Machine Learning

Prof. Sergei Gleyzer

Lecture

PH451, PH551

Feb 27, 2025

Final Projects

- **Semester long activity (40% of grade)**
- **Team activity**
  - **Self-designed project in consultation with instructor**
  - **Has to include an ML component**
  - **Can pick any topic you feel passionate about**
  - **Graduate students – can pick something from your specific field or research area**

Final Projects

- **Pre-proposal 5%**
- **Proposal 10%**
- **Project Outline and Demo 25%**
- **Presentation 10%**
- **Peer Evaluation 10%**
- **Final Project Submission (write-up, code) 40%**

Final Project Teams

- **New team assignments**
- **New teams are only for final projects**
  - **continue with existing teams for all remaining exercises and activities**
- **Please start to discuss project ideas**
  - **Pre-Proposals due after spring break**

Pre-Proposals

- Should establish:
  - Your team
  - 3 possible ideas for your project in your order of preference
- Pick potential problems that
  - You have an idea how to solve
  - Relevant to someone
  - Not yet solved (or provide a different solution)

Google Summer of Code 2024

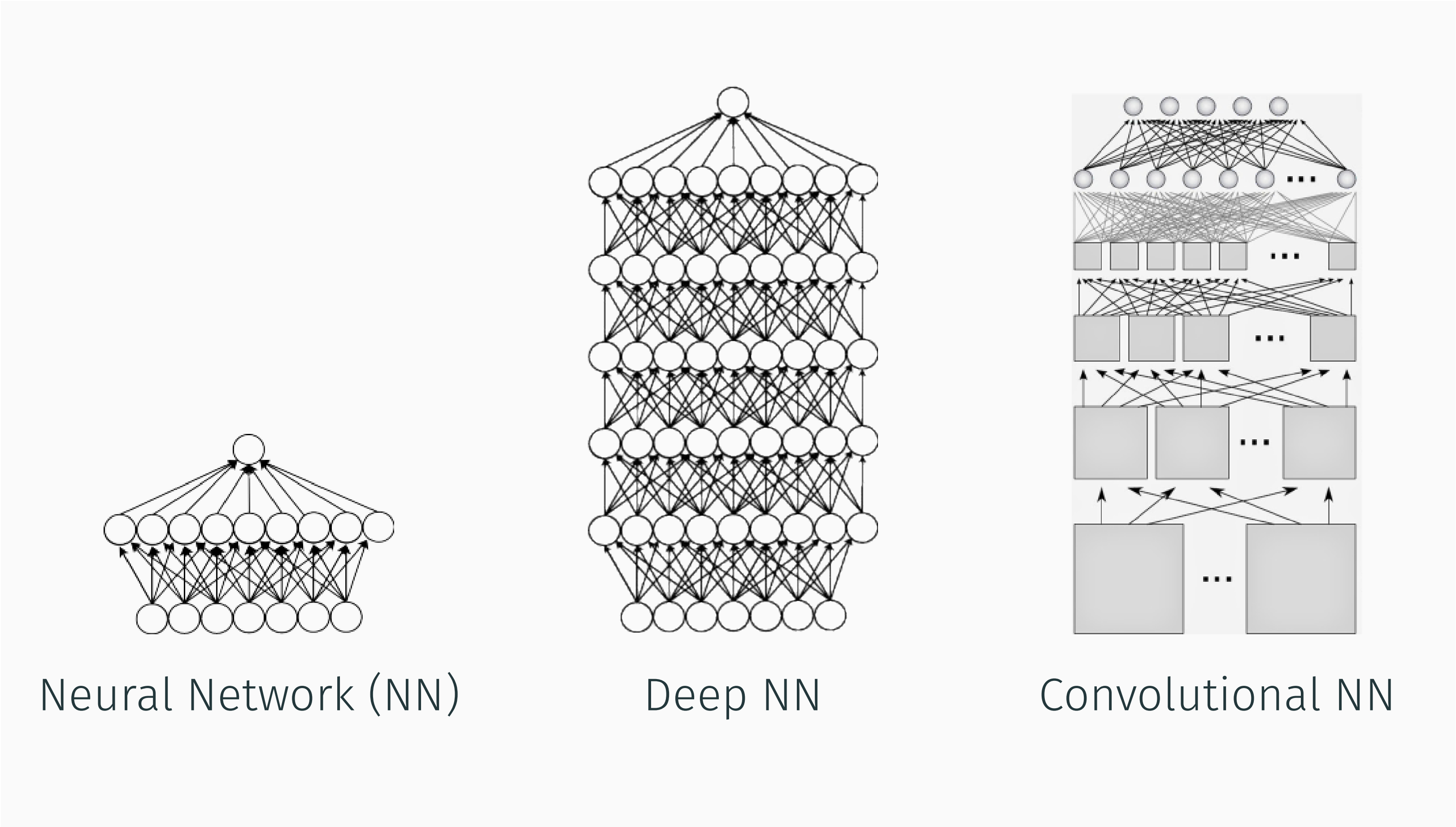
https://summerofcode.withgoogle.com/

- **Machine Learning for Science (ML4SCI): ml4sci.org**

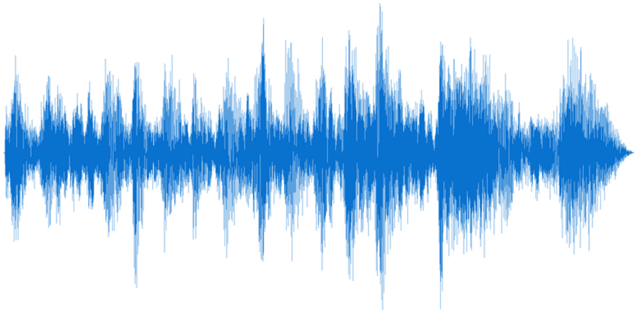
- **HumanAI: humanai.foundation**

Convolutional Networks

Convolutional Networks



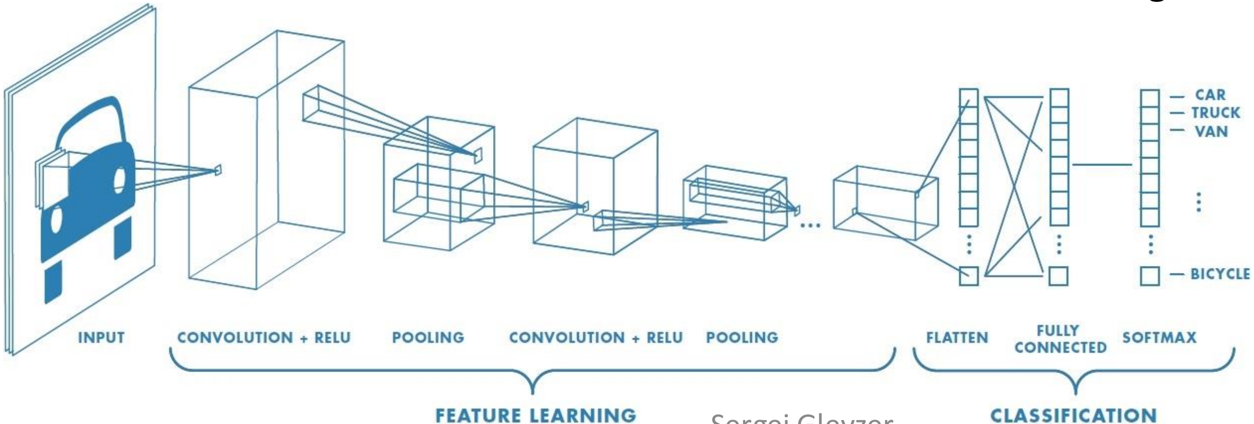
Convolutional Networks



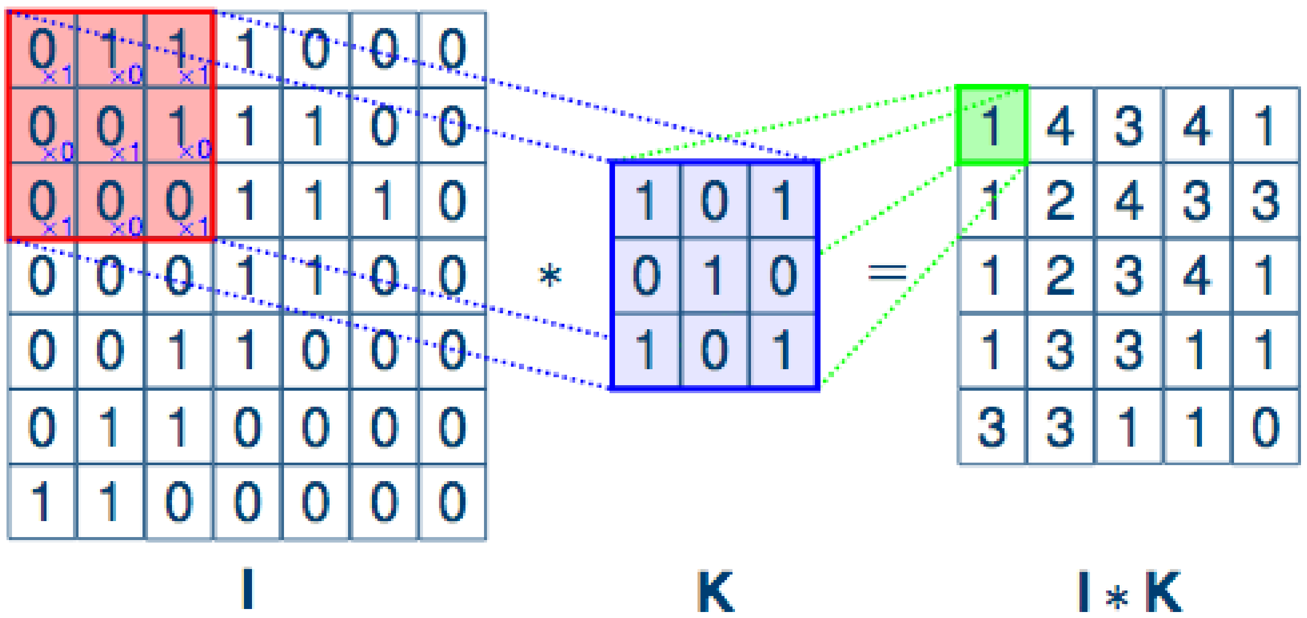
$[x^1\ x^2\ x^3 \ldots x^i]^T$
waveform heights

$[x^{11}\ x^{12} \ldots x^{1n}\ x^{21}\ x^{22} \ldots]^T$
pixel intensities

Feature learning



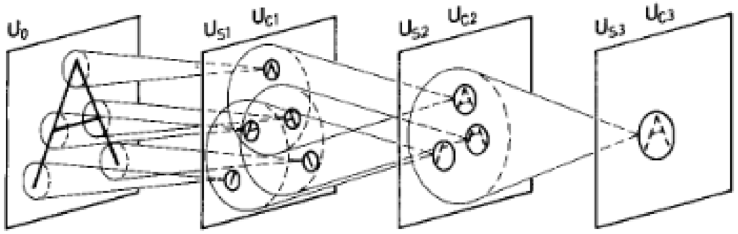
Convolution Example



Exploit structure, neighboring pixel dependence

Convolutional Nets

- Emerged from computer vision
  - Inspired by visual cortex of the brain
    - Simple cells that respond to environment (edges)
    - Complex cells with more response invariance



