# 12. BEYOND CONVOLUTIONAL NEURAL NETWORKS

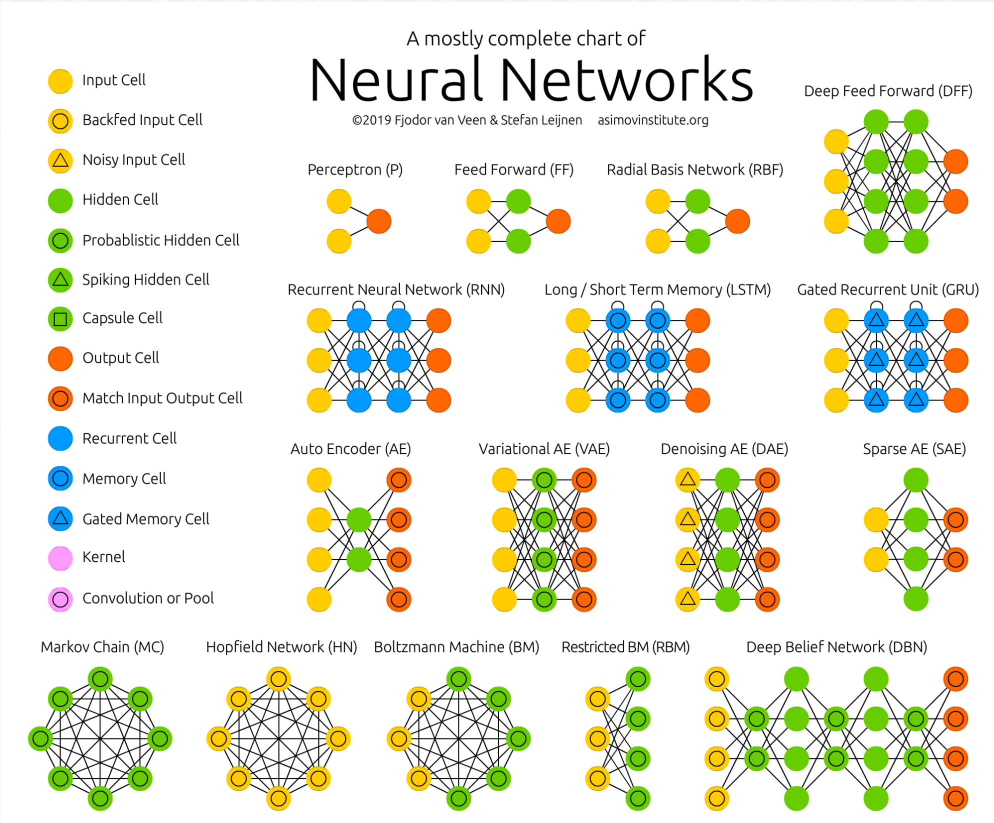
Stephan Robert-Nicoud
HEIG-VD/HES-SO

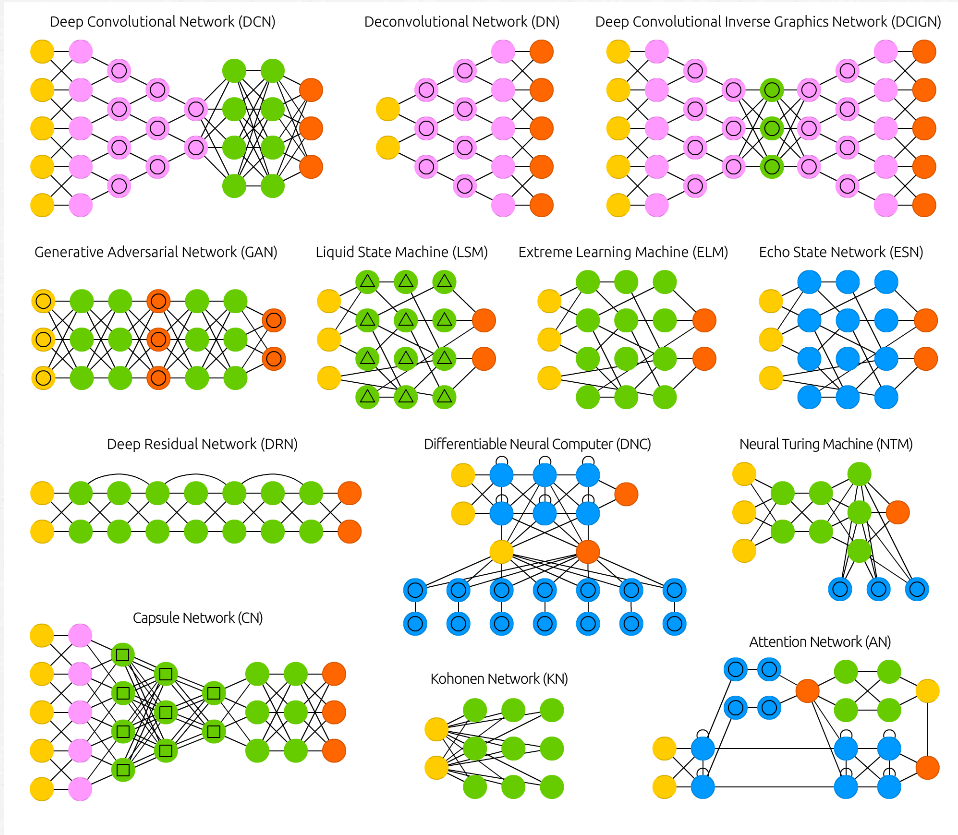
Credit: Andres Perez-Uribe

# Objectives

- Understand the principles behind encoder-decoder neural architectures
- Understand the capabilities of recurrent neural network architectures
- Recognize the sort of problems that can be treated with recurrent neural networks
- Analyze the motivations that have driven the development of novel architectures

