s: Long Short Term Memory networks [Hochreiter and Schmidhuber]
- 2006 : Paper on Deep Learning in Nature [Hinton and al.]
- 2012 : Imagenet Challenge Win [Krizhevsky, Sutskever, and Hinton]
- 2013 : First edition of ICLR
- 2013 : Memory networks [Weston and al.]
- 2014 : Adversarial Networks [Goodfelow and al.]
- 2014 : Google Net [Szegedy and al.]
- 2015 : Residual Networks [He et al.]



History

Key dates

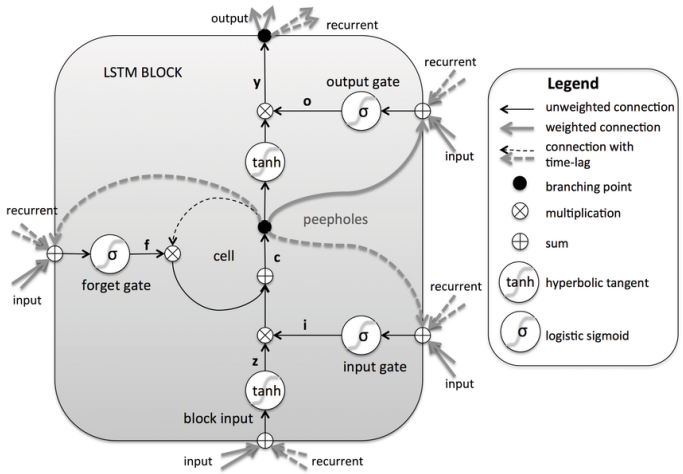
- 1943 : Formal neuron [McCuloch-Pitts]
- 1950 : Organization of neurons and learning rules [Hebb]
- 1960 : Perceptron [Rosenblatt]
- 1960 : Update rule [Widrow Hoff]
- 1969 : Limitations of the Perceptron [Minsky]
- 1980s : Back-propagation [Rumelhart and Hinton]
- 1990s : Convolutional Networks [LeCun and al.]
- 1990s: Long Short Term Memory networks [Hochreiter and Schmidhuber]
- 2006 : Paper on Deep Learning in Nature [Hinton and al.]
- 2012 : Imagenet Challenge Win [Krizhevsky, Sutskever, and Hinton]
- 2013 : First edition of ICLR
- 2013 : Memory networks [Weston and al.]
- 2014 : Adversarial Networks [Goodfelow and al.]
- 2014 : Google Net [Szegedy and al.]
- 2015 : Residual Networks [He et al.]



History

Key dates

- 1943 : Formal neuron [McCuloch-Pitts]
- 1950 : Organization of neurons and learning rules [Hebb]
- 1960 : Perceptron [Rosenblatt]
- 1960 : Update rule [Widrow Hoff]
- 1969 : Limitations of the Perceptron [Minsky]
- 1980s : Back-propagation [Rumelhart and Hinton]
- 1990s : Convolutional Networks [LeCun and al.]
- 1990s: Long Short Term Memory networks [Hochreiter and Schmidhuber]
- 2006 : Paper on Deep Learning in Nature [Hinton and al.]
- 2012 : Imagenet Challenge Win [Krizhevsky, Sutskever, and Hinton]
- 2013 : First edition of ICLR
- 2013 : Memory networks [Weston and al.]
- 2014 : Adversarial Networks [Goodfelow and al.]
- 2014 : Google Net [Szegedy and al.]
- 2015 : Residual Networks [He et al.]



History

Key dates

- 1943 : Formal neuron [McCuloch-Pitts]
- 1950 : Organization of neurons and learning rules [Hebb]
- 1960 : Perceptron [Rosenblatt]
- 1960 : Update rule [Widrow Hoff]
- 1969 : Limitations of the Perceptron [Minsky]
- 1980s : Back-propagation [Rumelhart and Hinton]
- 1990s : Convolutional Networks [LeCun and al.]
- 1990s: Long Short Term Memory networks [Hochreiter and Schmidhuber]
- 2006 : Paper on Deep Learning in Nature [Hinton and al.]
- 2012 : Imagenet Challenge Win [Krizhevsky, Sutskever, and Hinton]
- 2013 : First edition of ICLR
- 2013 : Memory networks [Weston and al.]
- 2014 : Adversarial Networks [Goodfelow and al.]
- 2014 : Google Net [Szegedy and al.]
- 2015 : Residual Networks [He et al.]

Deep Learning today

Spectaculary breakthroughs - fast industrial transfer

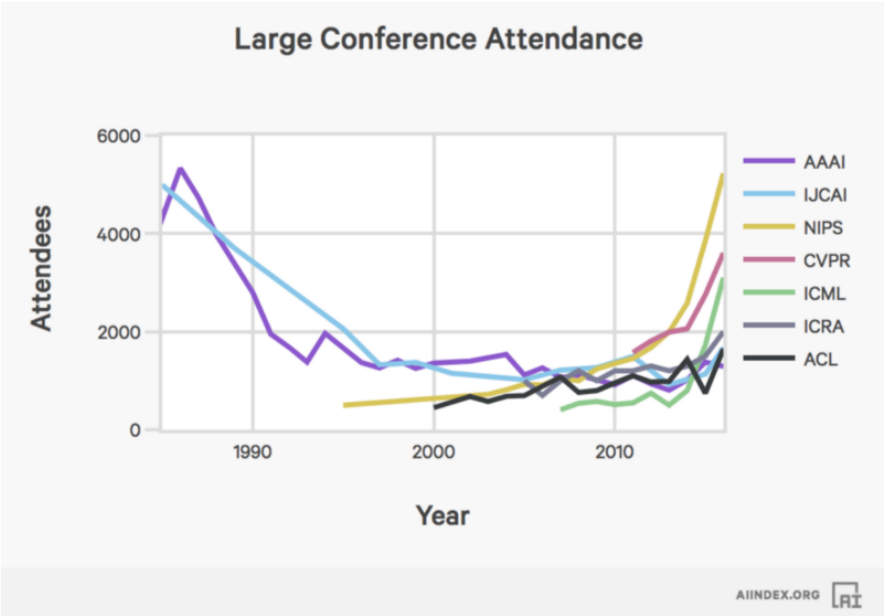
- Images, Videos, Audio, Speech, Texts
- Successful setting
  - Structured data (temporal, spatial...)
  - Huge volumes of datas
  - Huge models (millions of parameters)
  - Huge storage and computing resources (GPU, TPU)

	VGGNet	DeepVideo	GNMT
Used For	Identifying Image Category	Identifying Video Category	Translation
Input	Image 	Video 	English Text 
Output	1000 Categories	47 Categories	French Text
Parameters	140M	~100M	380M
Data Size	1.2M Images with assigned Category	1.1M Videos with assigned Category	6M Sentence Pairs, 340M Words
Dataset	ILSVRC-2012	Sports-1M	WMT'14

Machine Learning and Deep Learning today

Spectacular diffusion and activity

- Machine Learning and Deep Learning Conferences sold out early
- More attendees than ever seen in computer science conferences
- Exponential growth
- Semantic change in what AI means



DL research is going very fast !!

Example of an emerging topic: Generative Adversarial Networks

- First publication : 2014 by Ian J. Goodfellow, and al.
- Hundreds of publications (close to a thousand) papers since

New publication mode

- Wasserstein GANs, Martin Arjovsky and al.
  - Published on arXiv : Jan 2017
  - Published at ICML in Aout 2017
- Improved Training of Wasserstein GANs by Ishaan Gulrajani and al.
  - Published on arXiv : March 2017
  - Published at NIPS in December 2017
- Improving the Improved Training of Wasserstein GANs: A Consistency Term and Its Dual Effect by Xiang Wei and al.
  - Published on Openreview : Oct 2017
  - Accepted as poster at ICLR in 2018 (April 2018)

Few examples

Various applications

- Image Segmentation `https://youtu.be/VOC3huqHrss`
- Image Transformation `https://www.facebook.com/verge/videos/1661231000579903/`
- Language models : `https://pageperso.lis-lab.fr/benoit.favre/gulliver/index.html`
- ...