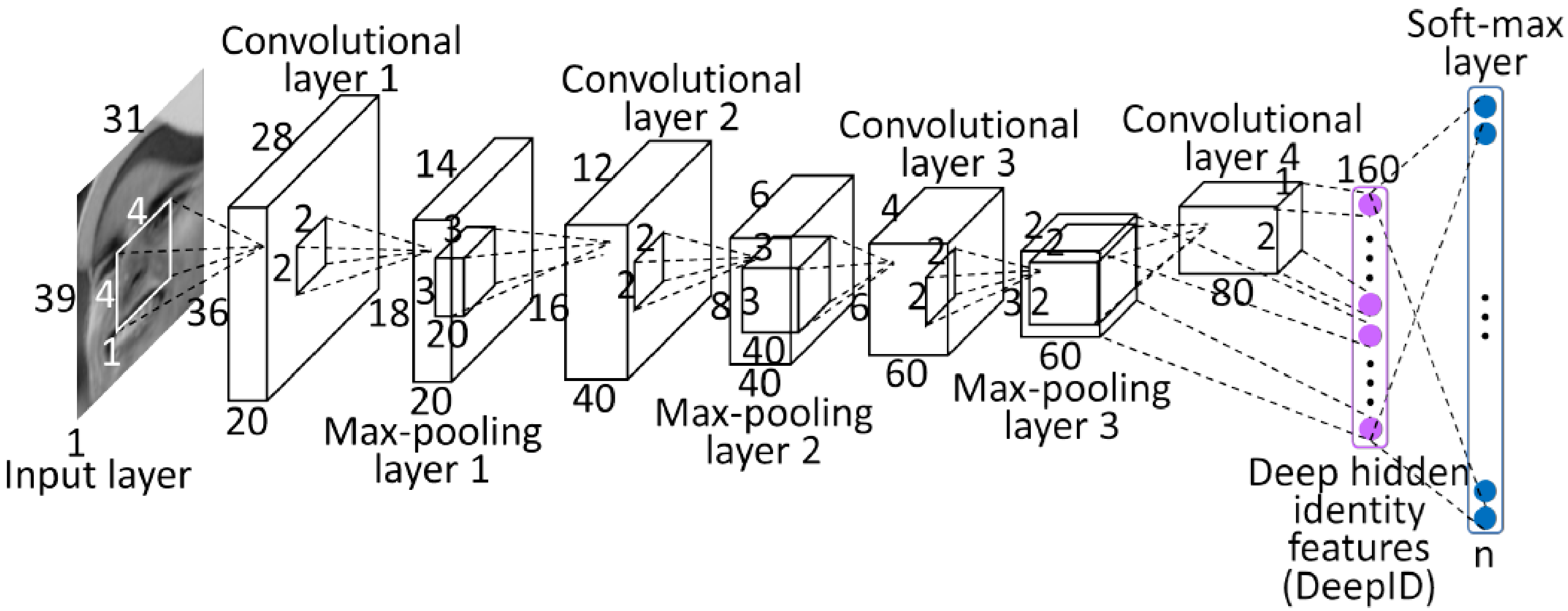
  - Neocognitron (1980)
    - Synthesis, pooling over inputs
    - Early model that inspired CNNs

Convolutional Nets

Convolutional Neural Networks:

- Began with image and sequence-based problems in computer vision
  - Images (2D)
  - CNN’s learn features with simple structures
    - Filters: repeatedly applied
    - Unsupervised learning during first stage

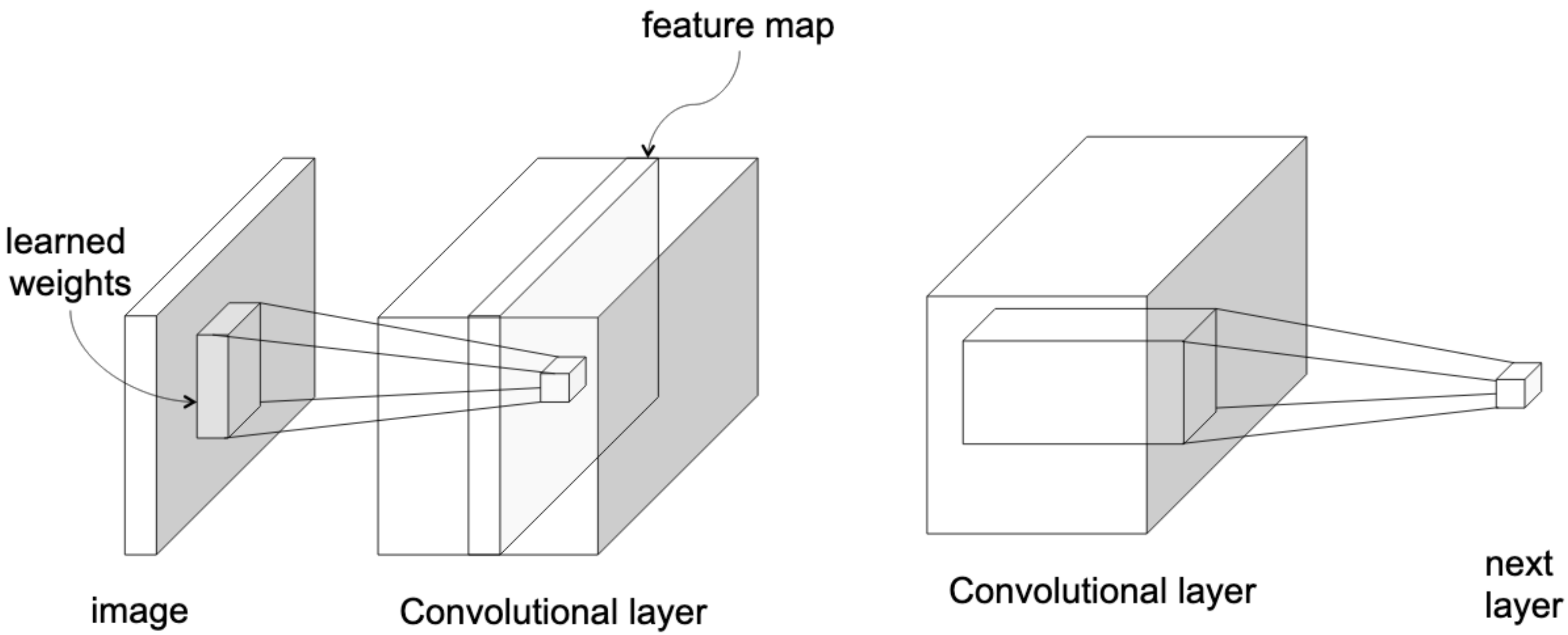
Convolutional NN



Feedforward structure, spatially arranged units:

