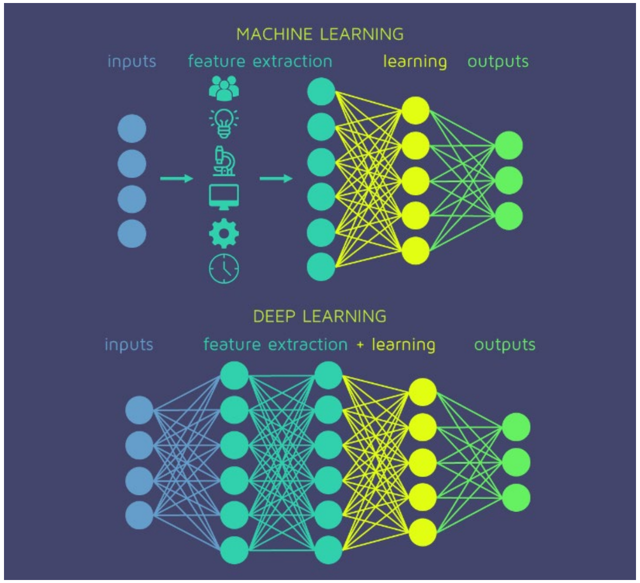
Machine Intelligence
Introduction to Deep Learning



Dr. Ahmad Al-Mahasneh

Slides from:

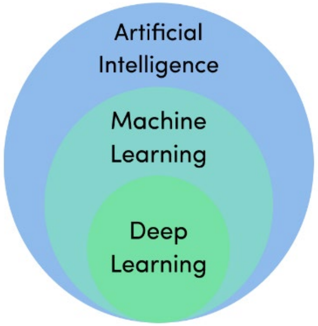
- Introduction to Deep Learning. Professor Qiang Yang http://www.cs.ust.hk/~qyang/
-

  Deep Learning Tutorial Slides PPT - BT Tepper bt.tepper.cmu.edu

Outline

- Introduction
- Convolutional Neural Network
- Sequence Modelling: RNN and its extensions

What is Deep Learning?



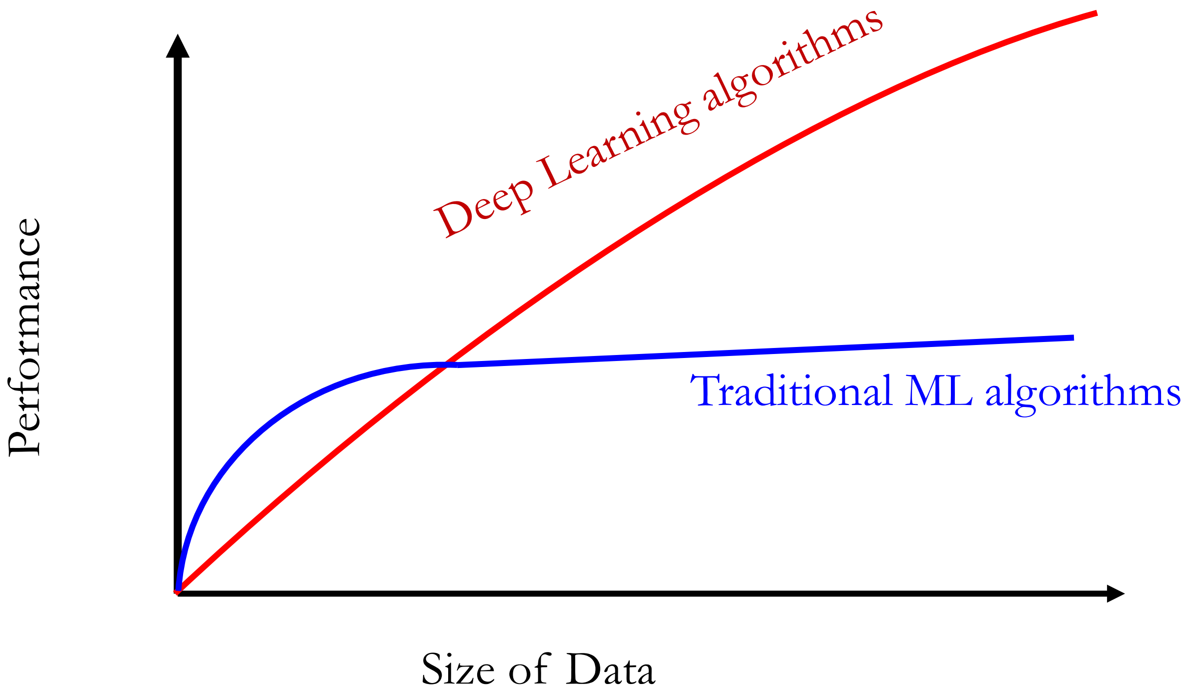
- Deep learning is a recent buzz word for neural networks.

- It describes **machine learning** framework that shows impressive performance on many **Artificial Intelligence** tasks.

- While increasing the hidden layers in the shallow networks does not improve its performance (sometimes it will degrade the performance), the increased number of layers in deep learning **allows for automatically learning the features in the input data (image).**

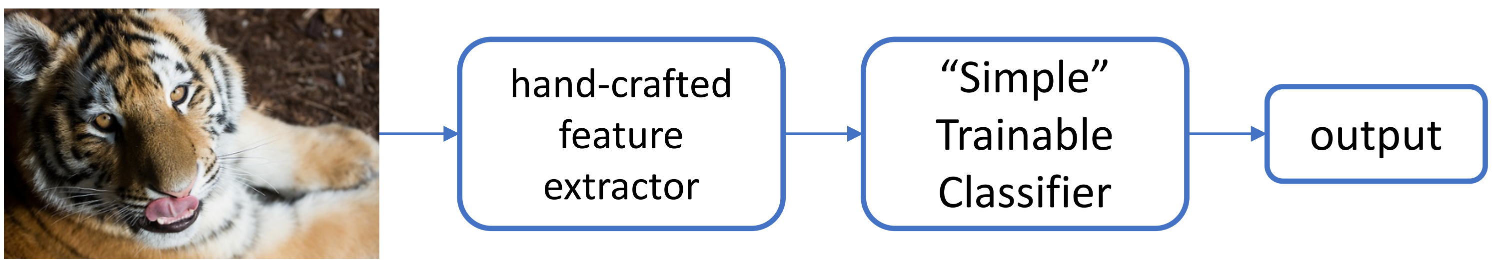
- Machine learning uses algorithms developed for specific tasks. Deep learning is more of a **data representation** based upon multiple layers of a matrix, where each layer uses output from the previous layer as input.