# The Neural Networks zoo

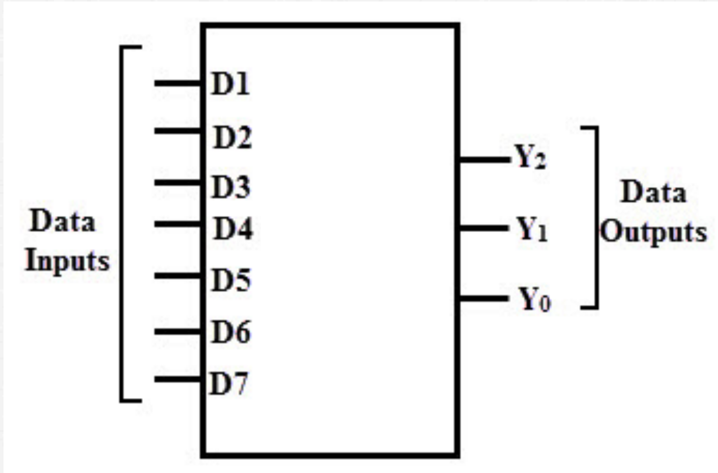



https://www.asimovinstitute.org/neural-network-zoo/

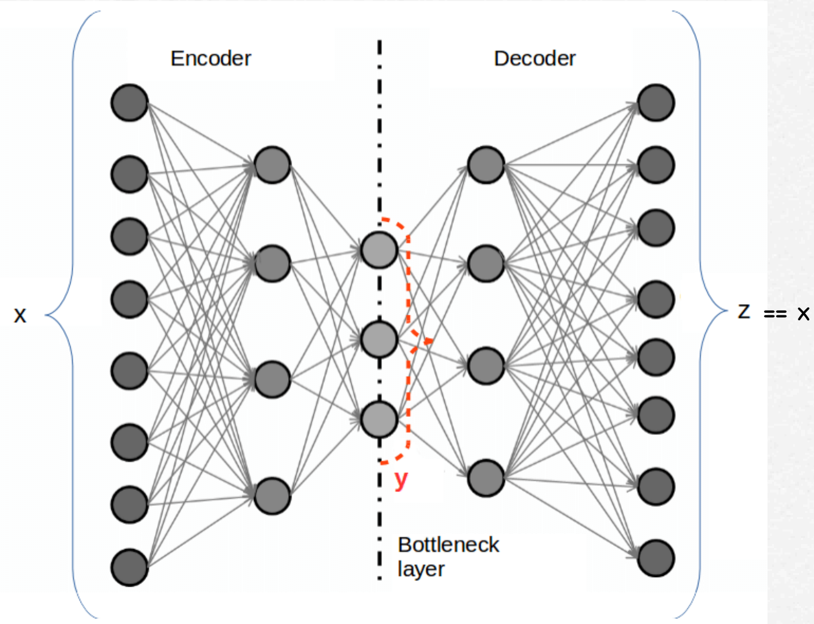
# “Combinational” neural networks

- Perceptron
- Multi-layer Perceptron (MLP)
- Radial Basis Function Network (RBF)*
- Convolutional Neural Networks (CNN)
- Deep Residual Networks (e.g., ResNet)