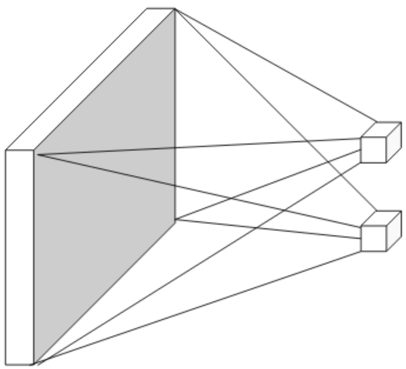
- 2-D feature maps – result of a convolution performed on the previous layer

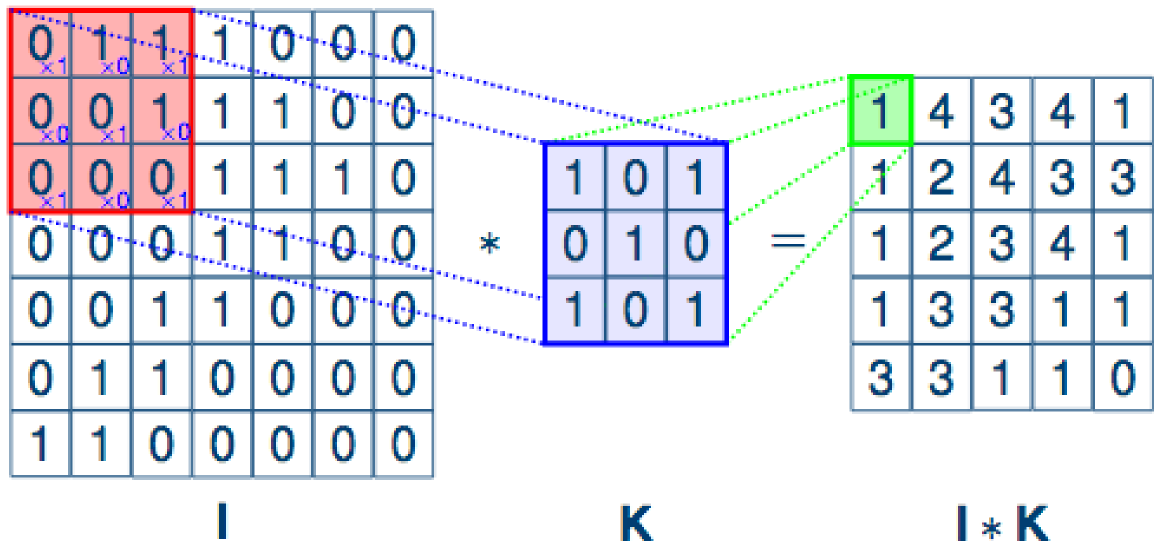
Convolutional NN



Input (Image) → Convolution → Activation (Non-Linear) → Spatial Pooling → Feature Maps

Convolution Example





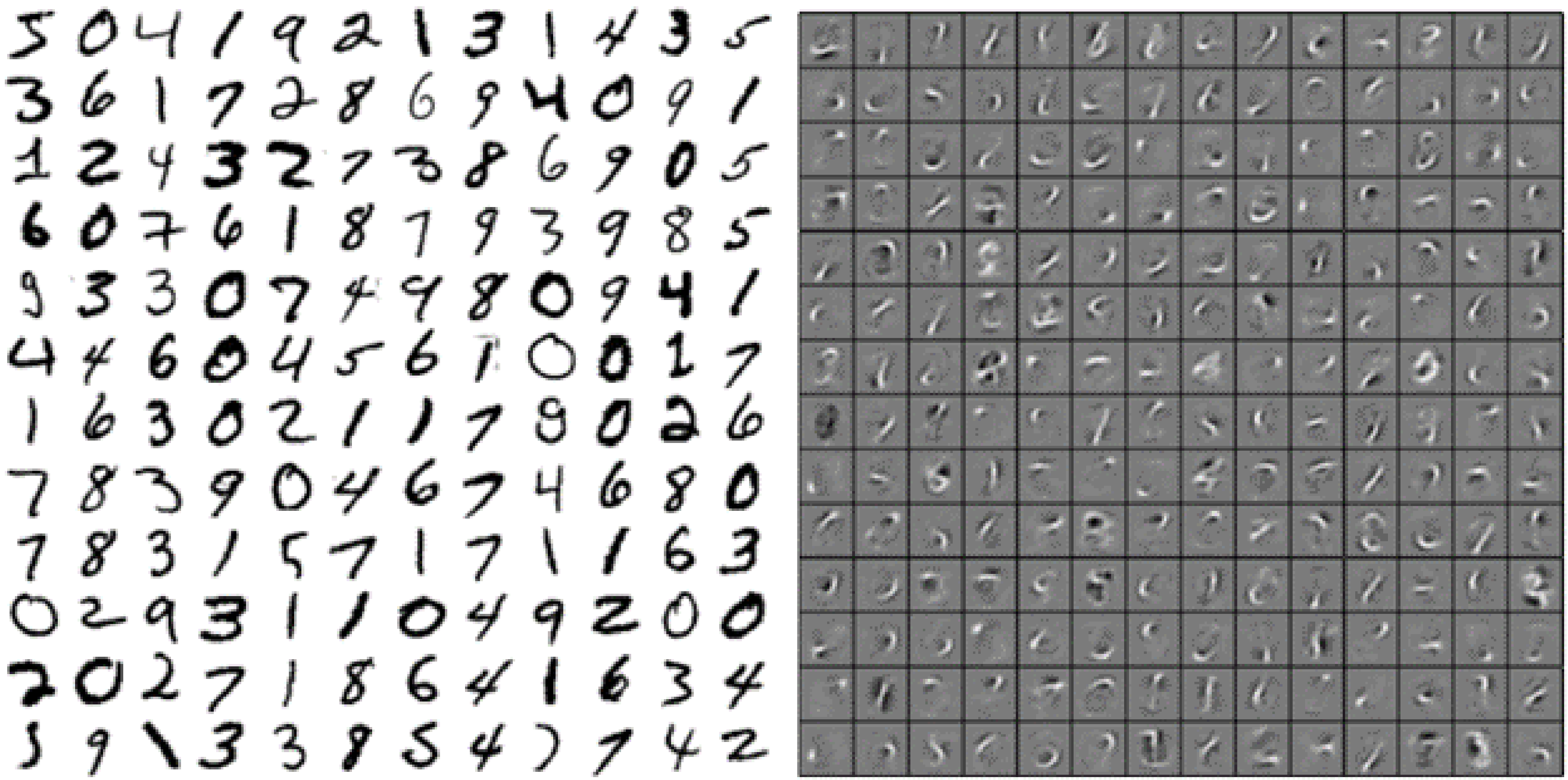
Same convolutional filter (set of weights) is applied to each element