Outline

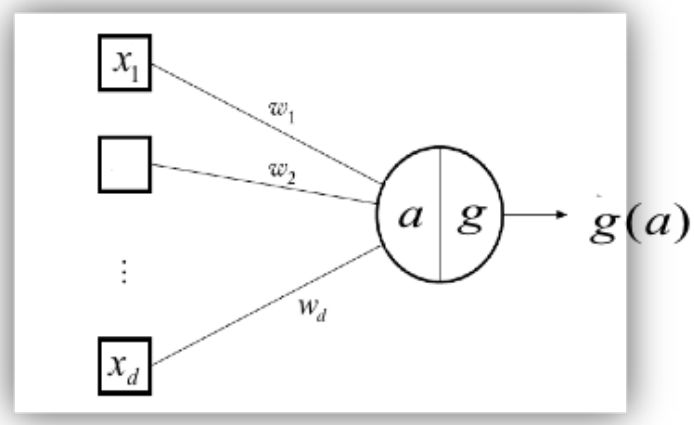
A single Neuron (biological inspiration [McCulloch and Pitts, 1943])

One Neuron

- Elementary computation

$$\text{activation} = w^T.x = \sum_j w_j x_j + w_0$$

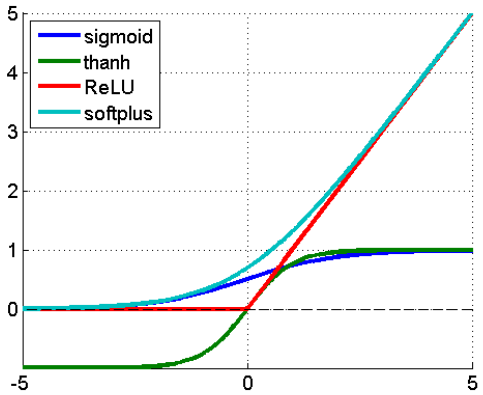
$$\text{output} = g(a(x))$$



Non linearity : g

- Sigmoide, Hyperbolic tangent, Gaussian
- Rectified Linear Unit (RELU)

$$f(x) = 0 \text{ if } x \leq 0$$
$$= x \text{ otherwise}$$



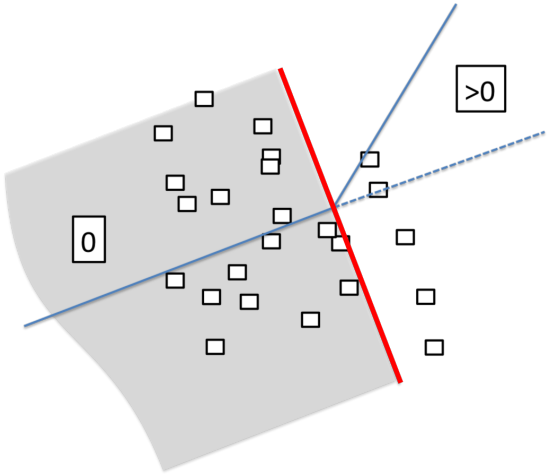
A single Neuron (biological inspiration [McCulloch and Pitts, 1943])

One Neuron

- Elementary computation

$$\text{activation} = w^T.x = \sum_j w_j x_j + w_0$$

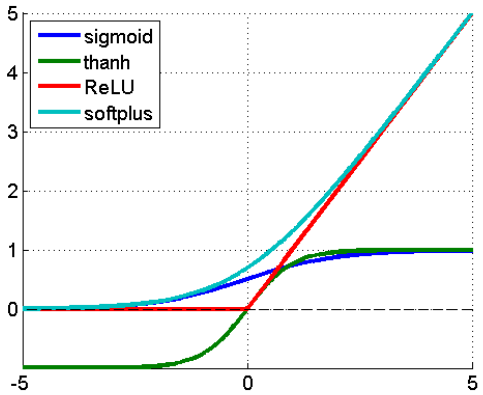
$$\text{output} = g(a(x))$$



Non linearity : g

- Sigmoide, Hyperbolic tangent, Gaussian
- Rectified Linear Unit (RELU)

$$f(x) = 0 \text{ if } x \leq 0$$
$$= x \text{ otherwise}$$



Multi Layer Perceptron (MLP)

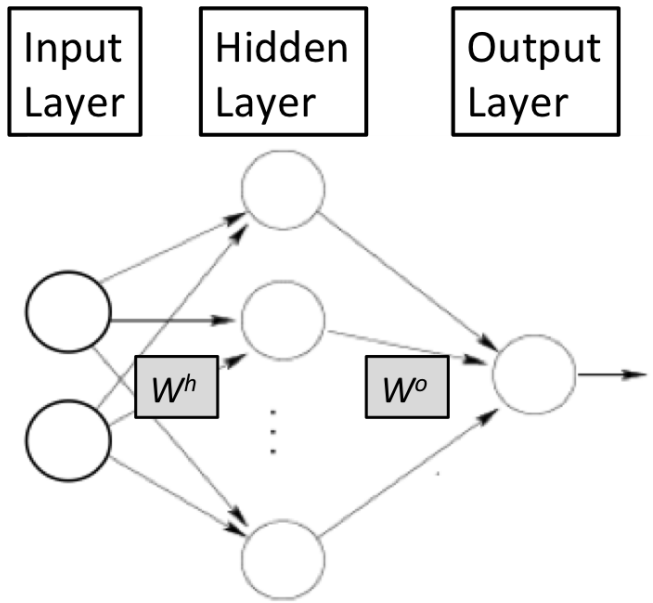
Structure

- Organization in successive layers
  - Input layer
  - Hidden layers
  - Output layer

Function implemented by a MLP

$$g(W^o.g(W^h x))$$

- Inference: Forward propagation from input to output layer
- Learning : Backpropagation algorithm (Stochastic Gradient Descent)



MLPs are Universal approximators

One layer is enough !

- Theorem [Cybenko 1989]: Let $\phi(\cdot)$ be a nonconstant, bounded, and monotonically-increasing continuous function. Let I_m denote the m-dimensional unit hypercube $[0,1]^m$. The space of continuous functions on I_m is denoted by $C(I_m)$. Then, given any $\epsilon > 0$, there exists an integer N, such that for any function $f \in C(I_m)$, there exist real constants $v_i, b_i \in \mathbb{R}$ and real vectors $w_i \in \mathbb{R}^m$, where $i = 1, \cdots, N$, such that we may define:

$$F(x) = \sum_{i=1}^{N} v_i \phi\left(w_i^T x + b_i\right)$$

  as an approximate realization of the function f where f is independent of ϕ ; that is : $|F(x) - f(x)| < \epsilon$ for all $x \in I_m$. In other words, functions of the form $F(x)$ are dense in $C(I_m)$.

- Existence theorem only
- Many reasons for not getting good results in practice

Outline

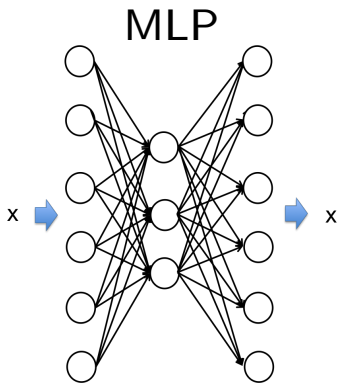
Autoencoders

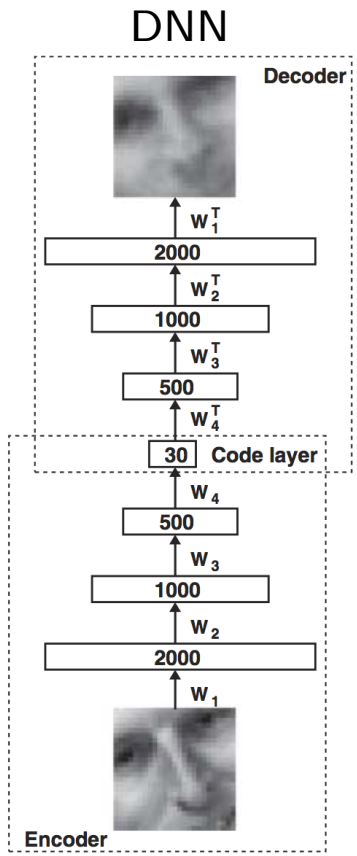
NN with Diabolo shape

- Reconstruct the input at the output via an intermediate (small) layer
- Unsupervised learning
- Non linear projection, distributed representation (close to PCA)

Deep NN with Diabolo shape

- Pioneer work that started the Deep Learning wave ([Hinton et al., Nature 2006])

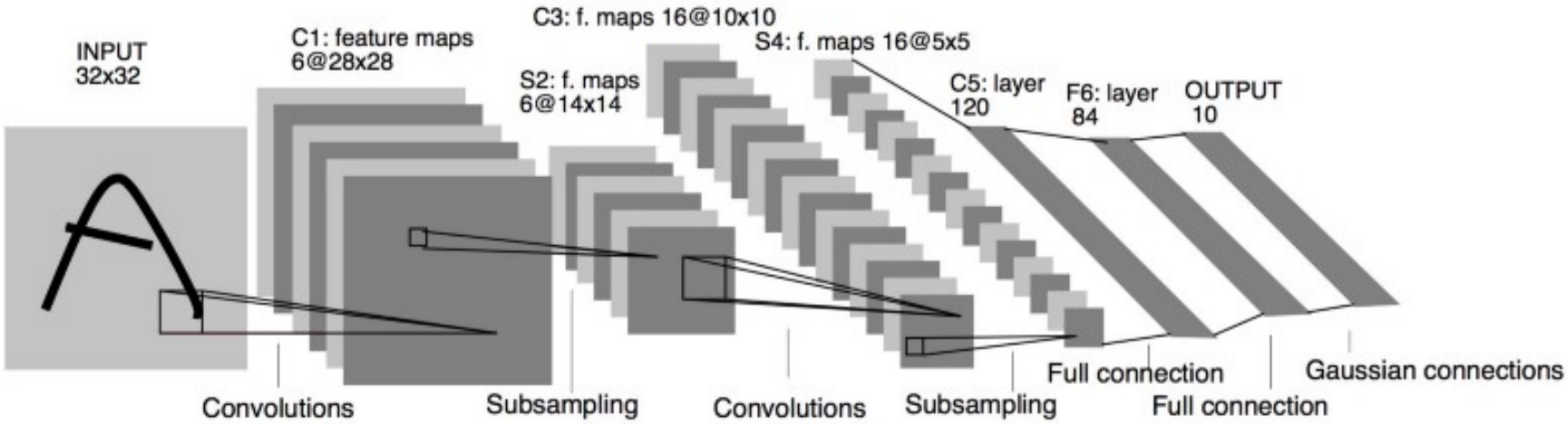




Convolutional models (CNNs)