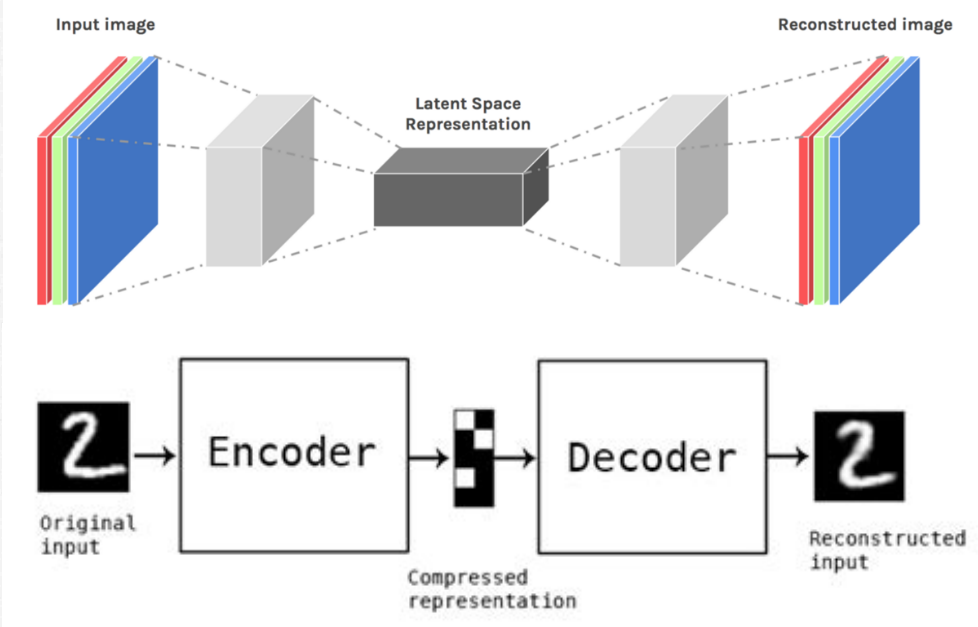



encoder

# Auto-encoders



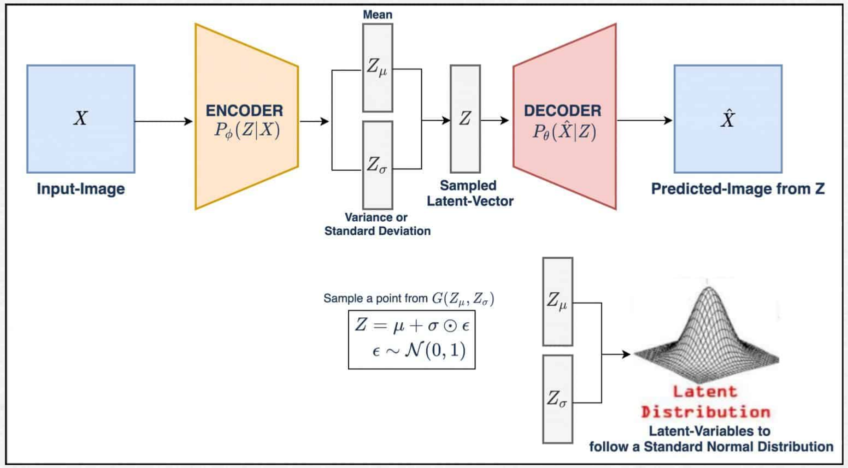

Fully-connected auto-encoder



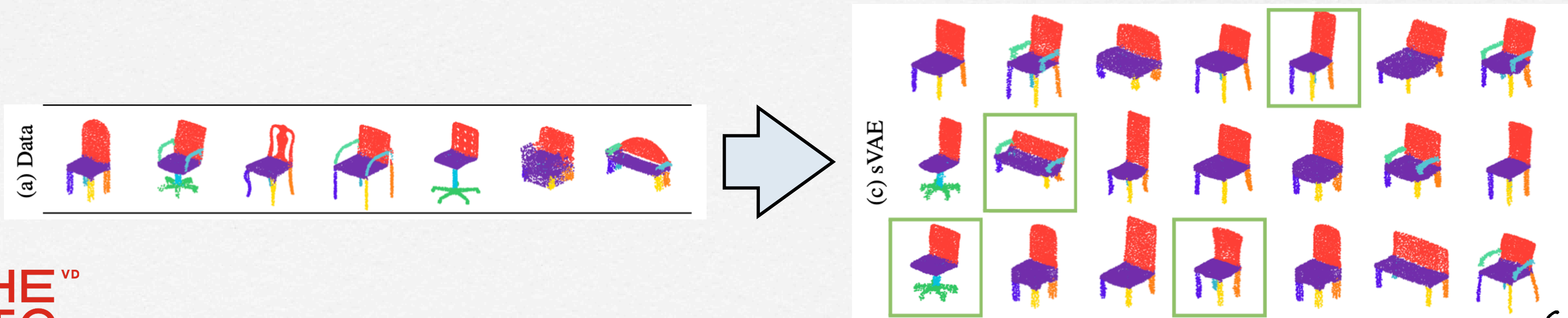

Deep auto-encoder

- An auto-encoder is trained to reproduce the input at the output from a reduced set of features (compressed information). It is used for denoising, compression, anomaly detection, characterization.

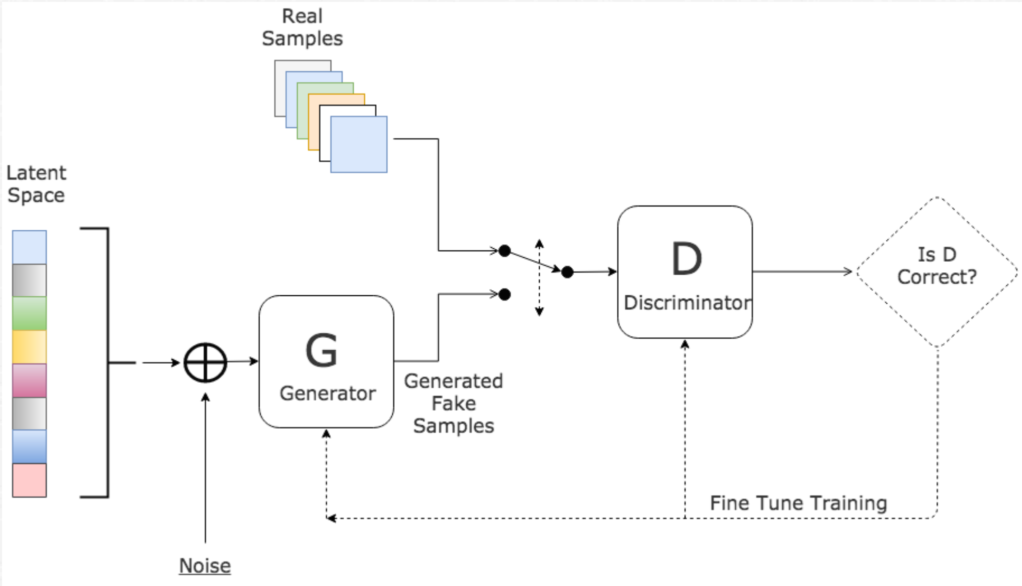
# Variational Auto-Encoders (VAE)




- We force the latent vectors to have a unit Gaussian distribution. Once trained, we can use the decoder part as a generator to create synthetic data, by adjusting the latent vectors:



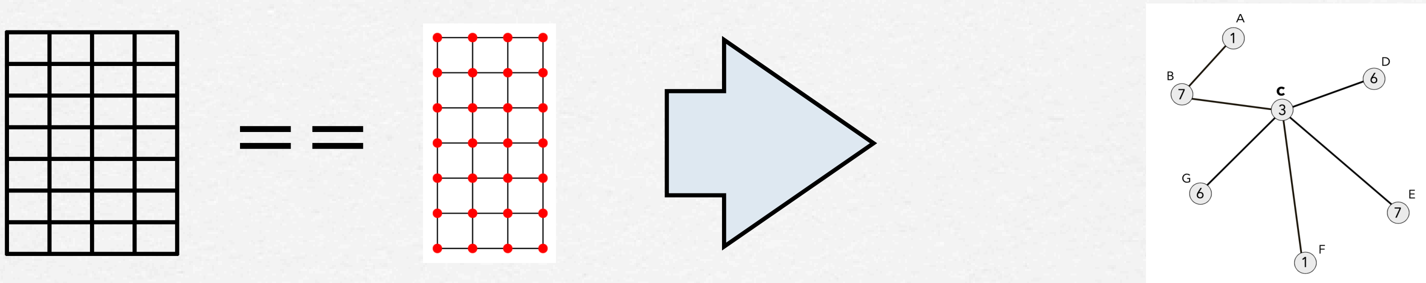
# Generative Adversarial Network (GAN)



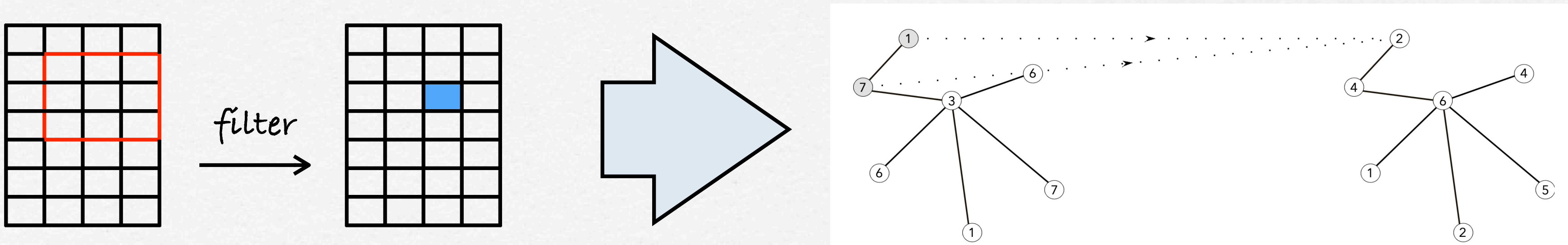

- We train two networks, G (decoder-type CNN) and D (encoder-type CNN) such that D gets better in classifying fake from real and G gets better in fooling D (e.g., in generating samples close to the real ones).
- After training, the outputs of G are synthetic data that closely resemble the real samples (e.g., faces of persons that do not exist).