Deep learning Vs. traditional ML



Introduction

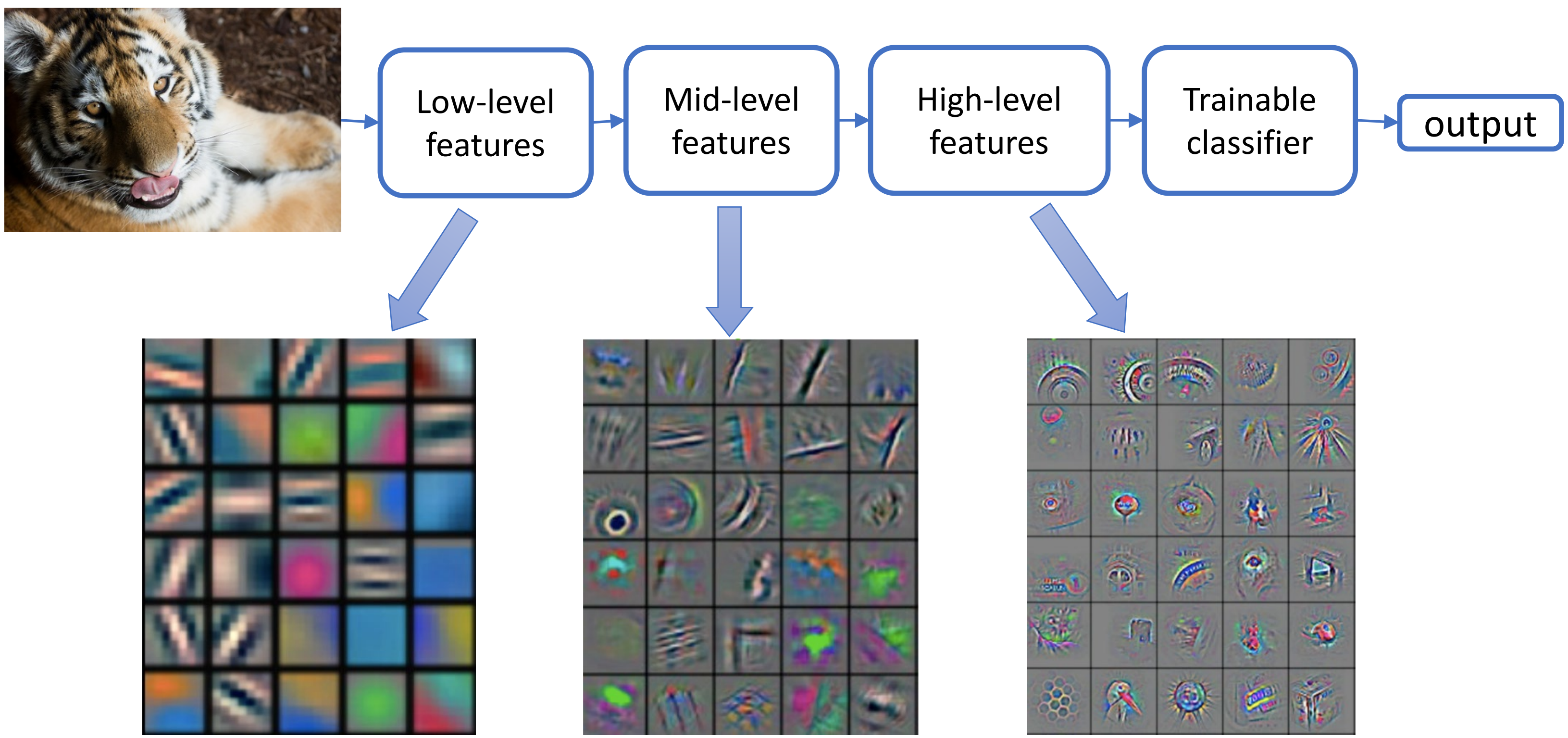
- Traditional pattern recognition models use hand-crafted features and relatively simple trainable classifier.



- This approach has the following limitations:
  - It is very tedious and costly to develop hand-crafted features
  - **The hand-crafted features are usually highly dependents on one application, and cannot be transferred easily to other applications**

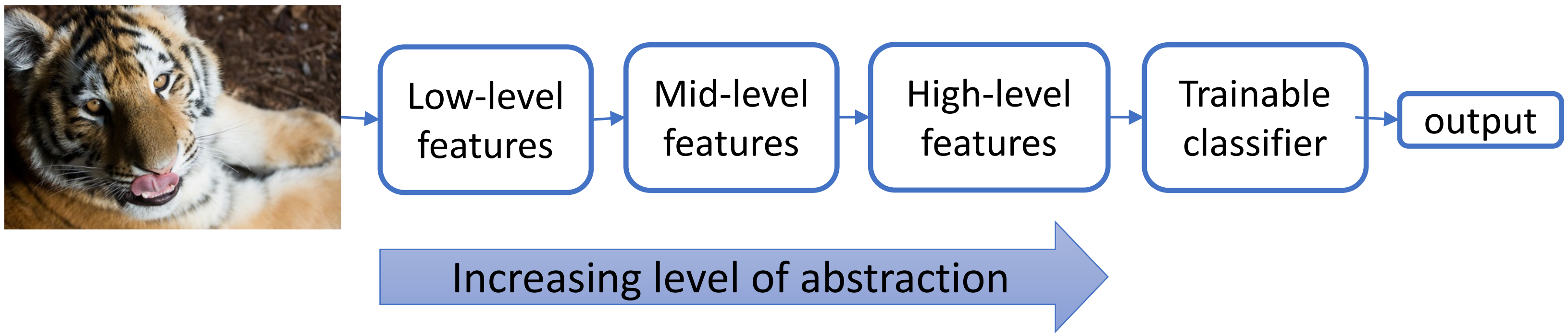
Deep Learning

- Deep learning (representation learning) seeks to learn rich hierarchical representations (i.e. features) automatically through multiple stage of feature learning process.



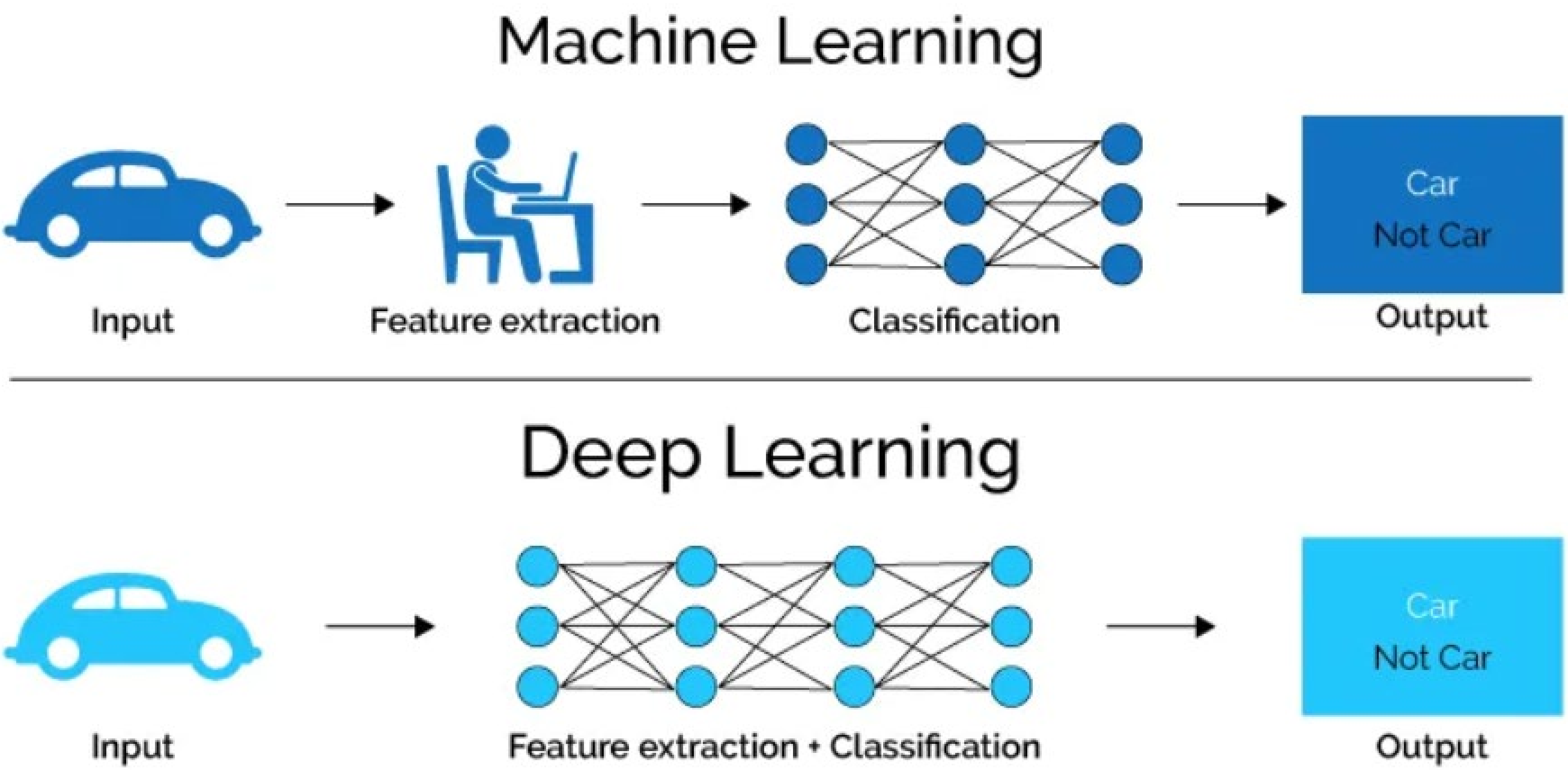
Feature visualization of convolutional net trained on ImageNet
(Zeiler and Fergus, 2013)