- An element in a single 2-D location can only receive input from elements in similar location from previous layers (locality)
- Same weights for each feature map (and different across maps)
- Exploit structure, neighboring pixel dependence

Filters

Convolutional Neural Networks:

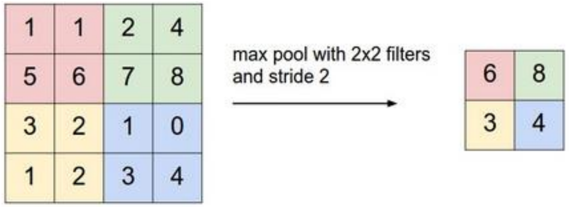
Unsupervised Feature Learning



Pooling

Down-sampling: shrink the size of the feature map

- Lower resolution that still contains important information

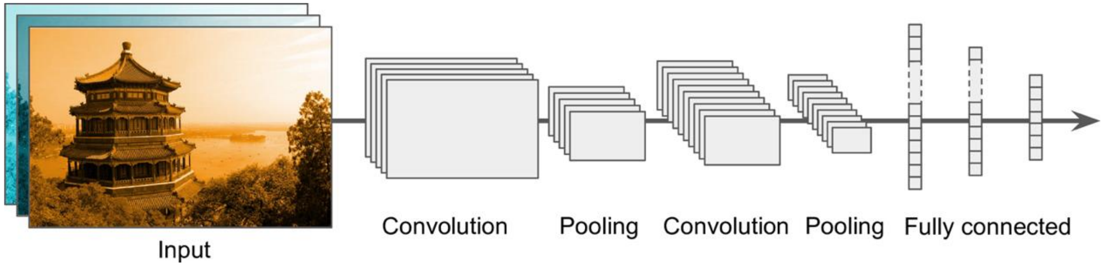


- Usually added after **convolution** and non-linearity (i.e. activation like ReLU) have been applied
- take the **average** (average-pooling) or **maximum activity** (max pooling) to represent the whole area
- Filter size is smaller than original feature map
- Helps model invariance to small local translations

Training

CNNs compute the stacked sequence of layers

- usually ending with a Fully-Connected Layer
- The FCN is the same as a regular neural networks

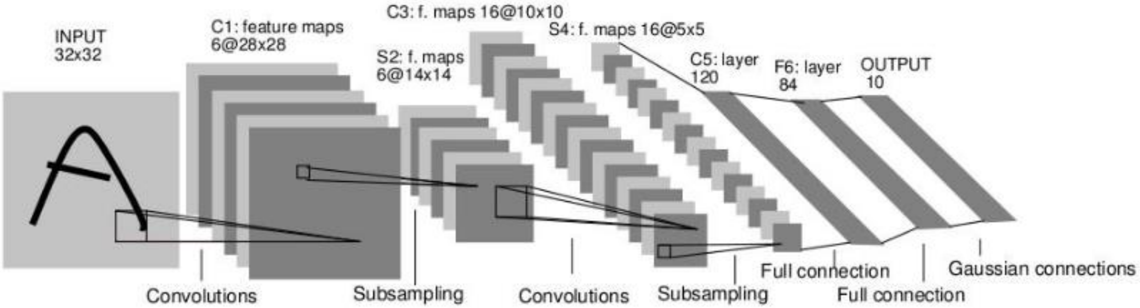


- Train with back-propagation

Some Well-known Architectures

- **LeNet5 (1990s)**
  - Early CNN used to read digits

LeNet 5

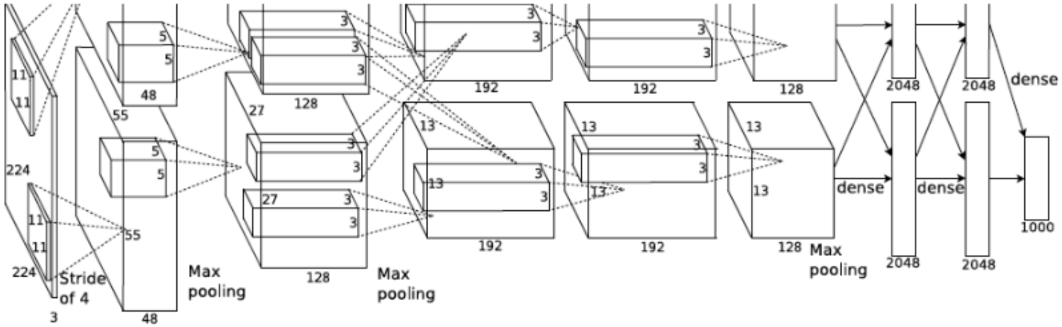


  - Y. LeCun et al., 1998
  - Average pooling, sigmoid, trained on MNIST

AlexNet

AlexNet (2012)