# Graph Neural Networks (1)

- Neural networks have been traditionally used to operate on fixed-size and/or regular-structured inputs (e.g., images). GNNw aim at elegantly process graph-structured data.



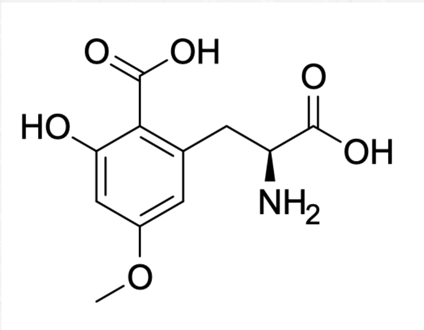
- Localized convolution mimicking CNNs:



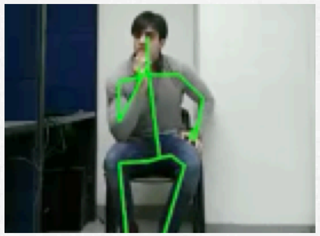

filters are applied on interconnected nodes

# Graph Neural Networks (2)

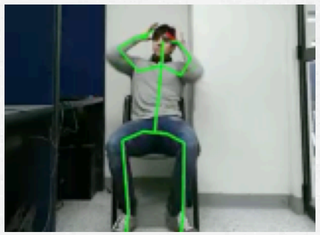
- Example application include:
- graph classification (toxic molecule or not, body posture, etc)
- node classification, node clustering
- temporal graphs, etc..



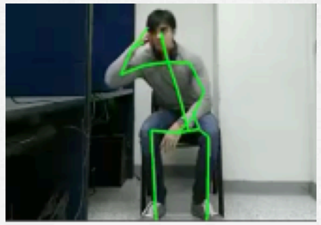

toxic molecule



thoughtful



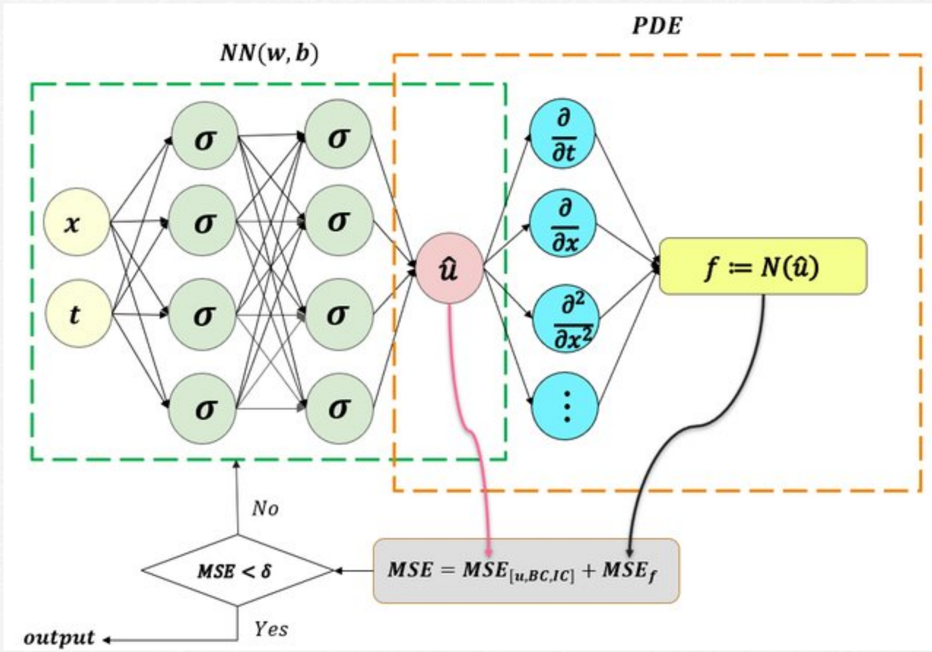
astonished



sad

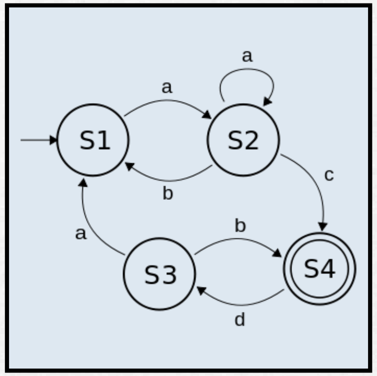
# Physics-Informed Neural Networks

- The prior knowledge of general physical laws acts in the training of neural networks (NNs) as a regularization agent that limits the space of admissible solutions, increasing the correctness of the function approximation.