Learning Hierarchical Representations

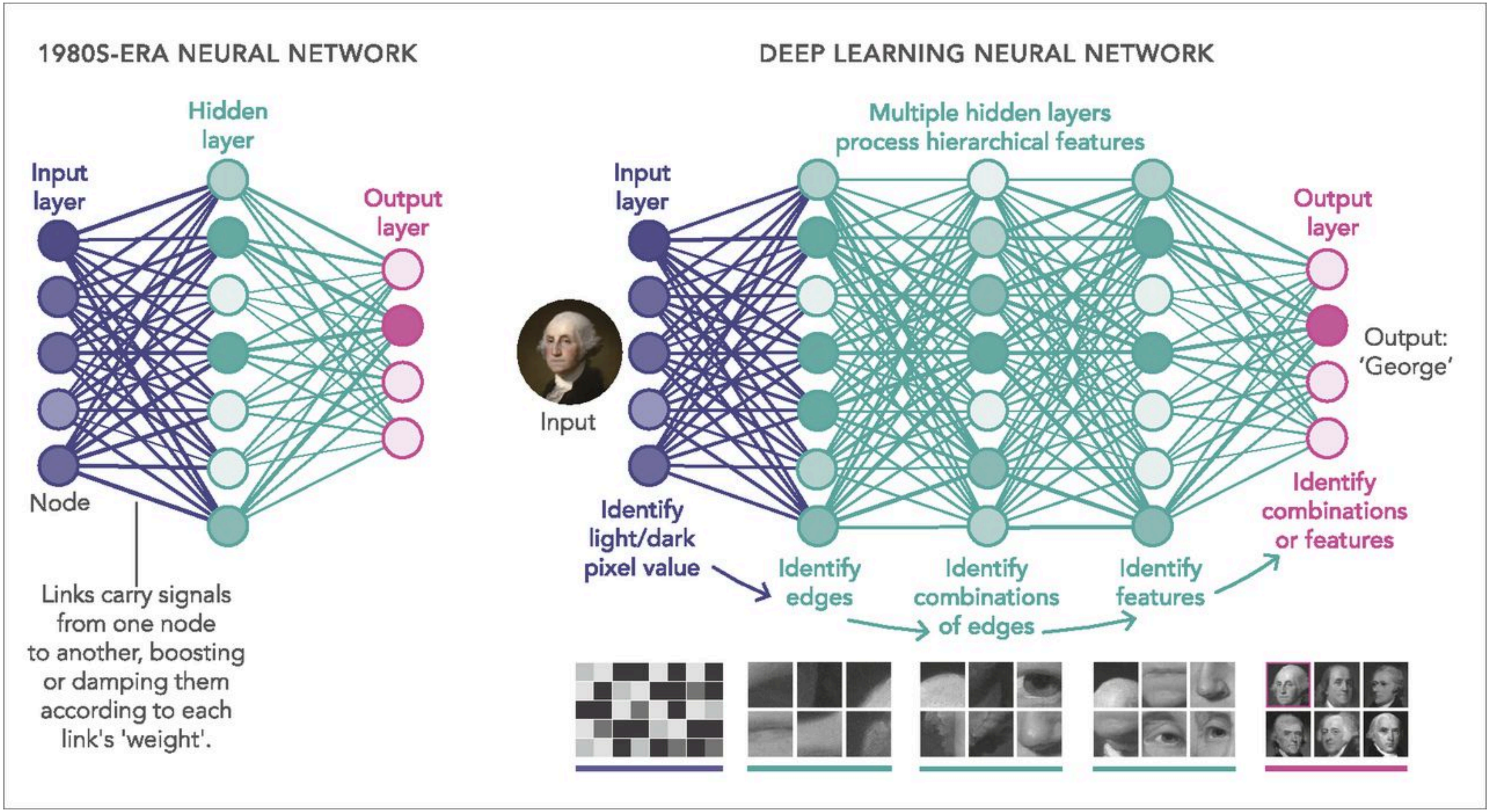


- Hierarchy of representations with increasing level of abstraction. Each stage is a kind of trainable nonlinear feature transform
- Image recognition
  - Pixel → edge → texton → motif → part → object
- Text
  - Character → word → word group → clause → sentence → story

Machine learning Vs. Deep learning



Machine learning Vs. Deep learning

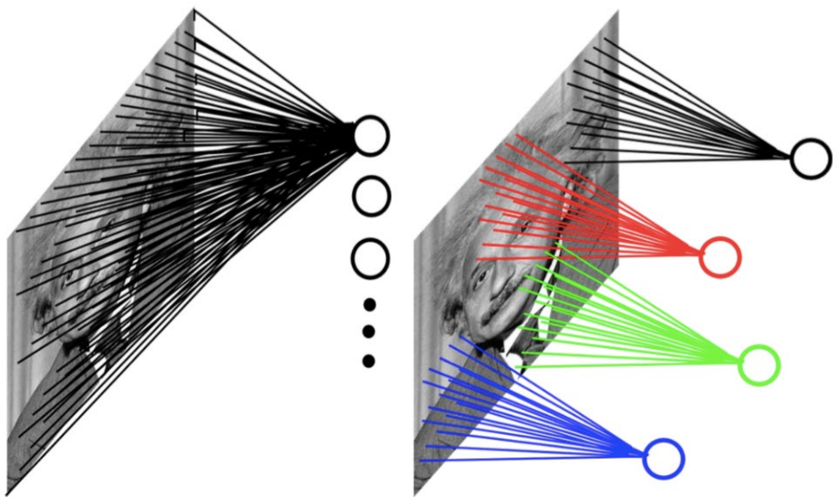


Supervised Learning

- Convolutional Neural Network
- Sequence Modelling
  - Why do we need RNN?
  - What are RNNs?
  - RNN Extensions
  - What can RNNs can do?