LeNet architecture [LeCun 1997]

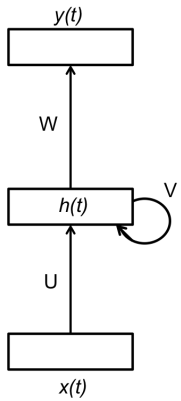
- Sequence of (convolutional + pooling) layers followed by dense layers



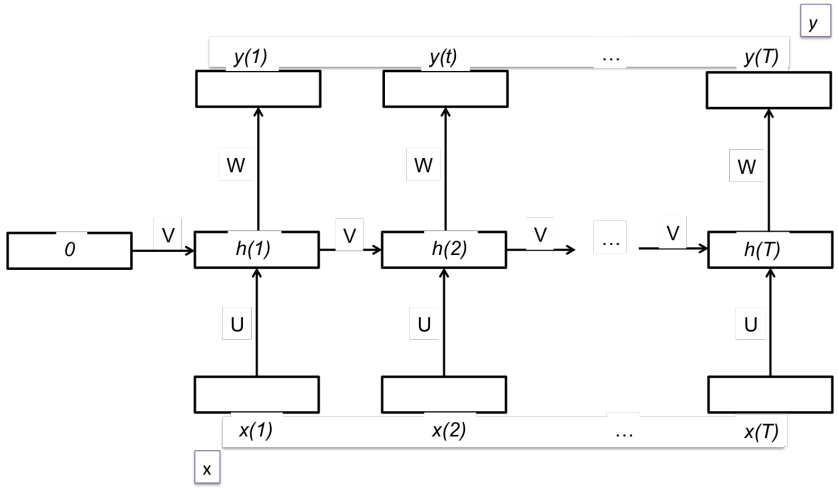
Recurrent NNs

RNNs in general

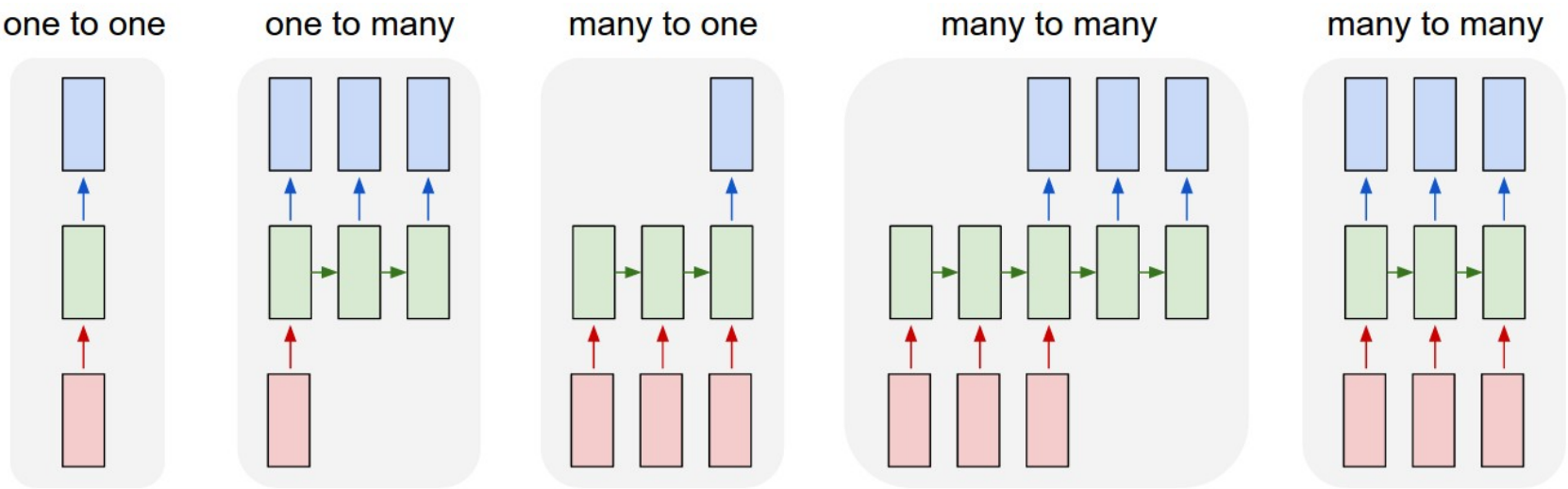
- RNN = NN with cycles in its connections
- Much more powerful than acyclic models (MLPs)
- Few RNNs architectures work well (e.g. Elman architceture)
- This model computes an output sequence from an input sequence



RNN



Various settings

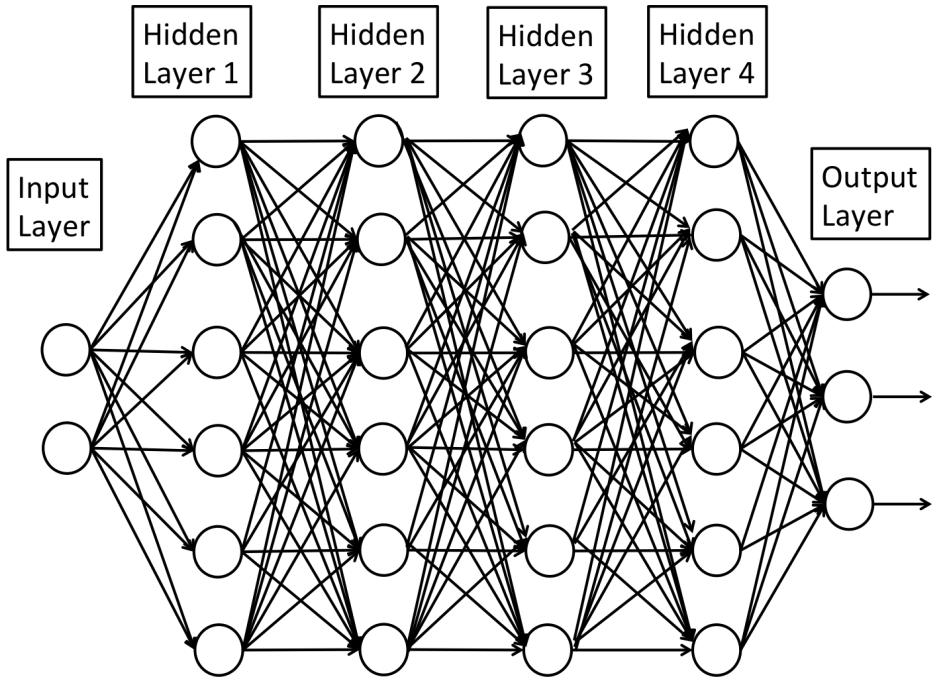


- One to One : MLP, CNN ...
- One to Many : Generation of a sequential process (speech, handwriting ...)
- Many to one : Sequence classification (e.g. activity recognition)
- Asynchronous Many to many : Machine Translation
- Synchronous Many to Many : POS tagging, Speech recognition...

Deep Learning = Representation Learning

DNNs = NNs with more than one hidden layer !

A series of hidden layers

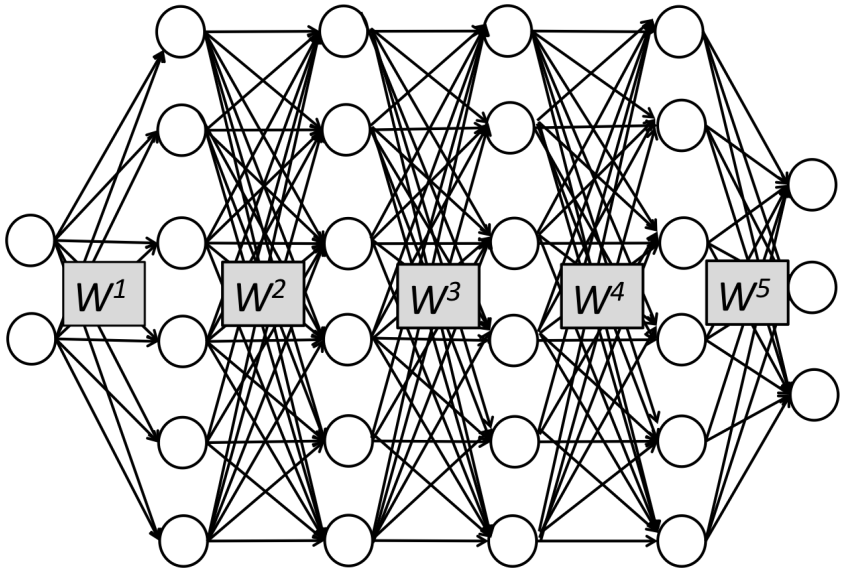


Deep Learning = Representation Learning

DNNs = NNs with more than one hidden layer !