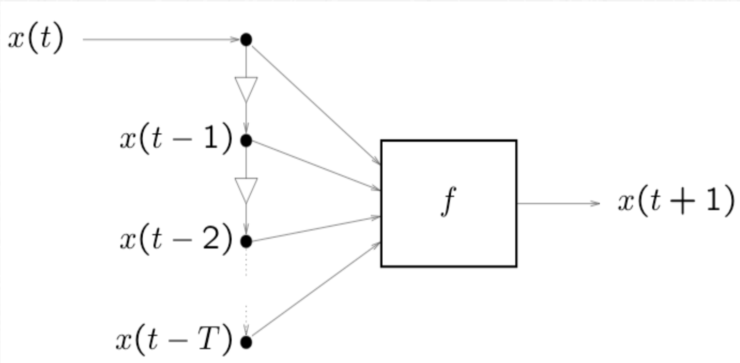
# “Sequential” Neural Networks

- **Taking care of sequences:** lots of information that we store in our brains is not random access, because they were learned as a sequence. Examples:
  - Try to list the alphabet backwards
  - Try to list the musical notes in an octave backwards
  - Try to say your phone number backwards

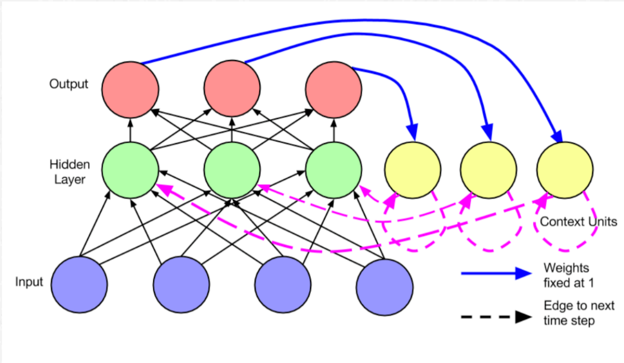



- **Lots of data is of temporal nature** and generally does not change in an abrupt manner. Examples:
  - ECG, temperature, stock values, etc. (time series)

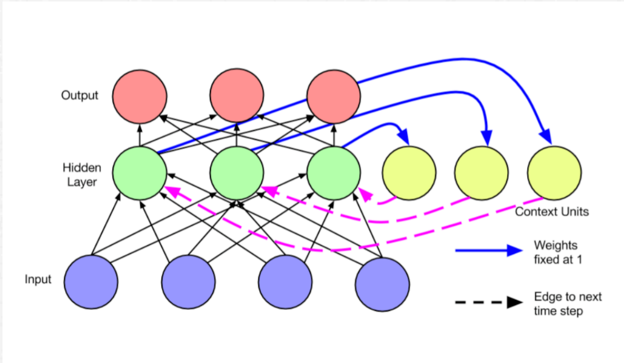
# Discrete time recurrent neural architectures




Time-Delay Neural Nets




Jordan Networks (1986)

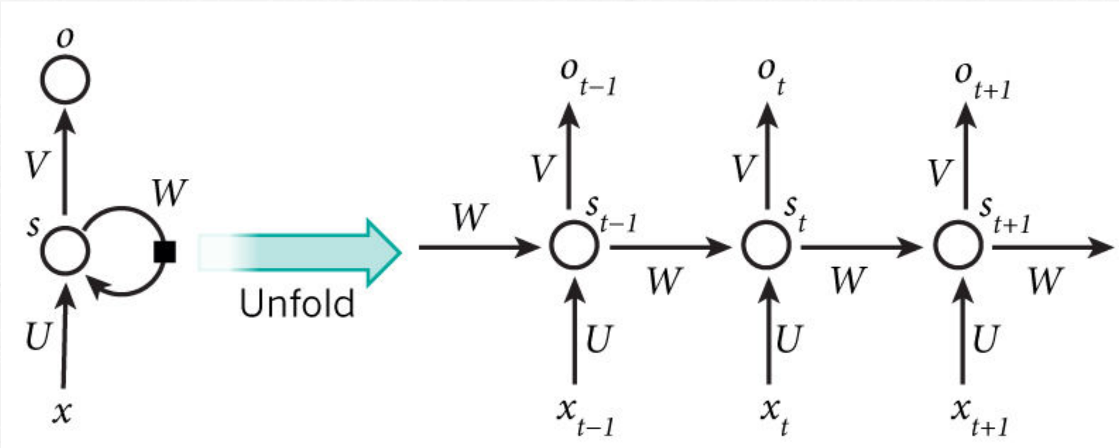



Elman Networks (1990)

from Lipton et al., "A Critical Review of Recurrent Neural Networks for Sequence Learning.", http://arxiv.org/abs/1506.00019

# Recurrent Neural Networks

- The training is similar to that of a feed-forward network, but each epoch must run through the observations in sequential order.

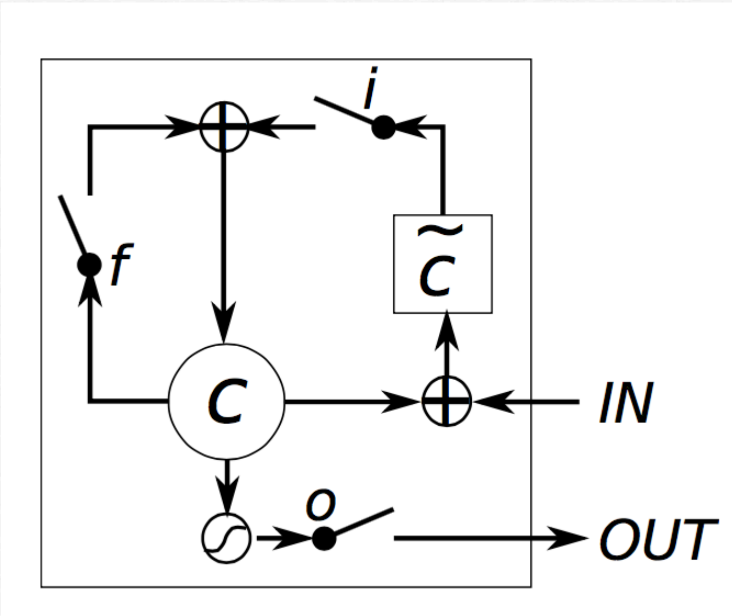



$$\frac{\partial E}{\partial W} = \sum_t \frac{\partial E_t}{\partial W}$$

- Supposing that the network has been unfolded to a depth of k=3, each training pattern consists of [x(t-1), x(t), x(t+1) ; O(t+1)]
- Given an error function $E(O_{t+1},\hat{O}_{t+1})$, the objective is to find W and U by using gradient descent (Backprop Through Time) to minimize E.
- BPTT (Paul Werbos, 1988) suffers from vanishing gradients

# Long Short-Term Memory

- To avoid the vanishing gradients problem, Schmidhuber et al. proposed to gate all operations to store only what is useful for a delayed response (prediction).