Convolutional Neural Network

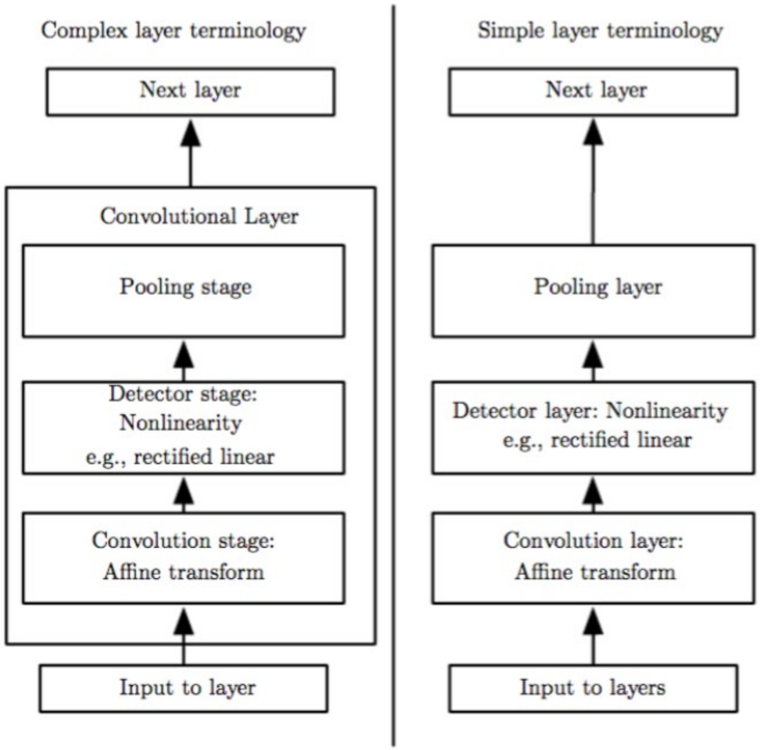
- Input can have very high dimension. Using a fully-connected neural network would need a large amount of parameters.
- Inspired by the neurophysiological experiments conducted by [Hubel & Wiesel 1962], CNNs are a special type of neural network whose hidden units are only connected to local receptive field. The number of parameters needed by CNNs is much smaller.



Example: 200x200 image

a) fully connected: 40,000 hidden units => 1.6 billion parameters
b) CNN: 5x5 kernel, 100 feature maps => 2,500 parameters

Three Stages of a Convolutional Layer

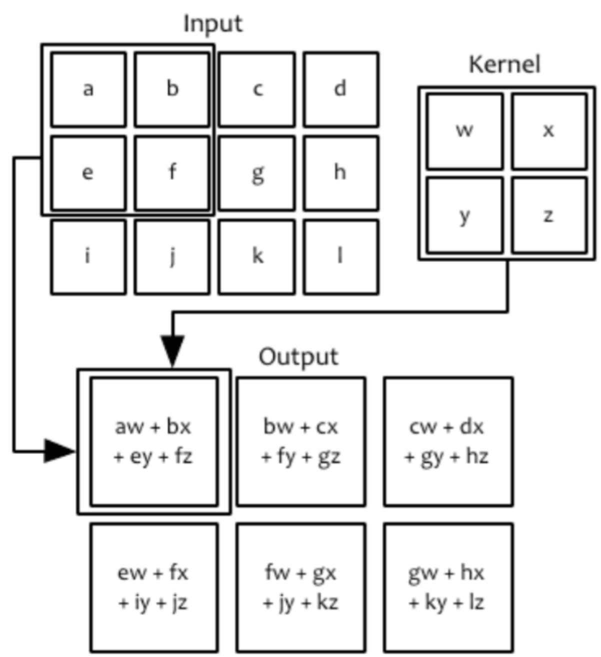


1. Convolution stage
2. Nonlinearity: a nonlinear transform such as rectified linear or tanh
3. Pooling: output a summary statistics of local input, such as max pooling and average pooling

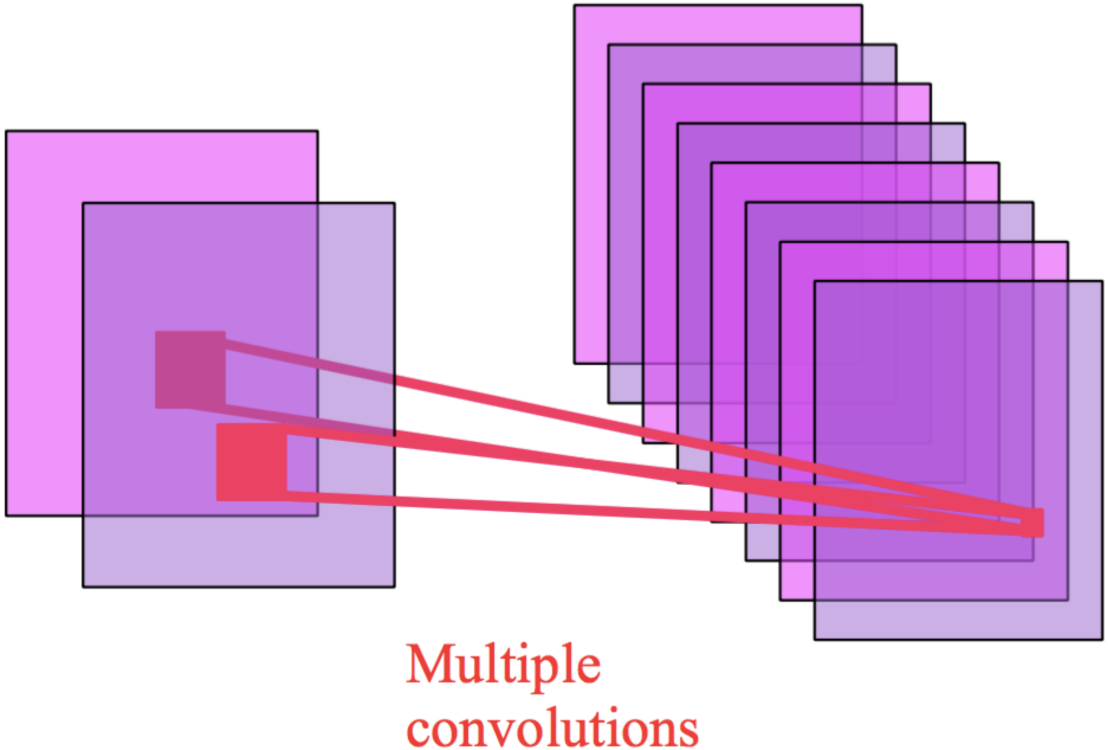
Convolution Operation in CNN

- Input: an image (2-D array) x
- Convolution kernel/operator(2-D array of learnable parameters): w
- Feature map (2-D array of processed data): s
- Convolution operation in 2-D domains:

$$s[i,j] = (x * w)[i,j] = \sum_{m=-M}^{M} \sum_{n=-N}^{N} x[i+m, j+n]\, w[m,n]$$



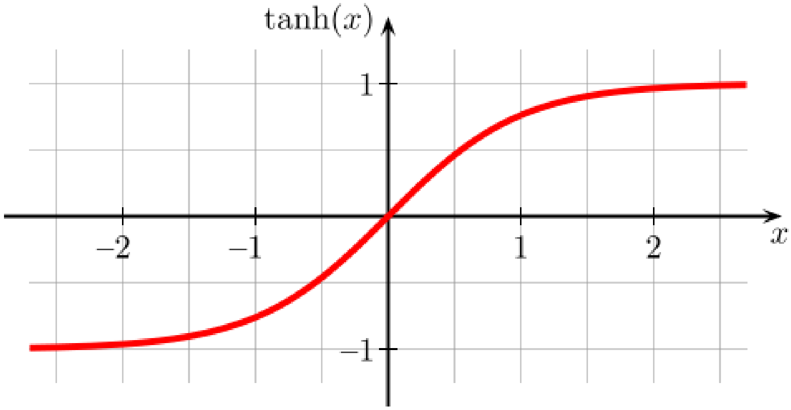
Multiple Convolutions



Usually there are multiple feature maps,
one for each convolution operator.