Computes a complex function of the input

$$y = g(W^k \times g(W^{k-1} \times g(...g(W^1 \times x))))$$

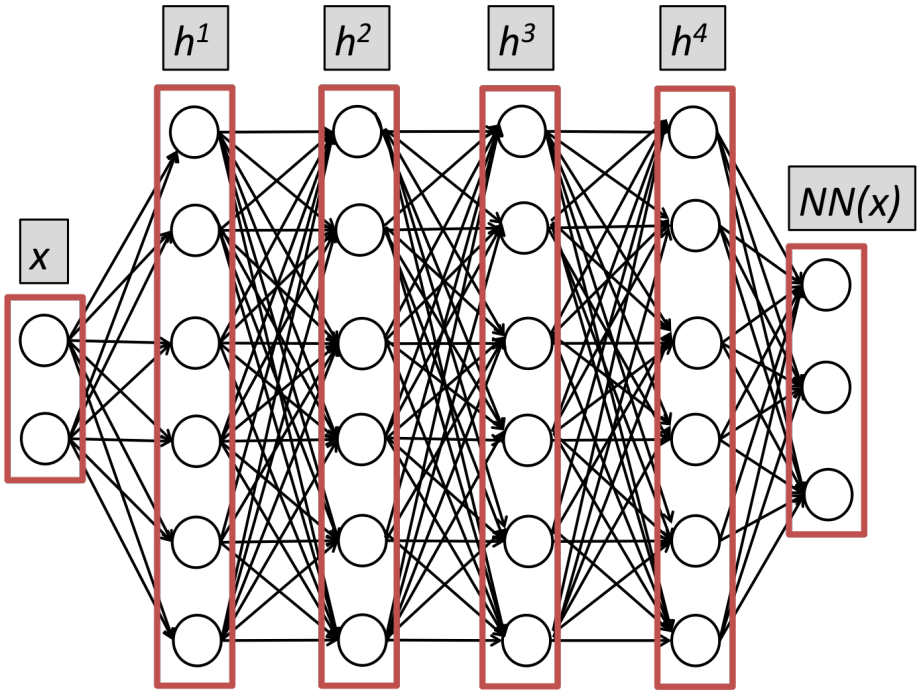


Deep Learning = Representation Learning

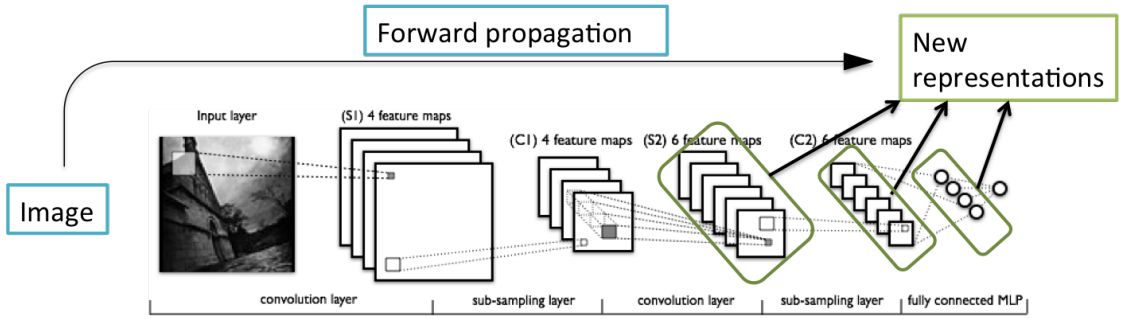
DNNs = NNs with more than one hidden layer !

Computes new representations of the input

$$h^i(x) = g(W^i \times h^{i-1}(x))$$



Towards "universal representation": images



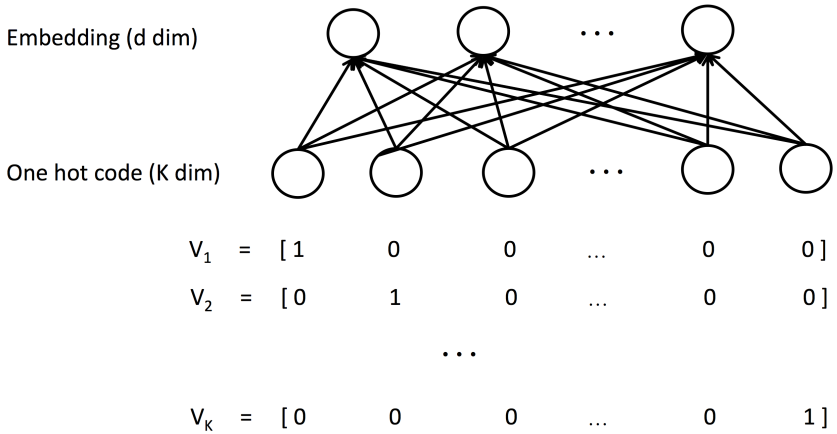
Reusability

- Many very deep architectures have been proposed by major actors (Google, Microsoft, Facebook...)
- Kind of universal representation of images: better to use these models' high features
  - With fine tuning (of upper layers) if enough training data are available on the target task
  - As a preprocessing if not

Towards "universal representation": embeddings of words

Embedding layer: Transformation layer for discrete/categorical inputs

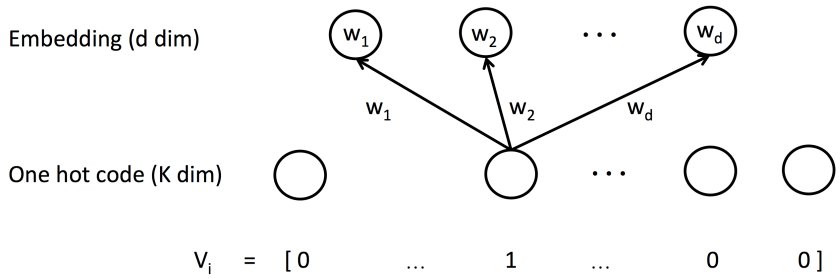
- Example : a Word in a Dictionary (Natural Language Processing tasks)
- Embedding = distributed representation. Not a new idea (LSA, LDA)



Towards "universal representation": embeddings of words

Embedding layer: Transformation layer for discrete/categorical inputs

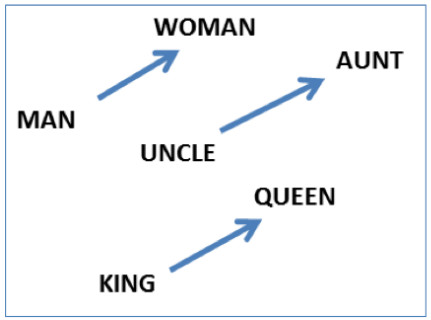
- Example : a Word in a Dictionary (Natural Language Processing tasks)
- Embedding = distributed representation. Not a new idea (LSA, LDA)

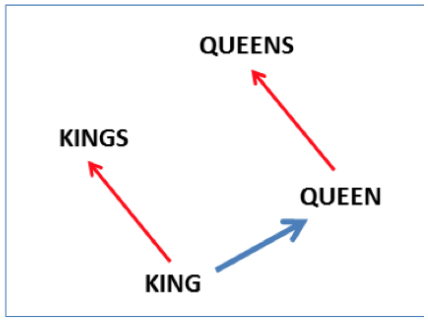


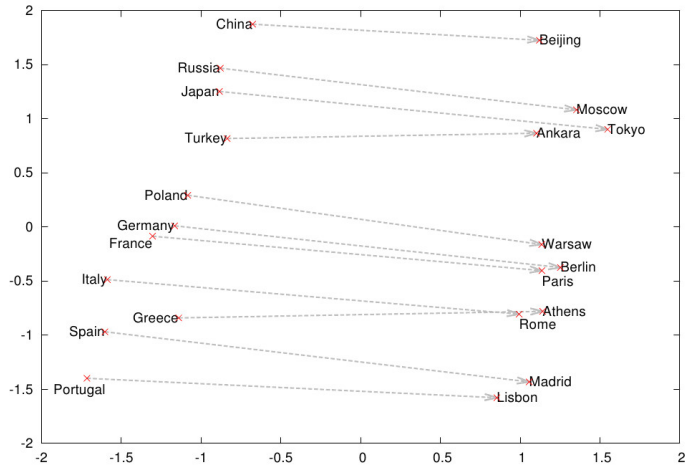
A particular interesting effect

Compositionality

- $Emb('King') + Emb('Woman') - Emb('Man') \approx Emb('Queen')$

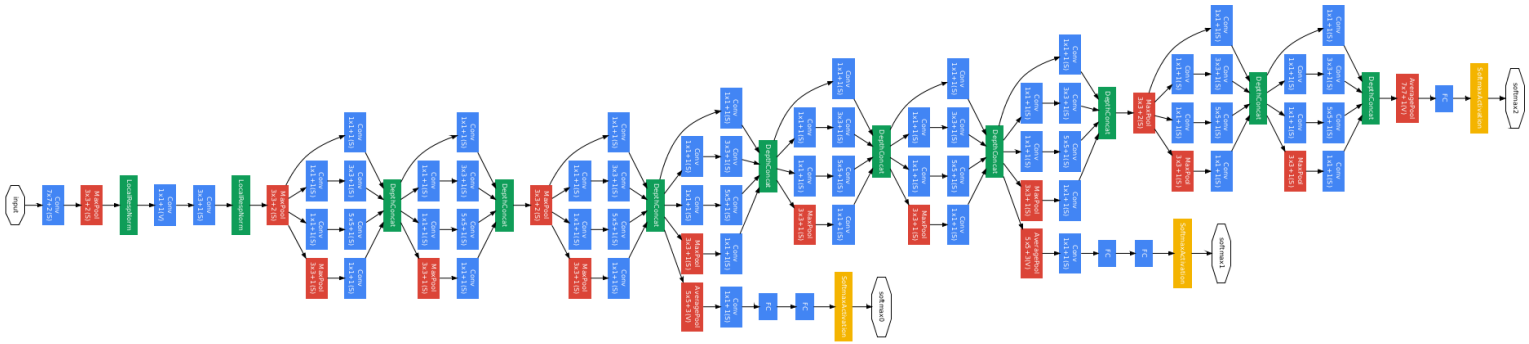






Looking for a good architecture: Lego game

How to reach such an architecture (GoogleNet 2014) ?

