Machine Learning and Artificial Intelligence: Part 2

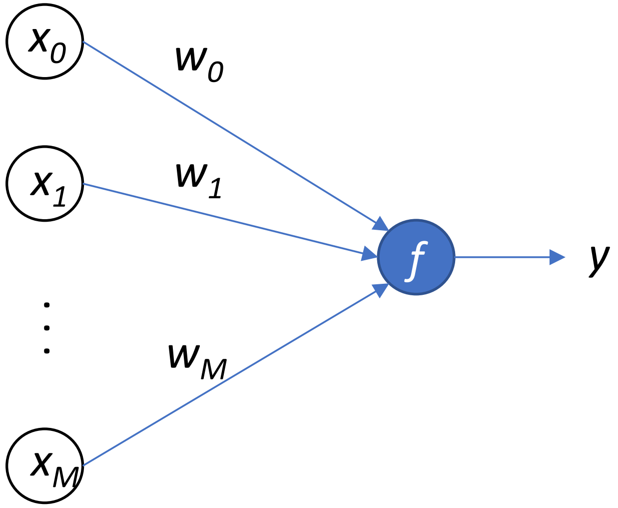
Neural Networks and Deep Learning

George Bezerra
Director of Data Science, TripAdvisor

SFI Complex Systems Summer School 2018

Logistic regression revisited

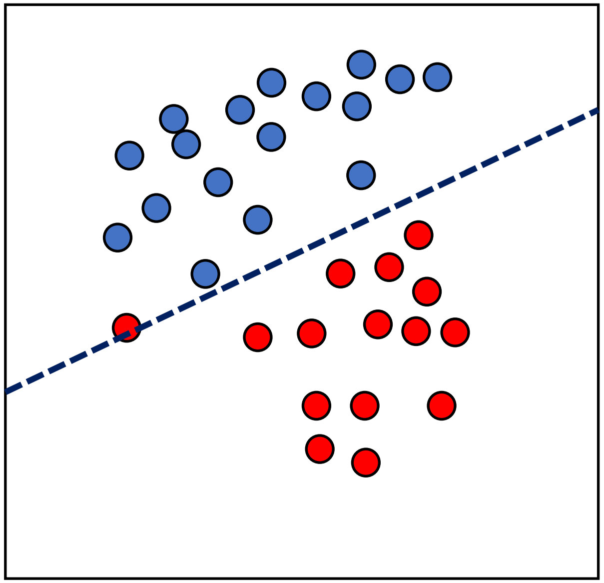
- Graphical representation:



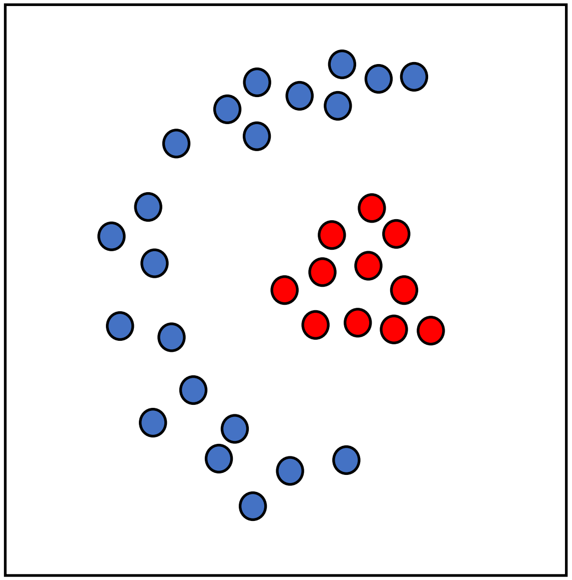
$$f(\boldsymbol{x}, \boldsymbol{w}) = \frac{1}{1 + e^{-\left(\sum_{j=0}^{M} w_j x_j\right)}}$$