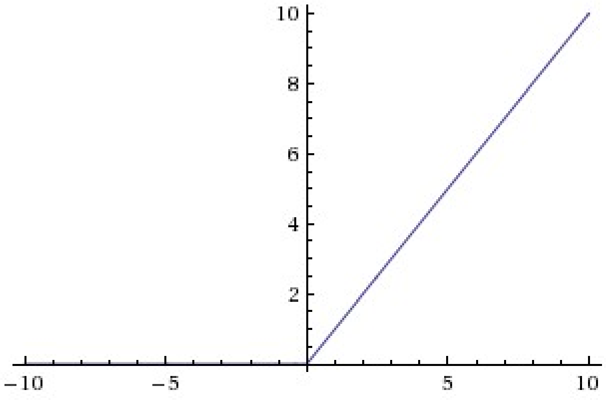
Non-linearity

Tanh(x)



$$\tanh(\mathrm{x}) = \frac{\mathrm{e}^{\mathrm{x}} - \mathrm{e}^{-\mathrm{x}}}{\mathrm{e}^{\mathrm{x}} + \mathrm{e}^{-\mathrm{x}}}$$

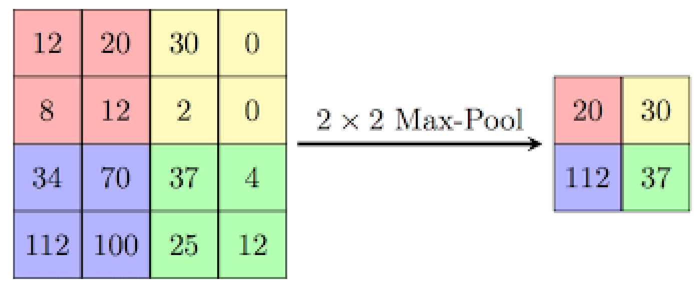
ReLU

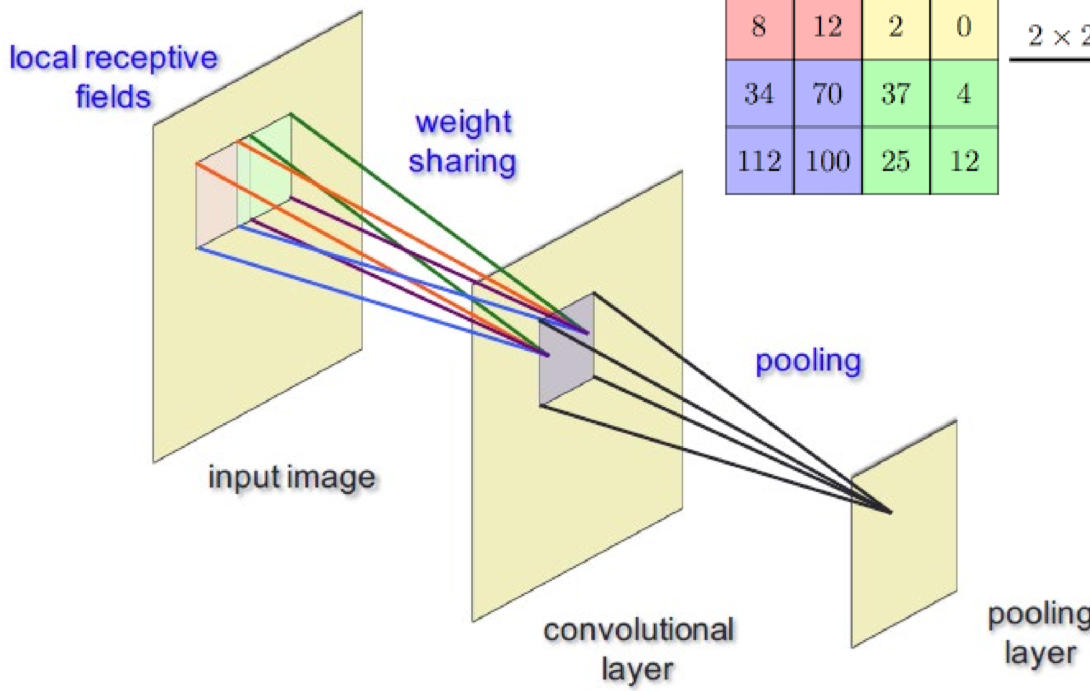


$$f(x) = \max(0, x)$$

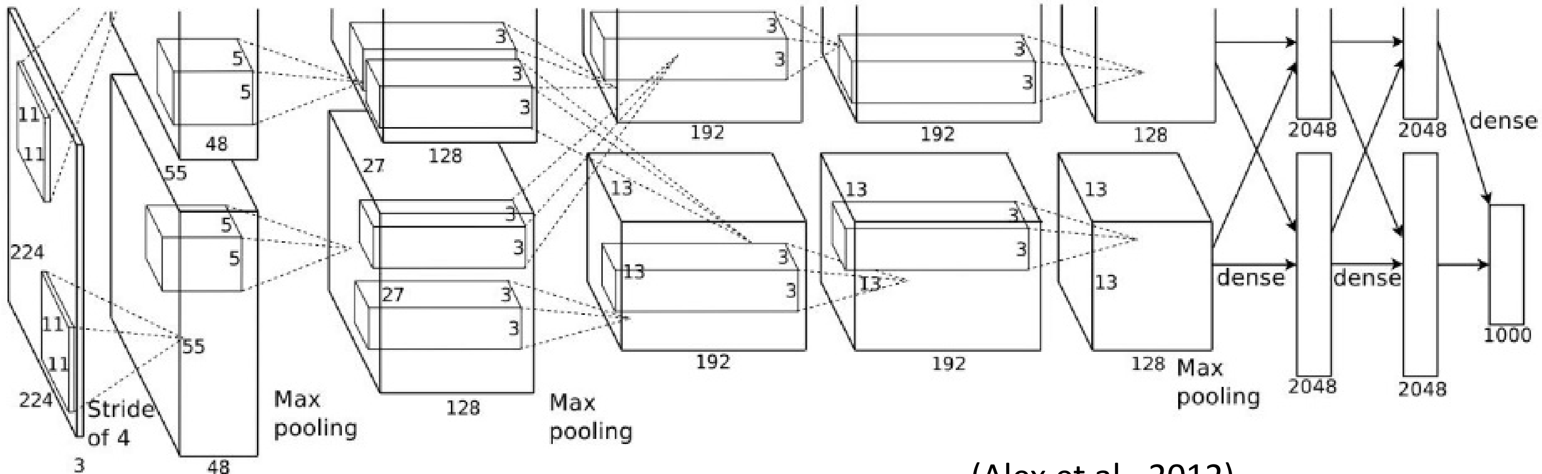
Pooling

- Common pooling operations:
  - Max pooling: reports the maximum output within a rectangular neighborhood.
  - Average pooling: reports the average output of a rectangular neighborhood (possibly weighted by the distance from the central pixel).





Deep CNN: winner of ImageNet 2012



(Alex et al., 2012)

- Multiple feature maps per convolutional layer.
- Multiple convolutional layers for extracting features at different levels.
- Higher-level layers take the feature maps in lower-level layers as input.

Deep CNN for Image Classification

Classification

Click for a Quick Example

Maximally accurate	Maximally specific
cat	1.79306
feline	1.74269
domestic cat	1.70760
tabby	0.94807
domestic animal	0.76846

CNN took 0.064 seconds.

Try out a live demo at
http://demo.caffe.berkeleyvision.org/

Sequence Modelling

- Why do we need RNN?
- What are RNNs?
- RNN Extensions
- What can RNNs can do?