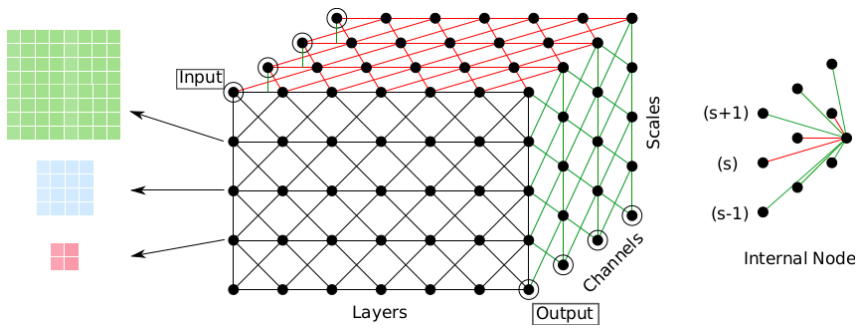


Searching for a good architecture requires making choices !!

Looking for a good architceture

Illustration [Verbeek 2017]

- **Simple** search (but for a large network)
  - 19 convolution layers and 5 pooling layers to set
  - Question: where to put the poooling layers? $\rightarrow$ 40 000 architectures !!
  - No question about layers' dimensions, activation function, kernels' size, pooling type etc
- Remember
  - 1 hour GPU on AWS = 1 $
  - Learning 1 model = Few hours
    $\Rightarrow$ Expensive design !!!
- Not much alternatives



Looking for a good architecture: use others' !!

Deep Models for High resolution images [Radford 2015]

Historical attempts to scale up GANs using CNNs to model images have been unsuccessful. This motivated the authors of LAPGAN (Denton et al., 2015) to develop an alternative approach to iteratively upscale low resolution generated images which can be modeled more reliably. We also encountered difficulties attempting to scale GANs using CNN architectures commonly used in the supervised literature. However, after extensive model exploration we identified a family of architectures that resulted in stable training across a range of datasets and allowed for training higher resolution and deeper generative models.

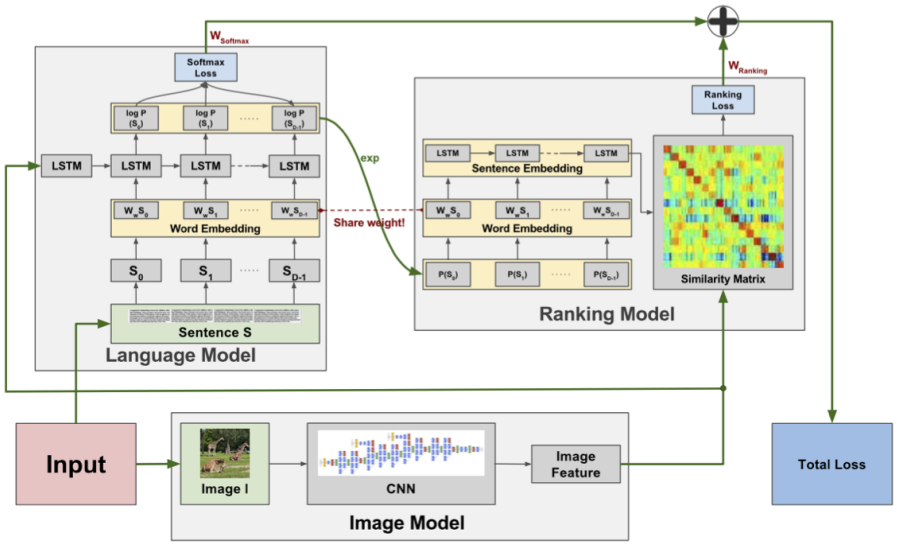
What are DNNs at the end?

Not more than MLPs

- Feed forward propagation enabling chain rule (backpropagation)
- Stochastic Gradient Descent Optimization
- Most ideas were there in the 90's

Yet something different...

- Much more complex architectures
- End to end learning

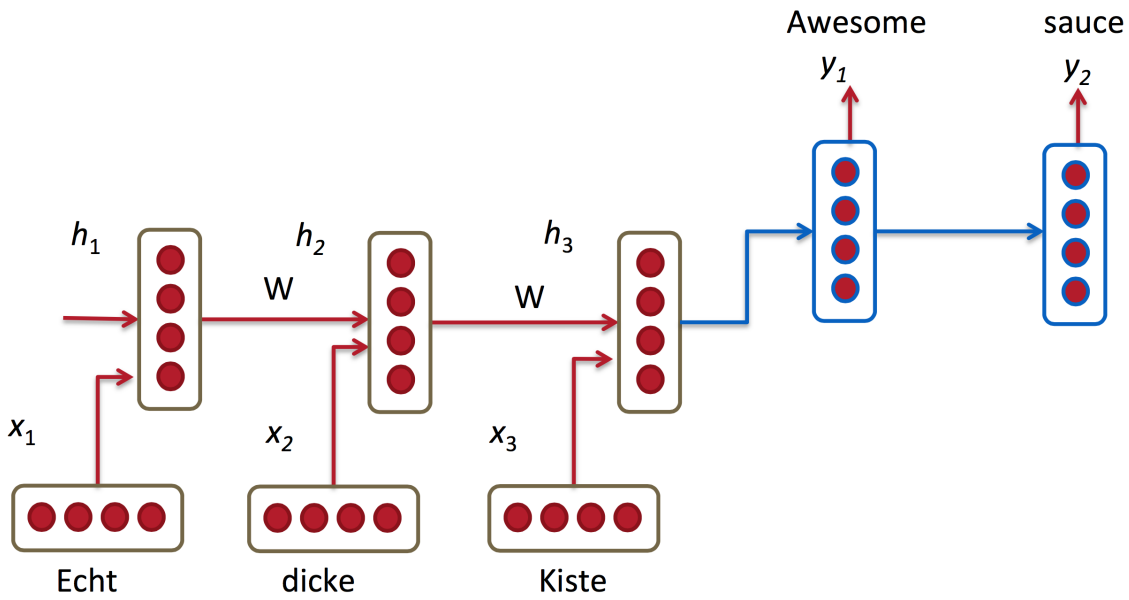


(Yeung et al., 2015)

Modules that encode structured data

Building brick for designing complex models

- Computing a low dimensional, dense representation of structured data: images (with CNNs), sequences (speech, text, video) with RNNs, trees (parse tree in NLP) with Recursive NNs
- At the basis of many complex systems