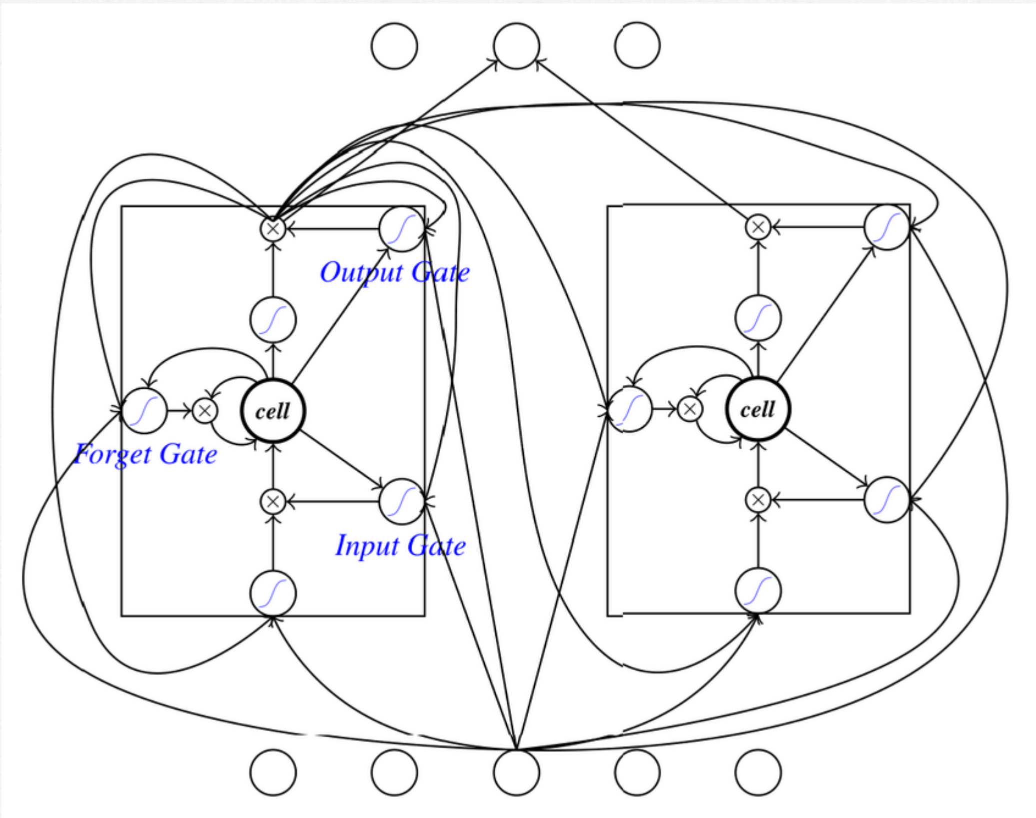



- A simplified model of a so-called "Long short-term memory (LSTM) unit" works like this:
- There are three gates (i,f,o) controlled by "Perceptrons" weighing the inputs and the recurrent outputs
- Given an input IN, we compute a new state c' that can:
  - be ignored (i = 0)
  - replace the previous state (f=0 and i=1)
  - be used to compute a new state C + c' (f=1 and i=1)
- Given a state C, the network outputs OUT (o=1) or not (o=0)

Hochreiter, Sepp; Schmidhuber, Juergen, "LSTM can solve hard long time lag problems", NIPS 1996

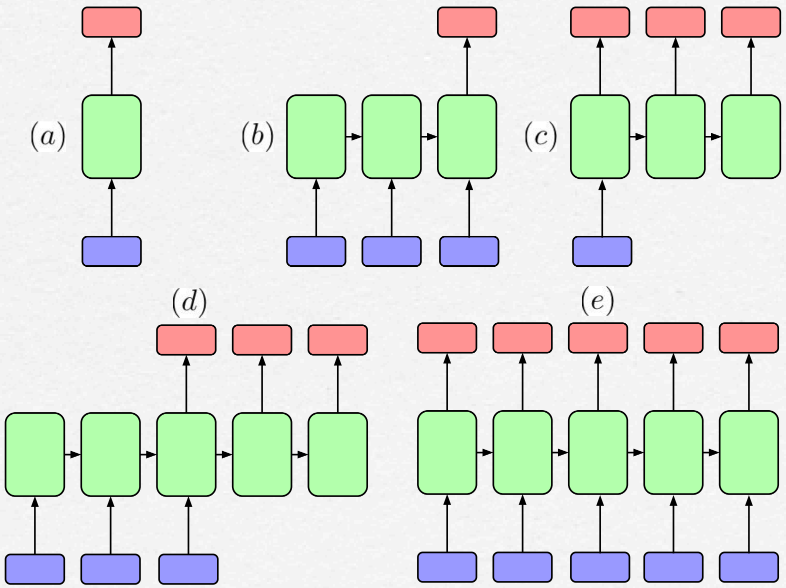
# Recurrent connectivity of LSTMs

y(t+1)




x(t-n), ... x(t-1), x(t)

# LSTM architectures & applications




a) Feed-forward network (not a recurrent architecture)

b) Text and video classification: a sequence is mapped to one fixed length vector

c) Image captioning: the input image is a single non-sequential data point.

d) Natural language translation, a sequence-to-sequence task (they might have varying and different sizes)