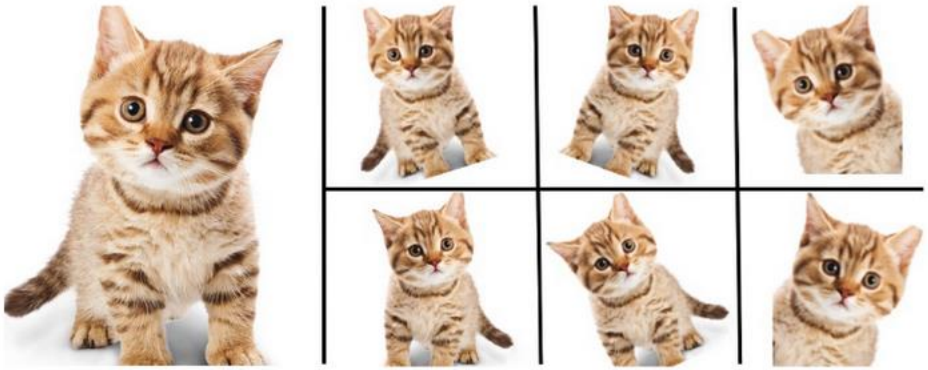
- Similar to LeNet but bigger and deeper model (8 layers, 60M params)



- ReLU activations, max pooling, dropout and data augmentation trained on GPUs on ImageNet
- Krizhevsky et al., 2012 (Imagenet 2012 winner)

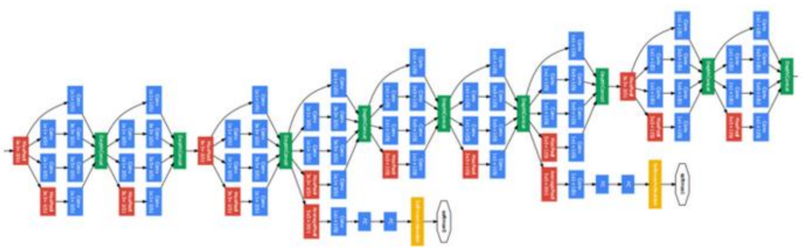
Data Augmentation

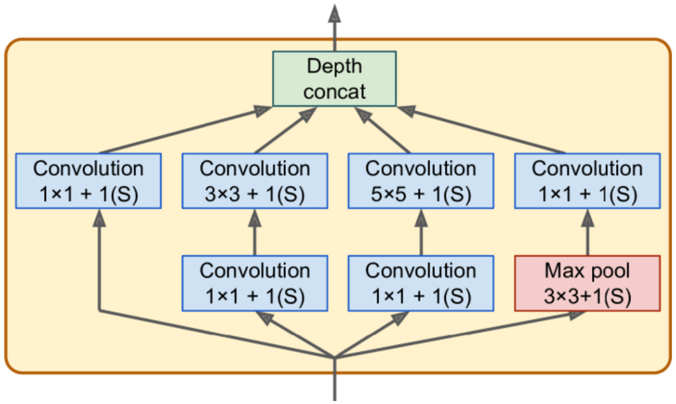
- **Useful technique for increasing training dataset size**
  - Apply rotations, shifts and re-sizing to make as many realistic training images as possible
  - Helps in training and to reduce overfitting



GoogleNet

- Szegedy et al., 2014
  - Much deeper than previous CNNs
  - **Inception modules**

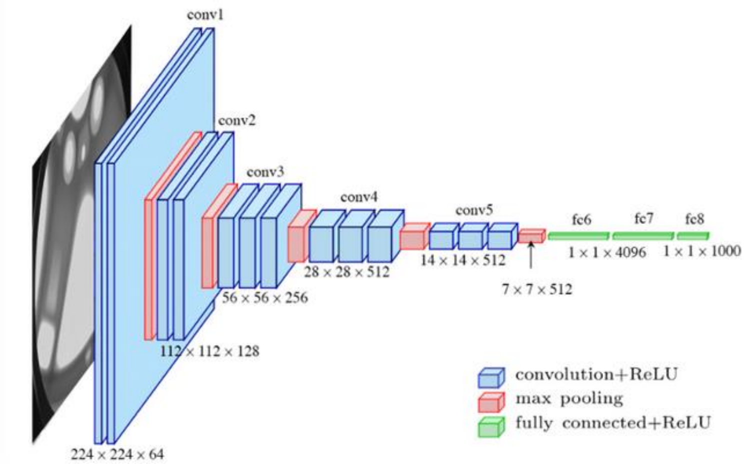




  - Multiple kernels stacked at same level
    - Concatenated along the depth dimension
    - Serve to capture information along the depth dimension across scales, bottlenecks to reduce dimensionality and behave like multi-dimensional layers

VGGNet

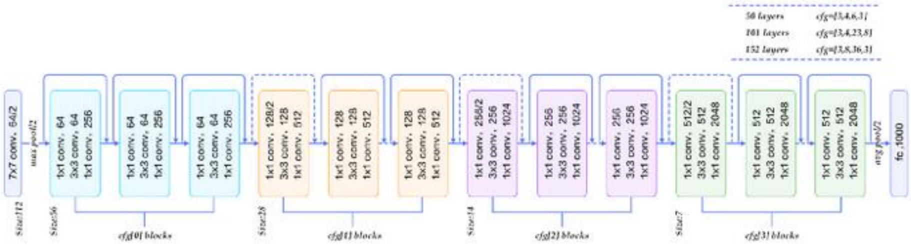
- Simonyan and Zisserman, 2014
  - Stacked smaller kernel-sized filters (3x3)

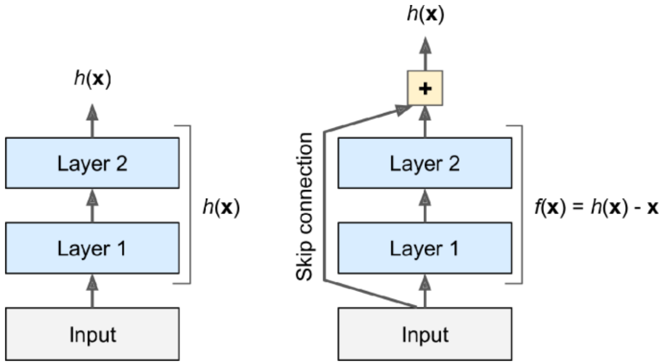


  - 16 layers: 2/3 convolutional, 1 max pool and repeat

ResNet

- He et al., 2015 (Imagenet 2015 winner)
  - Residual network with skip connections





  - 152 layers similar to VGG with skip connections (gated units) and batch normalization
  - Residual learning: h(x) – x
  - Helps propagate your signal across the whole network