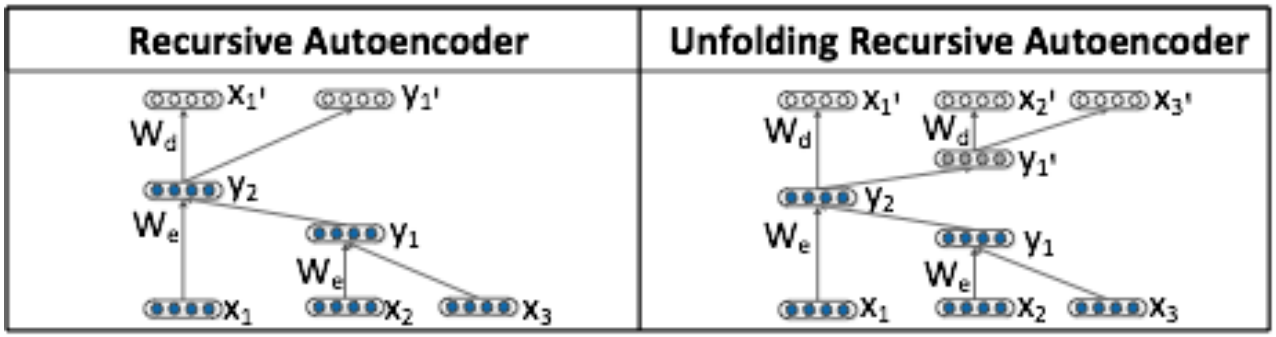


Machine translation with sequence to sequence models

Modules that encode structured data

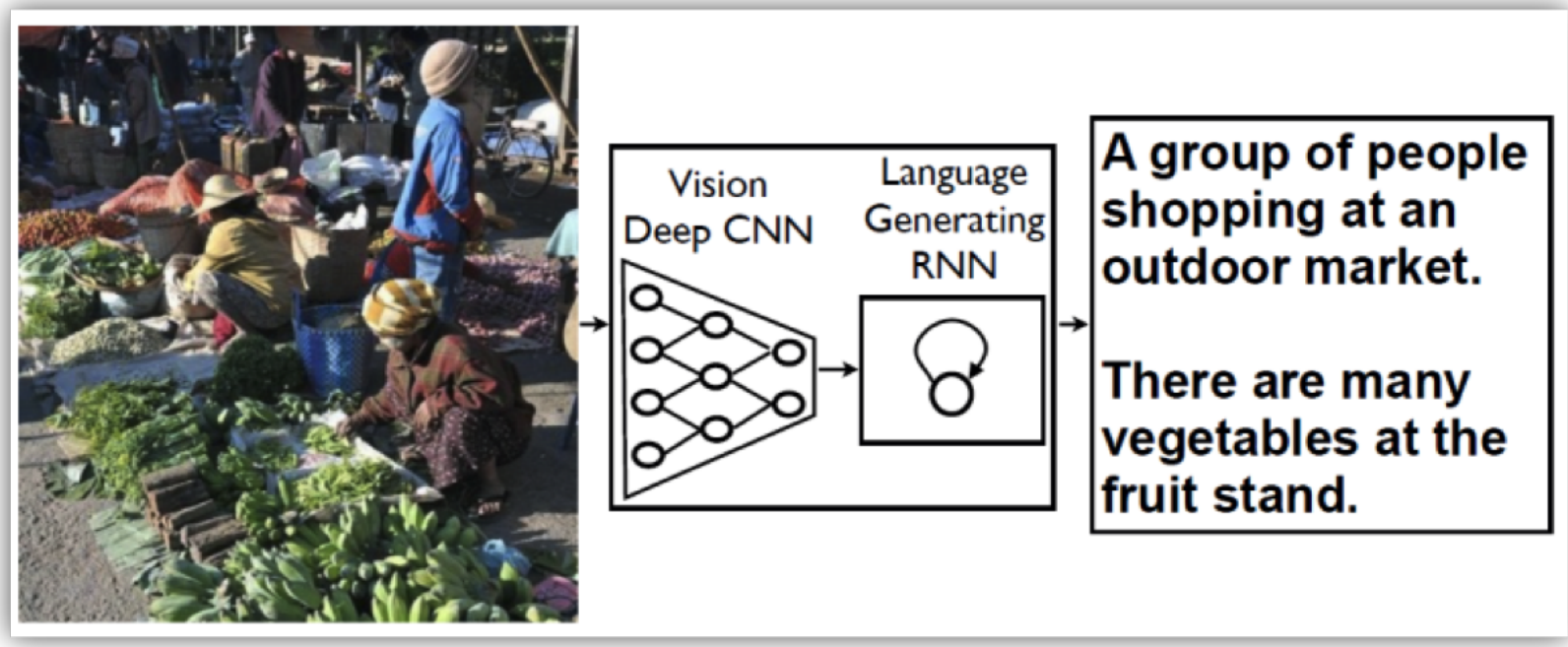
Building brick for designing complex models

- Computing a low dimensional, dense representation of structured data: images (with CNNs), sequences (speech, text, video) with RNNs, trees (parse tree in NLP) with Recursive NNs
- At the basis of many complex systems



Sentence representation with recursive NNs [Socher et al., 2013]

Using bricks to build complex systems



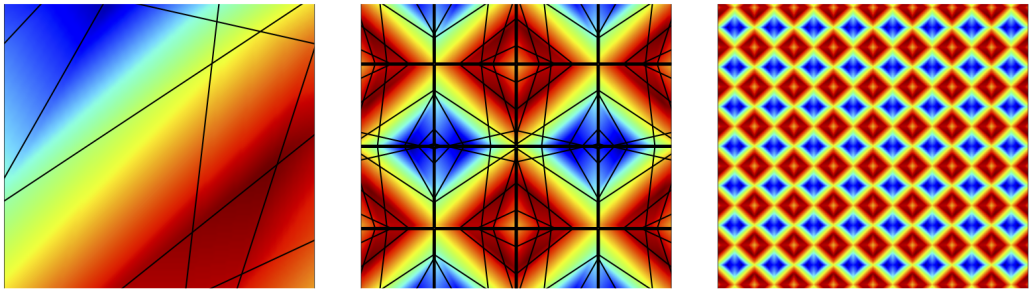
Automatic captioning [Honglak et al., 2014]]

Outline

Deep vs Shalow: Increased capacity?

Characterizing the complexity of functions a DNN may implement [Pascanu and al., 2014]

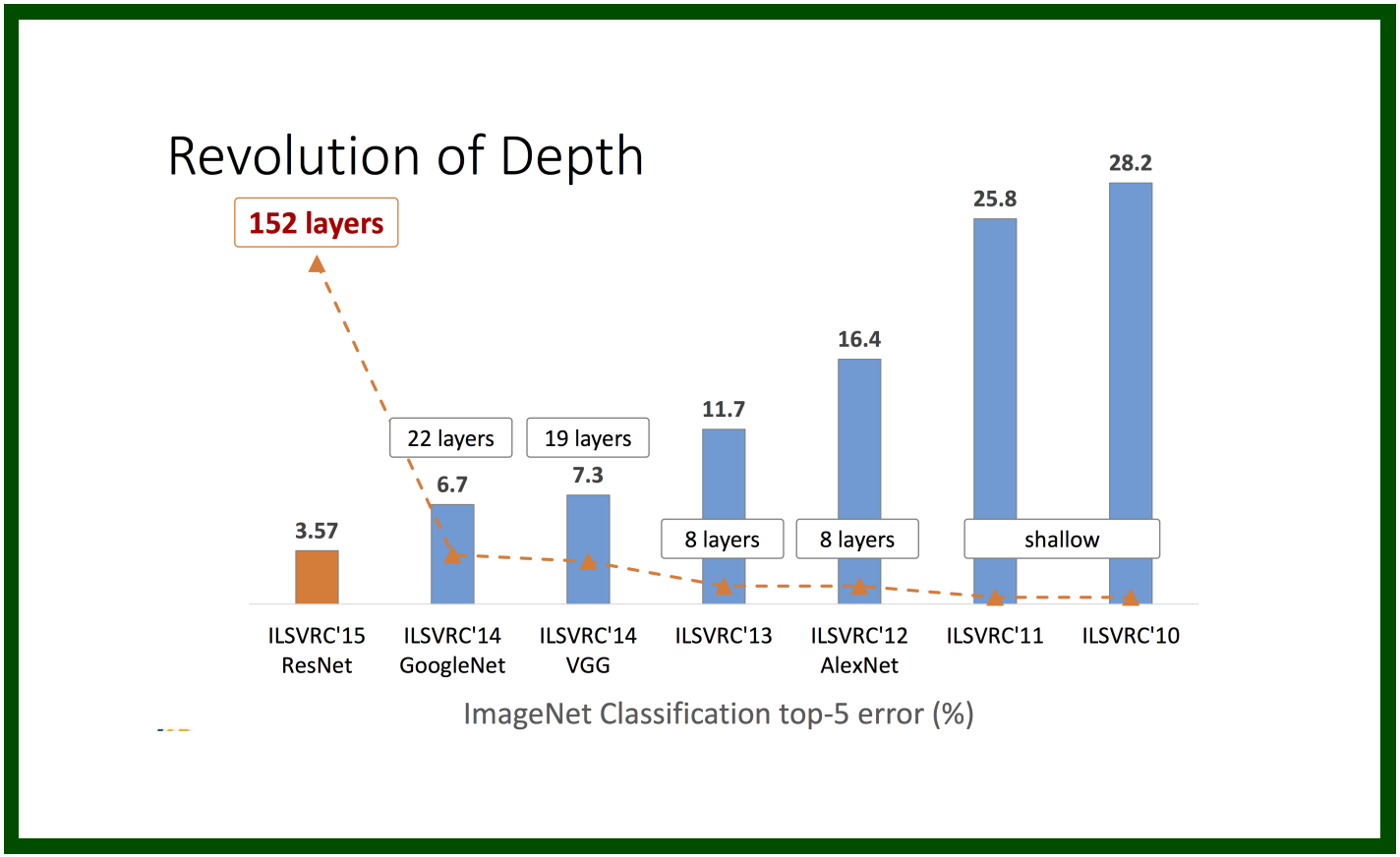
- DNNs with RELU activation function $\Rightarrow$ piecewise linear function
- Complexity of DNN function as the Number of linear regions on the input data
- Exponentially more regions per parameter in terms of number of HL



From [Pascanu and al., 2014]

- Right: Heat map of a function computed by a 4 layer model with a total of 24 hidden units. It takes at least 137 hidden units on a shallow model to represent the same function.