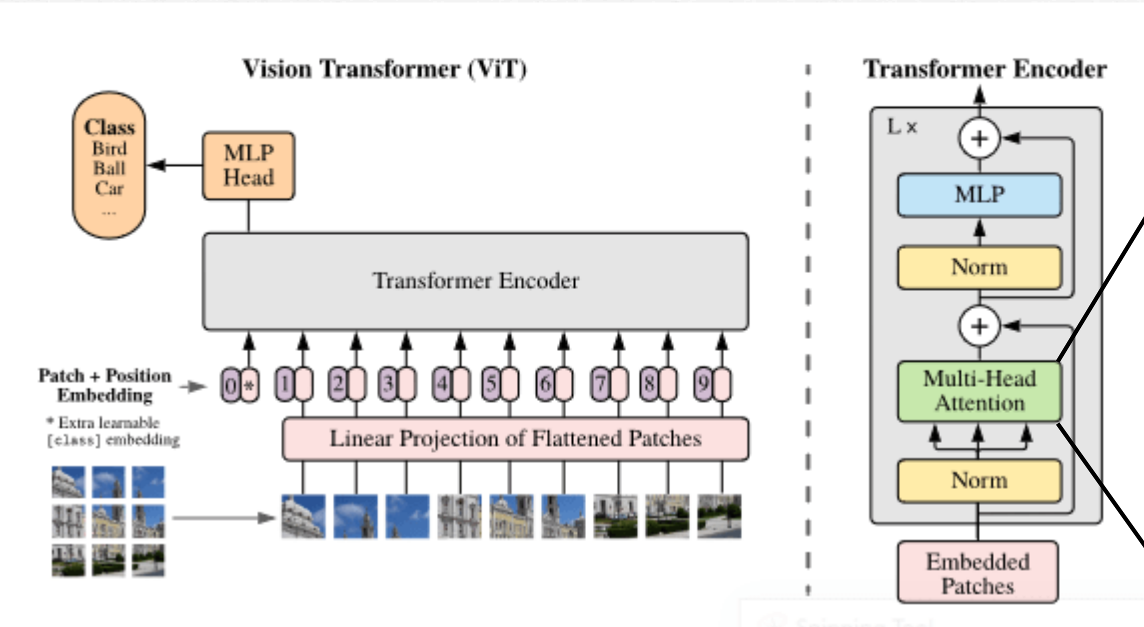
e) Learn a generative model for text, predicting at each step the following character.

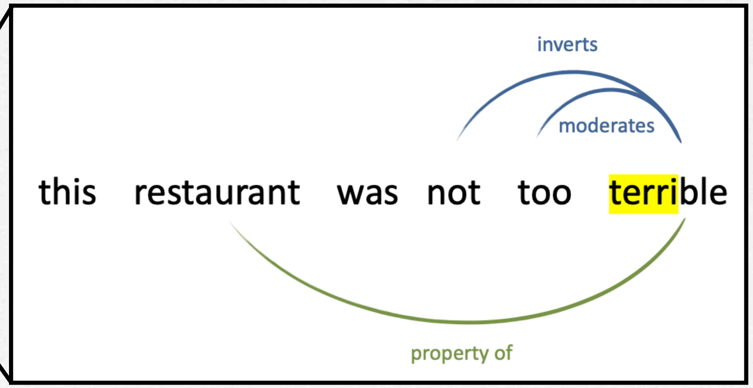
from Lipton et al., "A Critical Review of Recurrent Neural Networks for Sequence Learning.", http://arxiv.org/abs/1506.00019

# Transformers

- An architecture introduced in 2017 by the Google Brain team to deal with sequential data, like language. It processes all input data at once (e.g., not one word at a time). It is the building block of large language models like GPT.

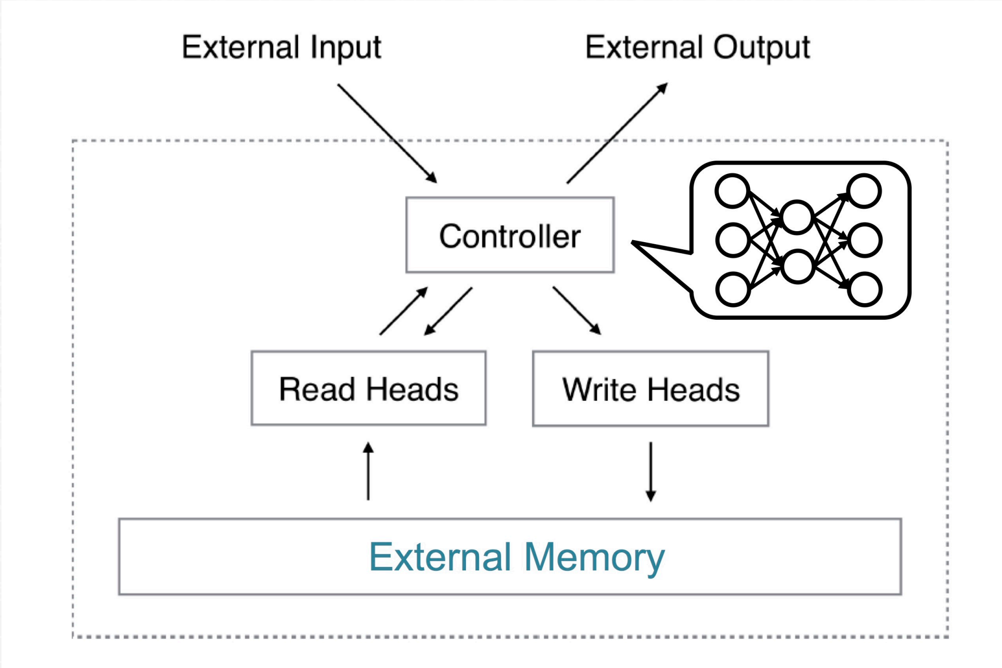
- Vision transformers (ViT) split an image into multiple patches that are then processed like words of a text. The Transformer learns relationships between different portions of the images.