Outline

- **Sequential Data**
- **Recurrent Neural Networks**

Sequential Data

Text


The New York Times

In Germany, cool politics as resistance

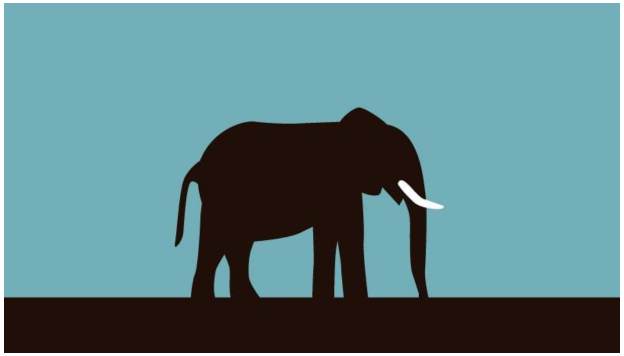
'Brexit' poses risk to hard-won Irish peace

Concerns over ethical ivory

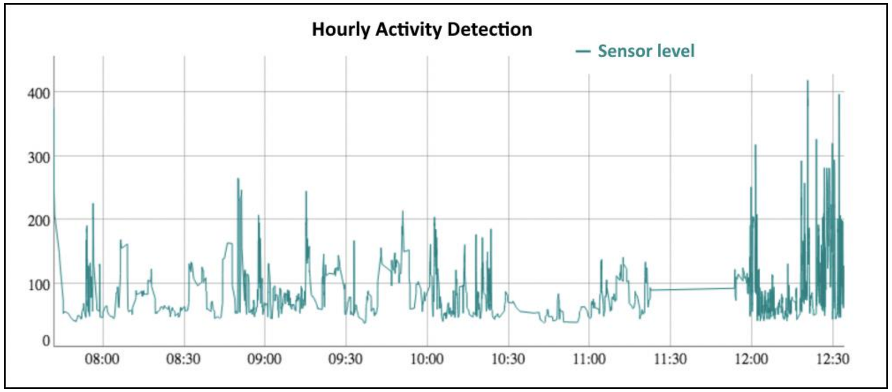
To build his label, a designer heads to Italy

INTERNATIONAL LUXURY CONFERENCE NOVEMBER 13-14, 2017 BRUSSELS WHAT'S NEXT: LUXURY IN A TURBULENT WORLD

Image



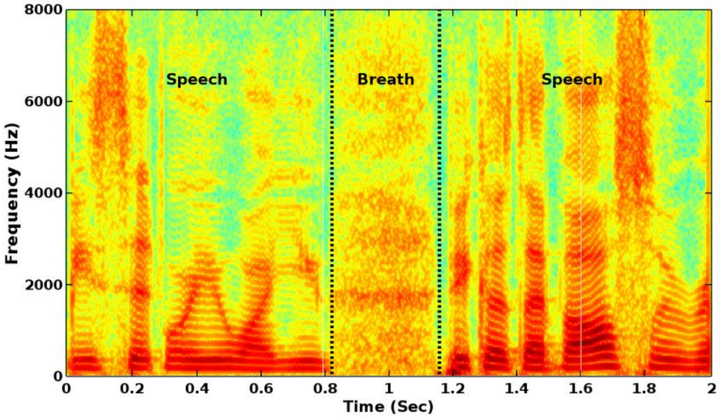
Time Series



Sequential Data

- **Properties:**
  - Elements occur in a particular order
  - May depend on other elements

- **Examples:**
  - Sentences
  - Images
  - Radio Waves
  - Temperature



Some Applications

- **Input:**
  - Fixed size
- **Output**
  - Sequence

The man in grey swings a bat
while the man in black looks on.

- **Example: image captioning**

Some Applications

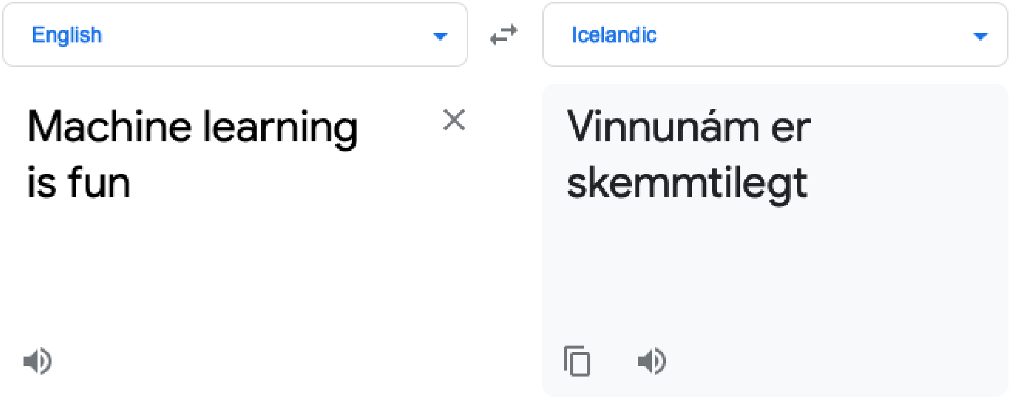
- **Input:**
  - Sequence
- **Output**
  - Fixed Size

- **Example: Sentiment Analysis**

Customer Feedback Text	Sentiment
"This café is great, the staff are really friendly and the coffee is delicious"	Positive
"I would not recommend this café to anyone. Their coffee is terrible and is really expensive"	Negative