Why do we need RNNs?

The limitations of the Neural network (CNNs)

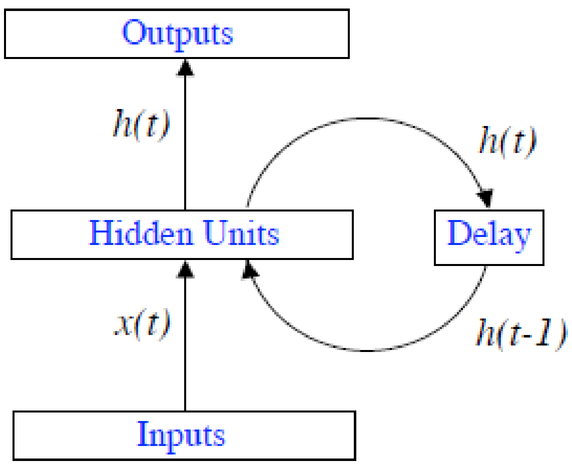
- Rely on the assumption of independence among the (training and test) examples.
  - After each data point is processed, the entire state of the network is lost
- Rely on examples being vectors of fixed length

We need to model the data with temporal or sequential structures and varying length of inputs and outputs

- Frames from video
- Snippets of audio
- Words pulled from sentences

What are RNNs?

Recurrent neural networks (RNNs) are connectionist models with the ability to selectively pass information across sequence steps, while processing sequential data one element at a time.



The simplest form of ***fully recurrent neural network*** is an MLP with the previous set of hidden unit activations feeding back into the network along with the inputs

Allow a 'memory' of previous inputs to persist in the network's internal state, and thereby influence the network output

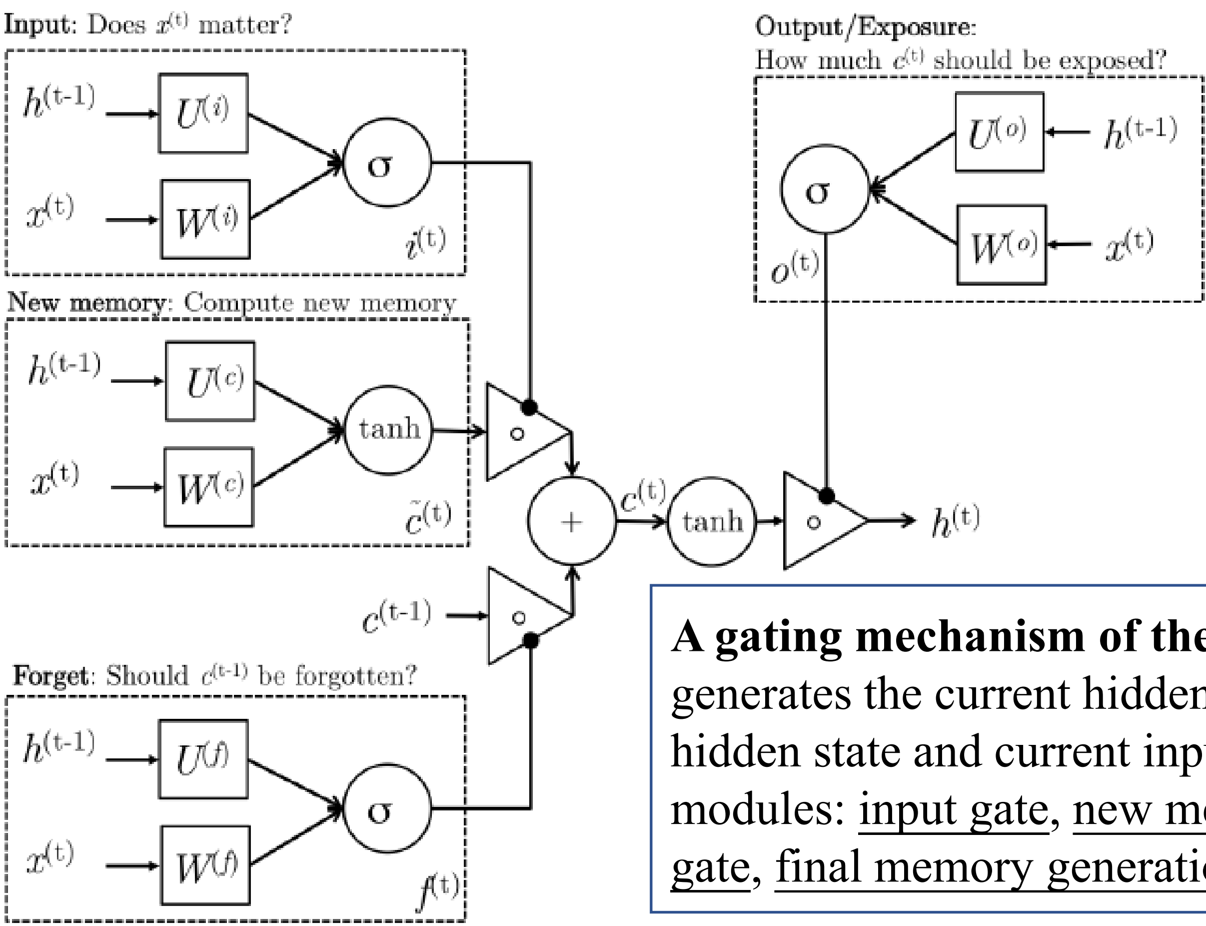
$$h(t) = f_H(W_{IH}x(t) + W_{HH}h(t-1))$$

$$y(t) = f_O(W_{HO}h(t))$$

f_H and f_O are the activation function for hidden and output unit; W_{IH}, W_{HH}, and W_{HO} are connection weight matrices which are learnt by training

RNN Extensions: Long Short-term Memory

The vanishing gradient problem prevents standard RNNs from learning long-term dependencies. LSTMs (Hochreiter and Schmidhuber, 1997) were designed to combat vanishing gradients through a *gating* mechanism.

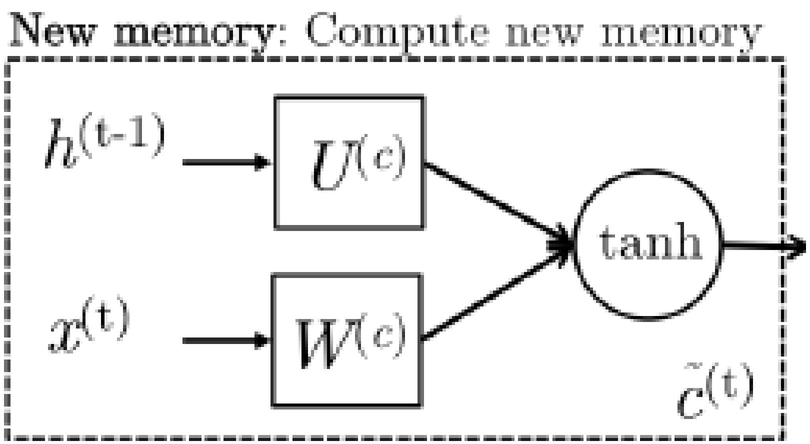


A gating mechanism of the LSTM , which generates the current hidden state by the past hidden state and current input ..It contains five modules: input gate, new memory cell, forget gate, final memory generation, and output gate.