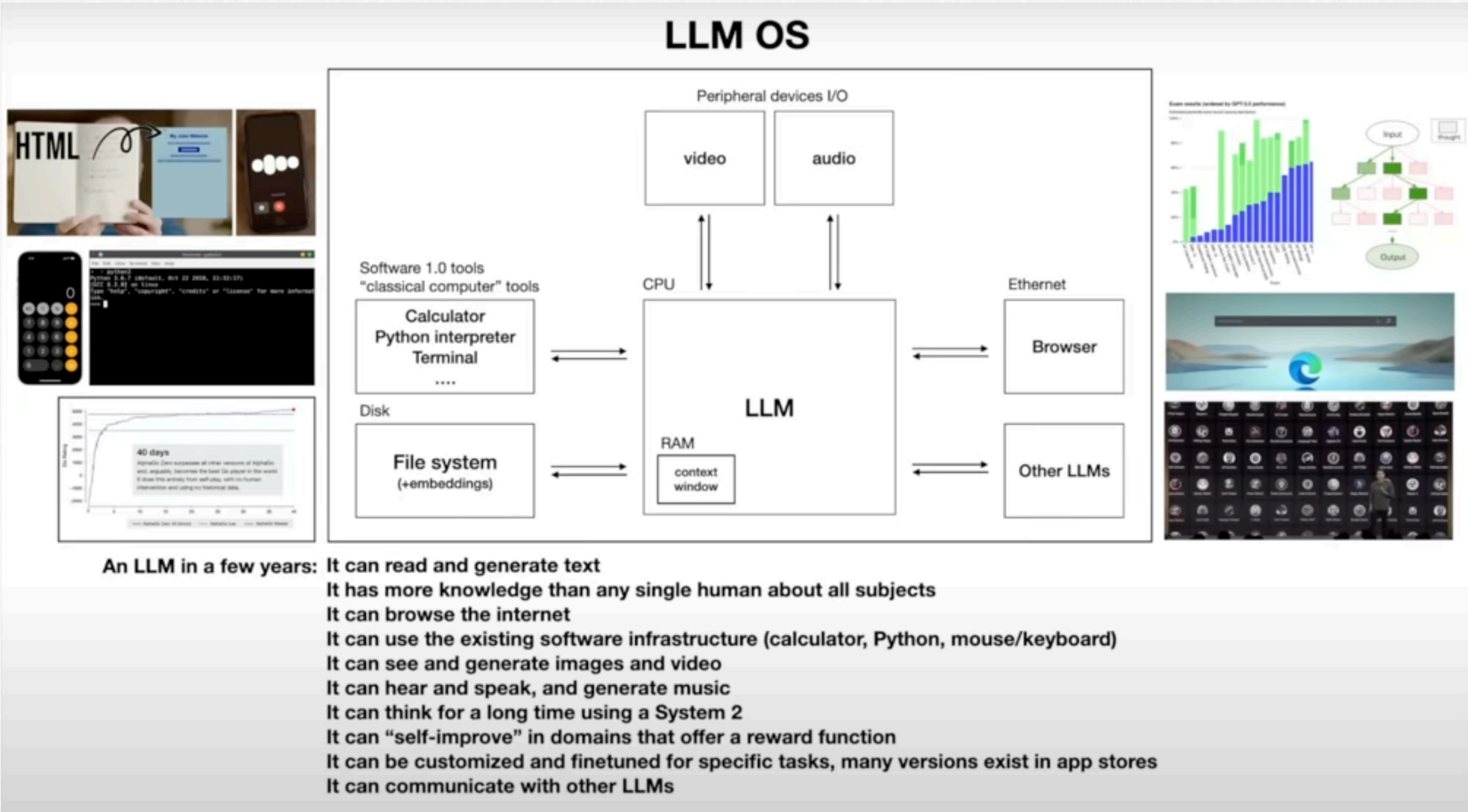



# Neural Turing Machine

- Proposed by Alex Graves (DeepMind; previously worked with Schmidhuber and Hinton) in 2014.
- It is basically a neural controller coupled to external memory resources.
- The memory interactions are differentiable, thus learnable by gradient descent.
- So far, they have allowed the learning of simple algorithms (copying, sorting, associative recall).



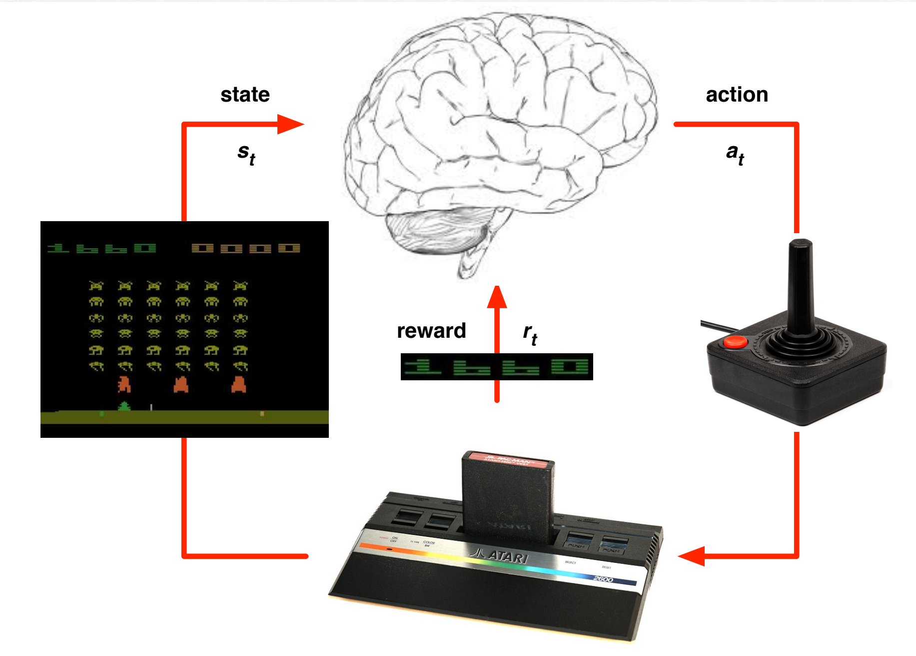
# LLM-based Operating Systems




from Andrej Karpathy

# Deep Q-Networks

- Q-learning is the basic reinforcement learning algorithm (e.g., learning by trail&error coupled to rewards). It learns a value function denoted by Q(s,a) whose values indicate how good it is to take action **a** while being in state **s**.




- In 2015, researchers from DeepMind used CNNs to learn representations from Atari game scenes and approximate the Q(s,a) value function that allows an agent to play the games.

# Further courses at HEIG-VD

- Supervised learning (Bayesian, Decision trees, Ensemble models, Support Vector Machines)
- Unsupervised learning (clustering & dimensionality reduction)
- Simulation & Optimisation
- eXplainable AI
- Traitement Automatique des Langues (NLP)
- Intelligence Artificielle pour les systèmes autonomes
- Machine Intelligence (semi-supervised learning, RL, AI & creativity, collective intelligence, artificial evolution, artificial life)
- Bioinformatique et biologie computationnelle
- Méthodes d'apprentissage pour l'optimisation
- Introduction à la vision par ordinateur
- etc…