The Times They Are A Changing



(slide from [Kaiming He])

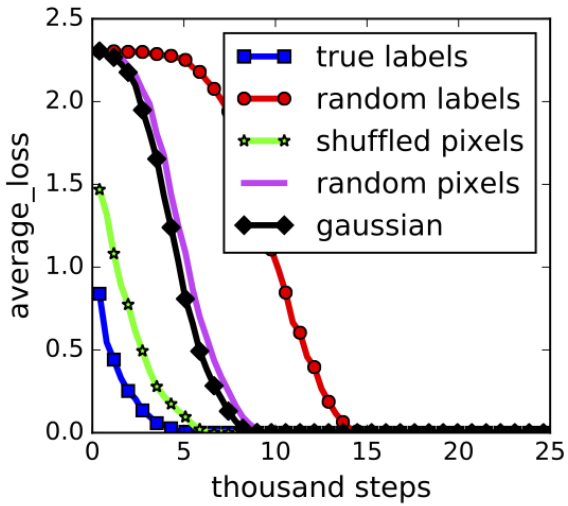
DL in the view of generalization, overtraining, local minimas etc

Traditional Machine Learning

- Overfiting is the enemy
- One may control generalization with appropriate regularization

Recent results in DL

- The Overfit idea should be revised for DL [Zhand and al., 2017]
  - Deep NN may learn noise !
  - Regularization may slightly improve performance but is not THE answer for improving generalization

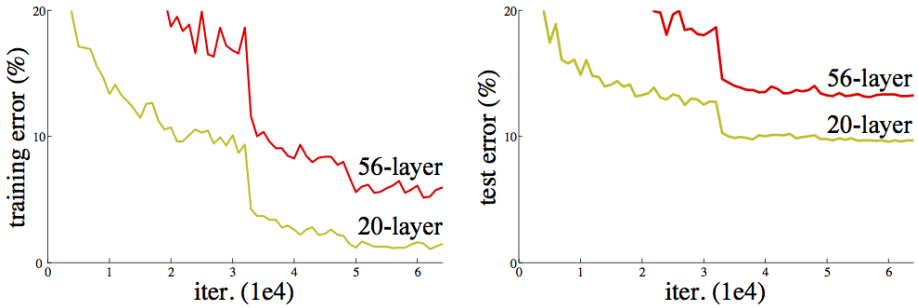


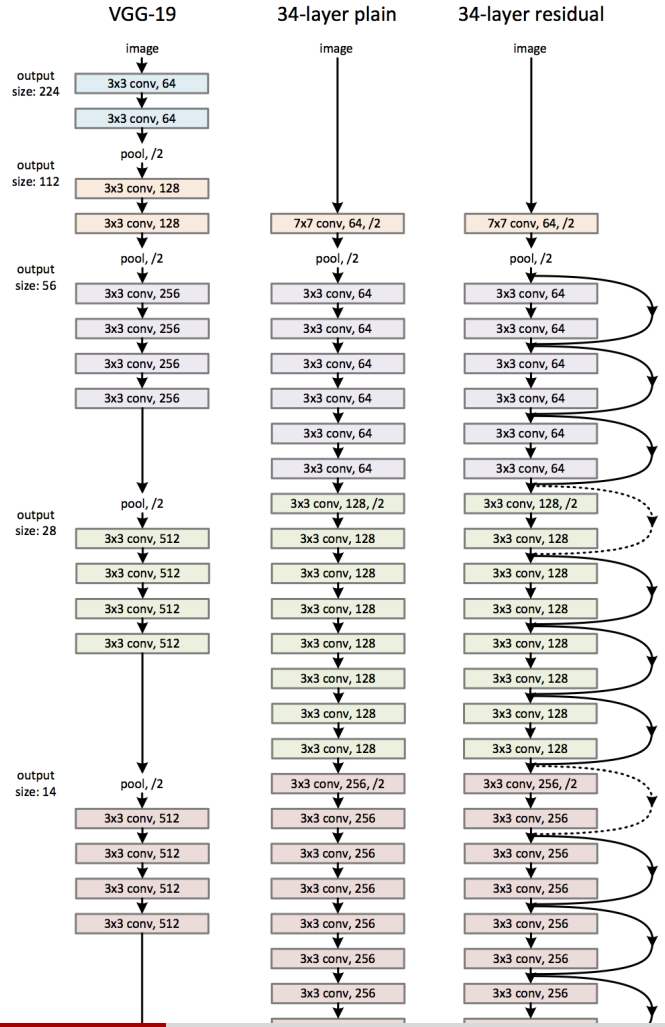
[Zhand and al., 2017]

Depth is not enough

Simply stacking layers does not work (CIFAR results) ! (figures form [He and al., 2015])

...

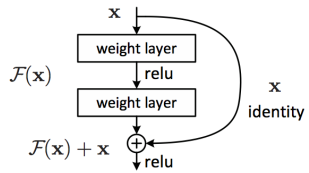




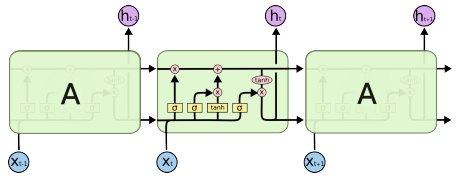
Depth is not enough

Making gradient flow for learning deep models (structural and optimization answers)

- Include the identity mapping as a possible path from the input to the output of a layer
  - ResNet building block [He and al., 2015]]



  - LSTM (deep in time) [Hochreichter and al., 1998]



  - Skip connections
- Dropout [Hinton et al., 2012], Batch normalization and local normalization [Ioffe et al., 2015]

Outline

Deep Learning future

Who knows?

- Close future
  - Unsupervised learning (GANs)
  - Few shot learning, one shot learning, meta-learning
  - With reinforcement learning (going further than Go playing)
  - Increasing models' capacity : Memory, attention mechanisms
- Making it practical for the real world
  - Explainabilty
  - Robustness to attacks
- Necessary next steps
  - Adaptation and reusing knowledge
  - Reasonning

Increasing capacity with additional external memory

Joe went to the kitchen. Fred went to the kitchen. Joe picked up the milk.
Joe travelled to the office. Joe left the milk. Joe went to the bathroom.
Where is the milk now? A: office
Where is Joe? A: bathroom
Where was Joe before the office? A: kitchen

Principle Memory Networks [Weston and al., 2015]

- Add memory that can be read and written to for prediction
- Ability to deal with complex question answering

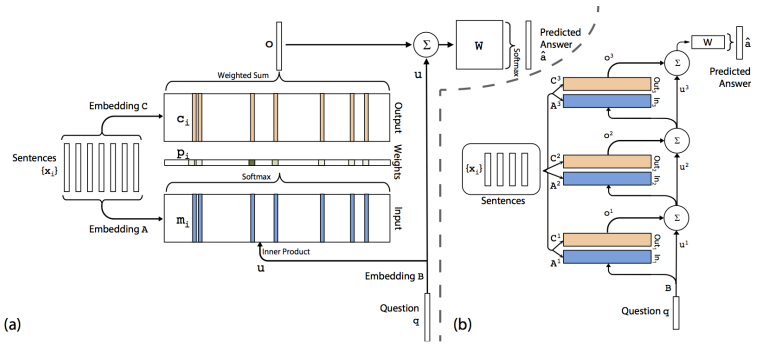


Figure 1: (a): A single layer version of our model. (b): A three layer version of our model. In practice, we can constrain several of the embedding matrices to be the same (see Section 2.2).

End to End Memory Networks [Sukhbaatar and al., 2015]

One step further: Reasoning

More complex reasonning tasks

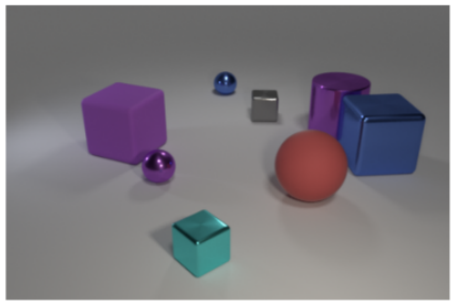


Figure 1: A sample image from the CLEVR dataset, with a question: "There is a purple cube behind a metal object left to a large ball; what material is it?"

[Hudson et al., 2018]

- Requires few steps of question answering like queries

Explainabilty

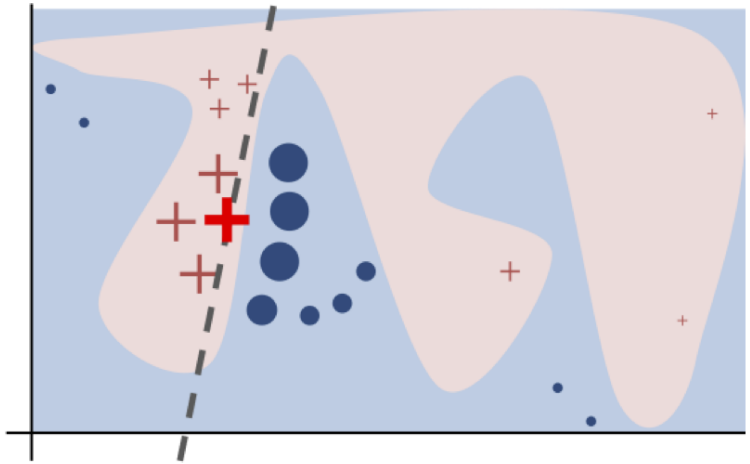
Black box models

- Machine Learning models are black box models
- Urgent needs for trustability with the increasing performance in real-life tasks (automatic cars, army, health etc)
- Few levels of understanability / explainability: Explain a decision on an input / the whole process
- Various kind of methods: Agnostic, model-based ...