RNN Extensions: Long Short-term Memory

A gating mechanism of the LSTM

New memory cell



$$\tilde{c}_t = \tanh(W^c x_t + U^c h_{t-1})$$

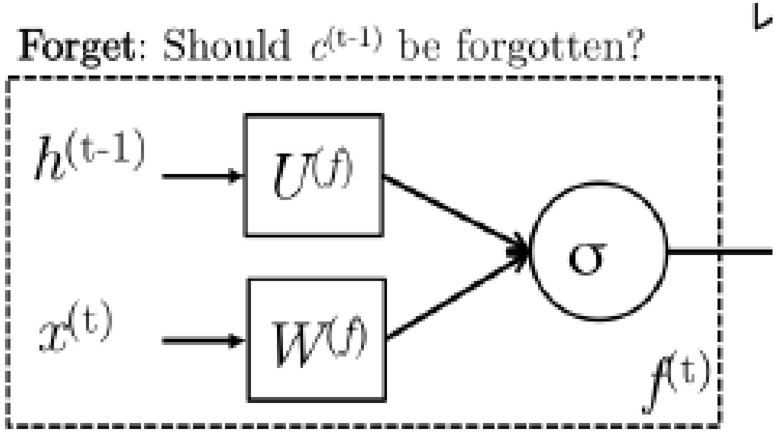
New memory

use the input word and the past hidden state to generate a new memory which includes aspects of the new input

RNN Extensions: Long Short-term Memory

A gating mechanism of the LSTM

Forget gate



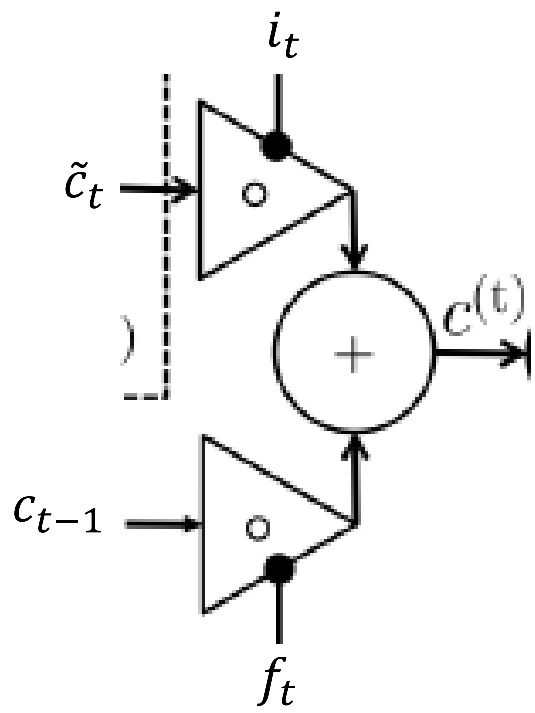
$$f_t = \sigma(W^f x_t + U^f h_{t-1})$$

The forget gate looks at the input word and the past hidden state and makes an assessment on whether the past memory cell is useful for the computation of the current memory cell

RNN Extensions: Long Short-term Memory

A gating mechanism of the LSTM

Final memory cell



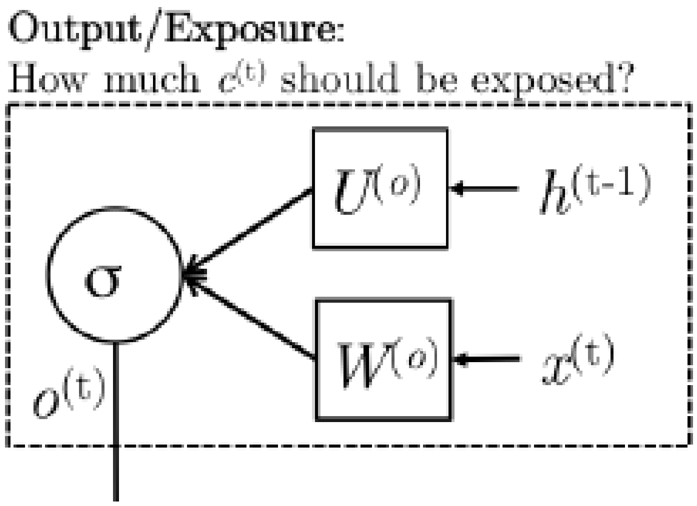
$$c_t = f_t \circ c_{t-1} + i_t \circ \tilde{c}_t$$

This stage first takes the advice of the forget gate f_t and accordingly forgets the past memory c_{t-1}. Similarly, it takes the advice of the input gate i_t and accordingly gates the new memory. It then sums these two results to produce the final memory

RNN Extensions: Long Short-term Memory

A gating mechanism of the LSTM

Output gate



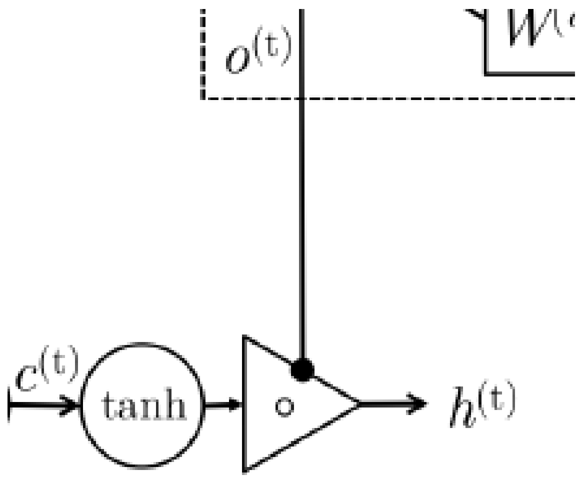
$$o_t = \sigma(W_o x_t + U_o h_{t-1})$$

This gate makes the assessment regarding what parts of the memory c_t needs to be exposed/present in the hidden state h_t.

RNN Extensions: Long Short-term Memory

A gating mechanism of the LSTM

The hidden state



$$h_t = o_t \circ \tanh(c_t)$$

What can RNNs can do?

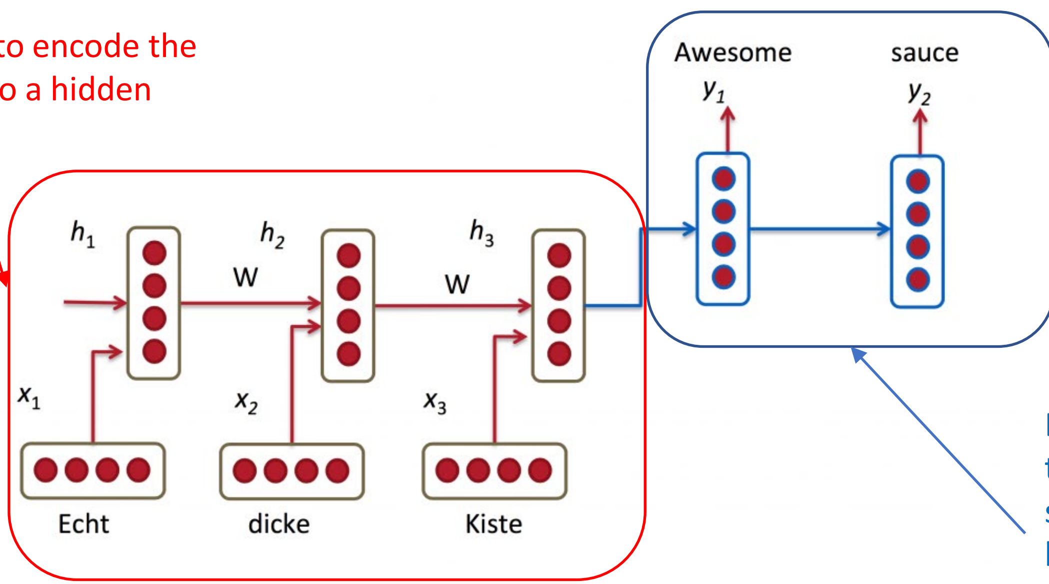
Machine Translation

Visual Question Answering

Machine Translation

In machine translation, the input is a sequence of words in source language, and the output is a sequence of words in target language.