Some Applications

- **Input:**
  - Sequence
- **Output**
  - Sequence



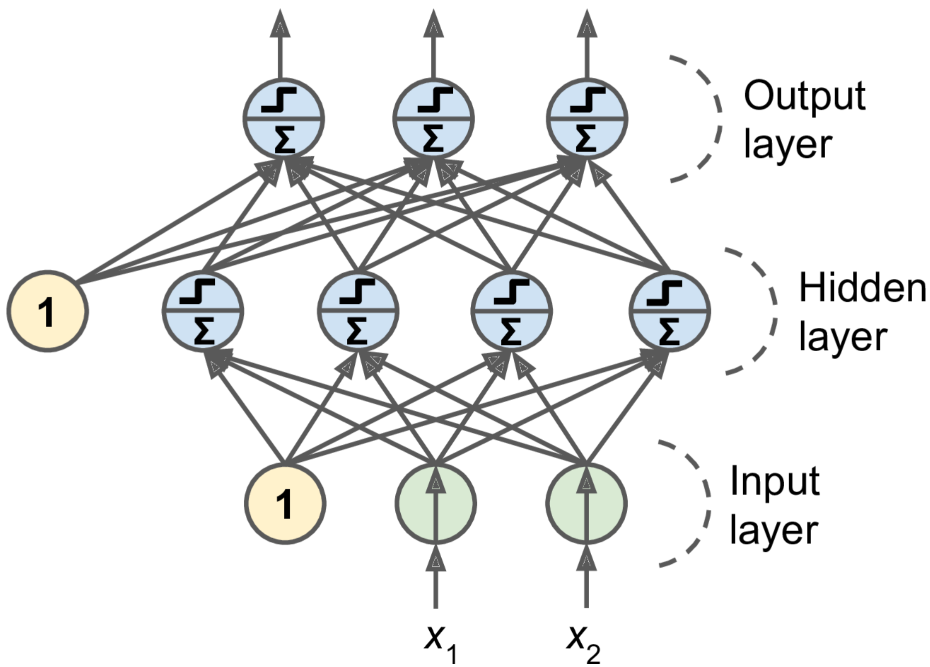
- **Example: Google Translate**

Recurrent Neural Networks

- **Extensions of deep neural networks to directed graphs and sequences**
  - Rumelhalt, Hinton, Williams (1986)
  - Dynamic behavior in the time domain
  - Introduce ideas of memory, feedback loops to accommodate sequential data
  - Key idea: capture information from **the past** in a **hidden state**

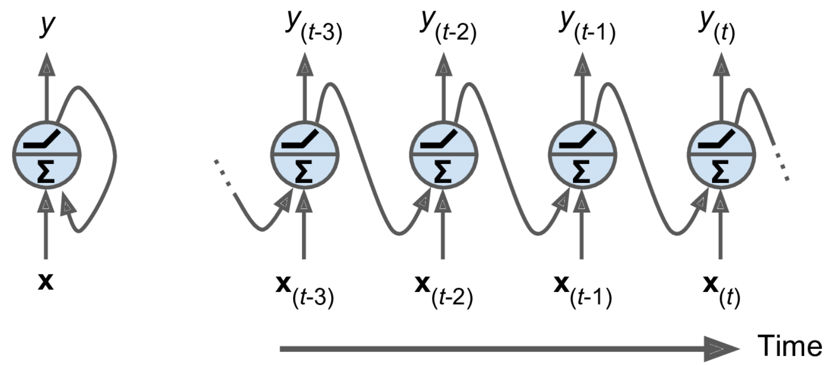
RNN vs MLP

MLP

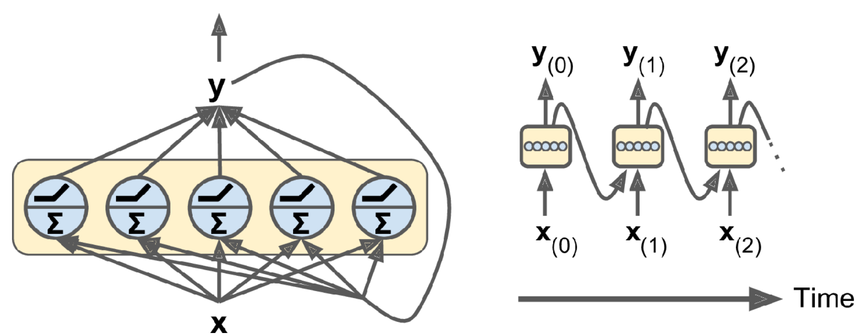


No loops

RNN neuron (unrolled)



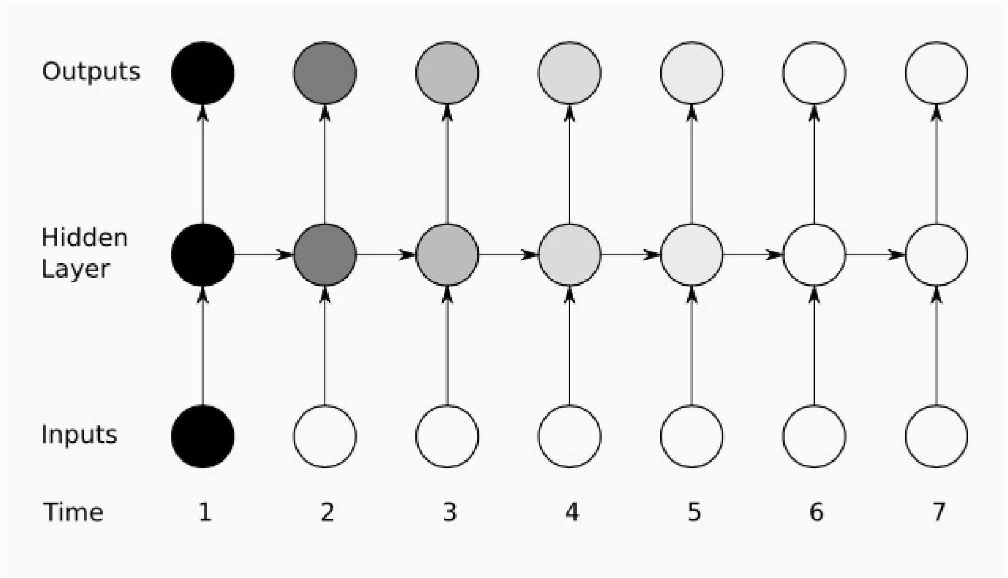
RNN layer (unrolled)



Basic RNN

Advantages:

- Weights are shared across layers
- Uses previous hidden state
  - Weights of each layer are not learned independently



- A form of “memory”
- Train with backpropagation (through time)