Explaining a decision by sampling around an input [Ribeiro et al., KDD 2016]

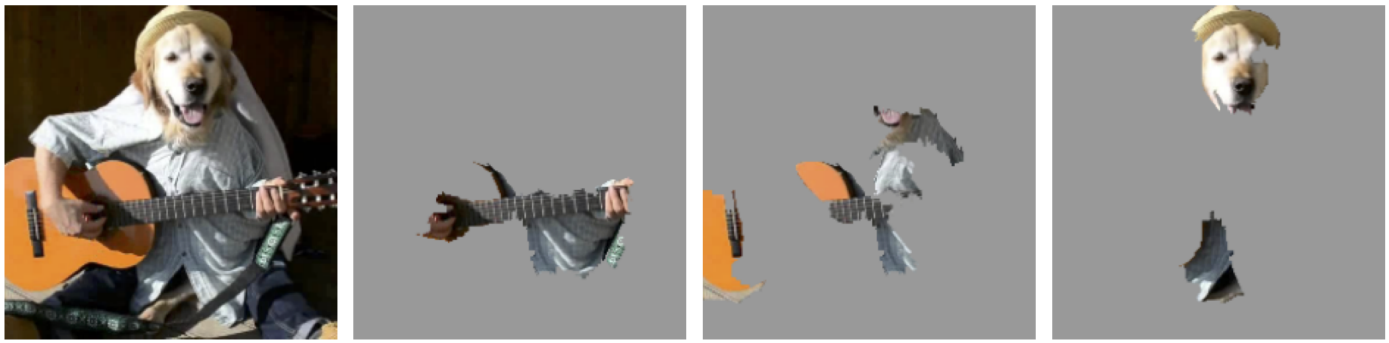
Main idea

- Decision boundary is loccaly linear (and one may then use standard techniques for linear models)



Method for explaining the decision on input x

- Sample points around a particular input x
- Fit a linear model



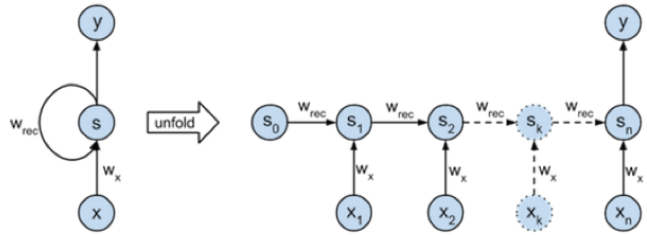
(a) Original Image (b) Explaining *Electric guitar* (c) Explaining *Acoustic guitar* (d) Explaining *Labrador*

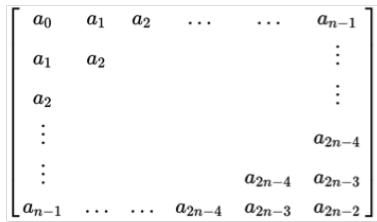
Explaining a model by approximating it

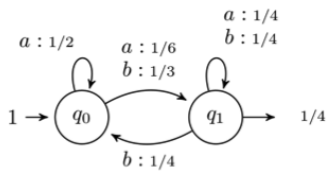
Main idea

- Distillation idea: Learn a simple model from a large one by using its outputs as targets

Example on RNNs [Goudian et al., 2018]







$$\alpha_0 = \begin{bmatrix}1\\0\end{bmatrix} \quad \alpha_\infty = \begin{bmatrix}0\\1/4\end{bmatrix}$$

$$M_a = \begin{bmatrix}1/2 & 1/6\\0 & 1/4\end{bmatrix} \quad M_b = \begin{bmatrix}0 & 1/3\\1/4 & 1/4\end{bmatrix}$$

Robustesse au piratage

Chatbots et apprentissage en ligne

- "A peine lancée, une intelligence artificielle de Microsoft dérape sur Twitter. L'entreprise américaine a lancé Tay, un chatbot censé discuter avec des adolescents sur les réseaux sociaux. Mais des propos racistes se sont glissés dans ces échanges." [Le Monde, Mars 2016]

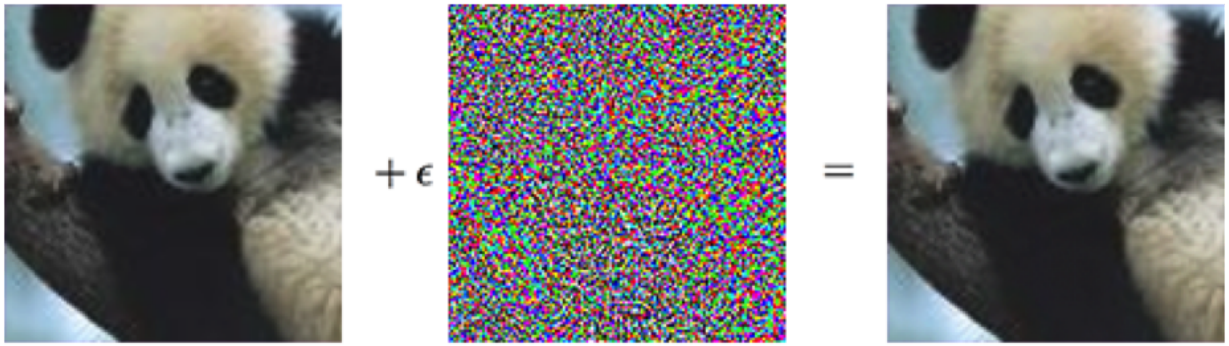
Biais de conception

- L'application aveugle de l'apprentissage-machine risque d'amplifier les biais qui sont présents dans les données [Tolga Bolukbasi, NIPS 2016].
- Outils de plongement lexical
  - L'homme est à la femme ce que le roi est à ... ? À la reine
  - L'homme est à la femme ce que le chirurgien est à . . . ? l'infirmière
  - ... ce que le programmeur informatique est à. . . ? la femme au foyer
  - . . . ce que l'architecte est à... ? la décoratrice
  - ... ce que le commerçant est à... ? la ménagère .

Adversarial examples [Goodfellow et al., 2015]

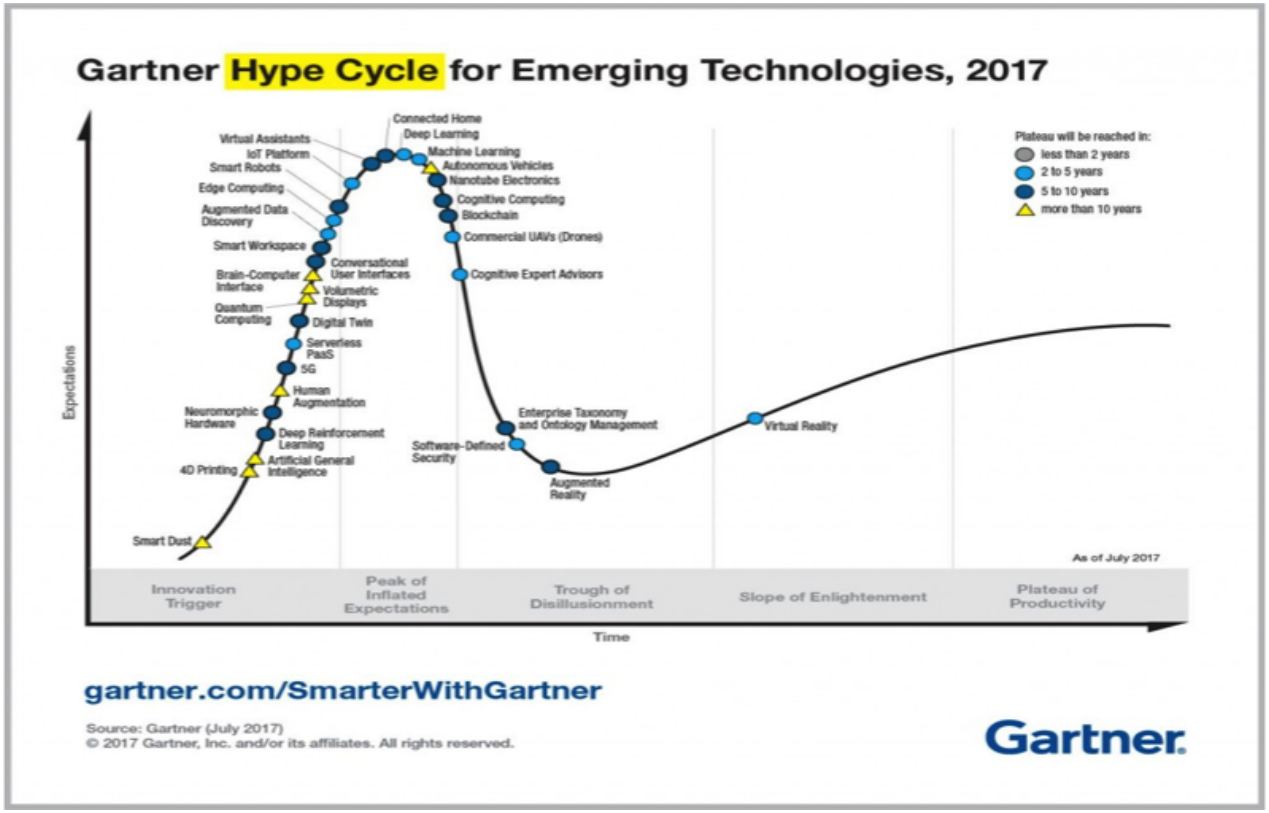
A ML/DL weakness ?

- Despite their high accuracy DNNs may be very weak, then untrustable for real life use
- Why ?
  - One may easily find adversarial noise that may fool the DNNs
  - There exist universal adversarial noise: working for any input sample
  - It may be very robust



What's next?

The next step: Generalized AI vs Narrow A



Outline

Conclusion

This does not happen twice in a researcher's life !

- Machine Learning and Deep Learning are everywhere, many opportunities for designing new systems, solving new tasks
- Huge spread of Machine Learning and Deep Learning ideas: Neurosciences, Security, Health, HEP...
- On the contrary ow what many people say and think
  - It is not so easy (except if you reuse the code (= the design and the learning) from someone else)
  - Many thinks to understand
  - Many things to invent