Encoder: An RNN to encode the input sentence into a hidden state (feature)



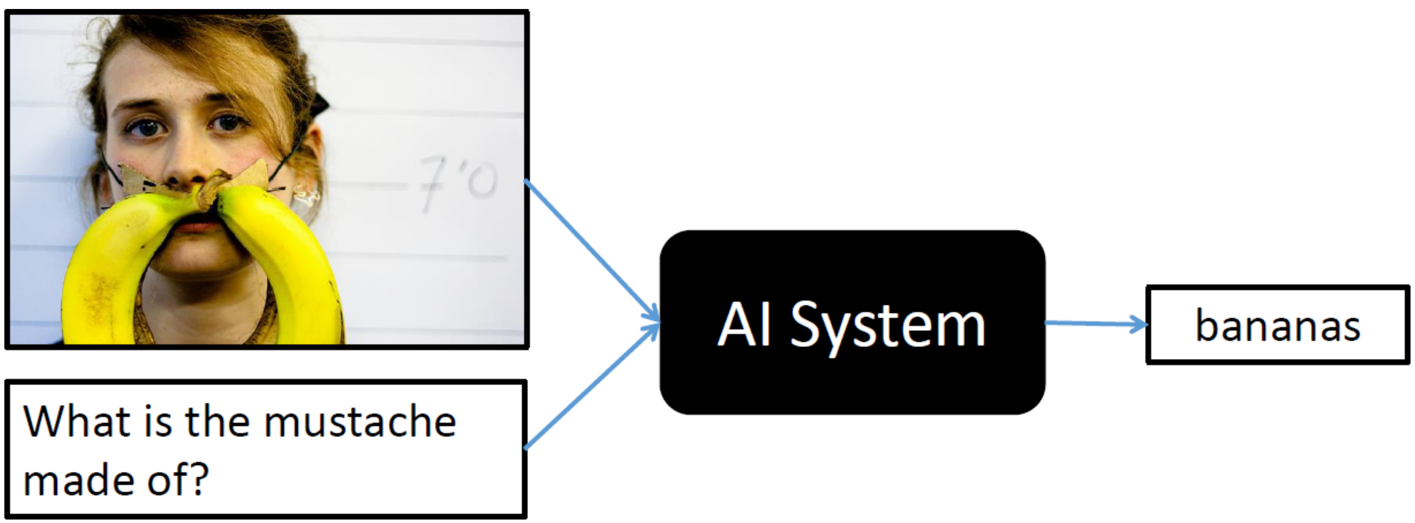
Decoder: An RNN with the hidden state of the sentence in source language as the input and output the translated sentence

Encoder-decoder architecture for machine translation

Visual Question Answering (VQA)

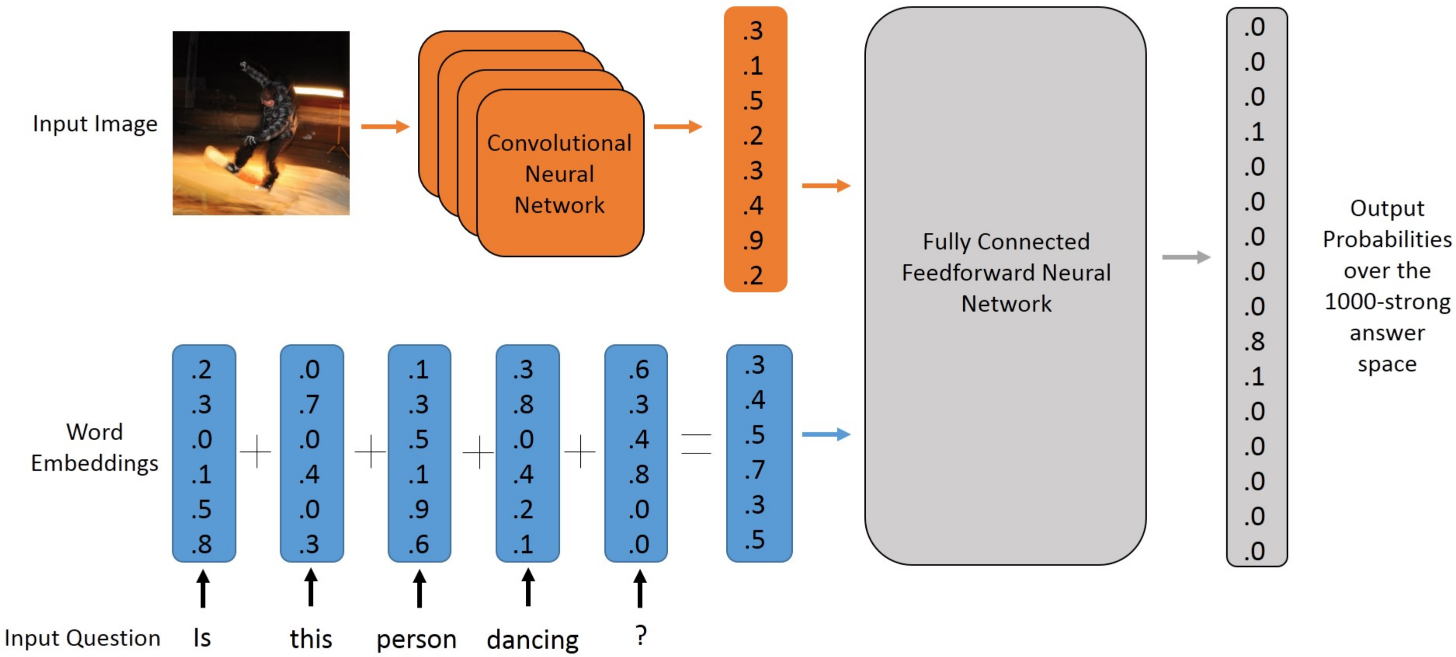
[Demo Website]

VQA: Given an image and a natural language question about the image, the task is to provide an accurate natural language answer



Picture from (Antol et al., 2015)

Visual Question Answering



The output is to be conditioned on both image and textual inputs. A CNN is used to encode the image and a RNN is implemented to encode the sentence.

References

- Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Lawrence Zitnick, C., & Parikh, D. (2015). VQA: Visual question answering. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 2425-2433).
- Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *Science*, *313*(5786), 504-507.
- Schuster, M., & Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *Signal Processing, IEEE Transactions on*, *45*(11), 2673-2681.
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (pp. 1097-1105).
- Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*(8), 1735-1780.
- Mnih, Volodymyr, et al. "Playing atari with deep reinforcement learning."*arXiv preprint arXiv:1312.5602* (2013).
- Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." *Nature* 518.7540 (2015): 529-533.
- Silver, David, et al. "Mastering the game of Go with deep neural networks and tree search." *Nature* 529.7587 (2016): 484-489.
- Silver, D. (2015). Deep Reinforcement Learning [Powerpoint slides]. Retrieve from http://www.iclr.cc/lib/exe/fetch.php?media=iclr2015:silver-iclr2015.pdf
- Lecun, Y., & Ranzato, M. (2013). Deep Learning Tutorial [Powerpoint slides]. Retrieved from http://www.cs.nyu.edu/~yann/talks/lecun-ranzato-icml2013.pdf
- The deep learning book: introduction.

Slides from:

- Introduction to Deep Learning. Professor Qiang Yang http://www.cs.ust.hk/~qyang/
- Deep Learning Tutorial Slides PPT - BT Tepper bt.tepper.cmu.edu