The LR decision boundary is **linear**

Linear separation

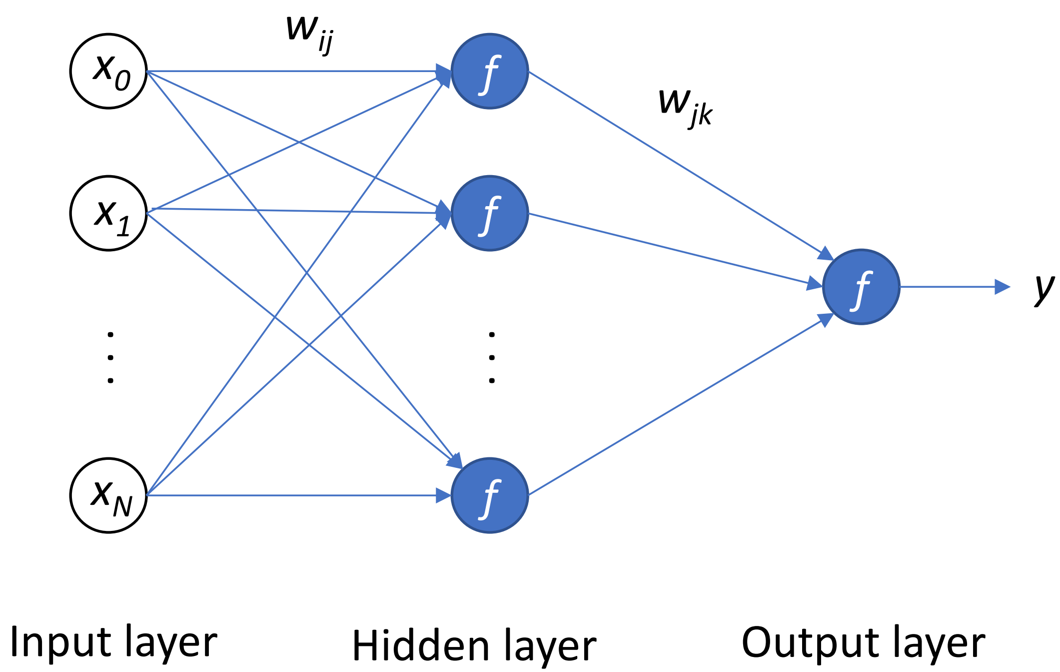


Nonlinear separation



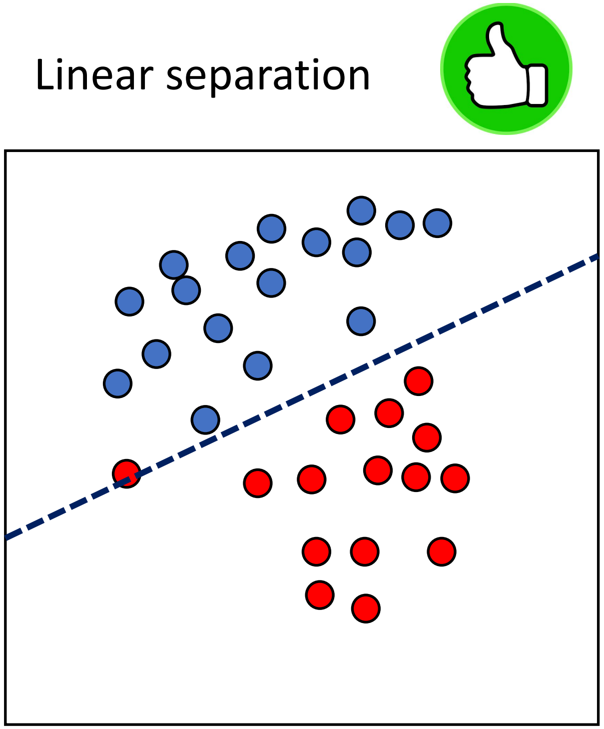
Multi-layer perceptron (MLP)

- Stacking logistic regressions together

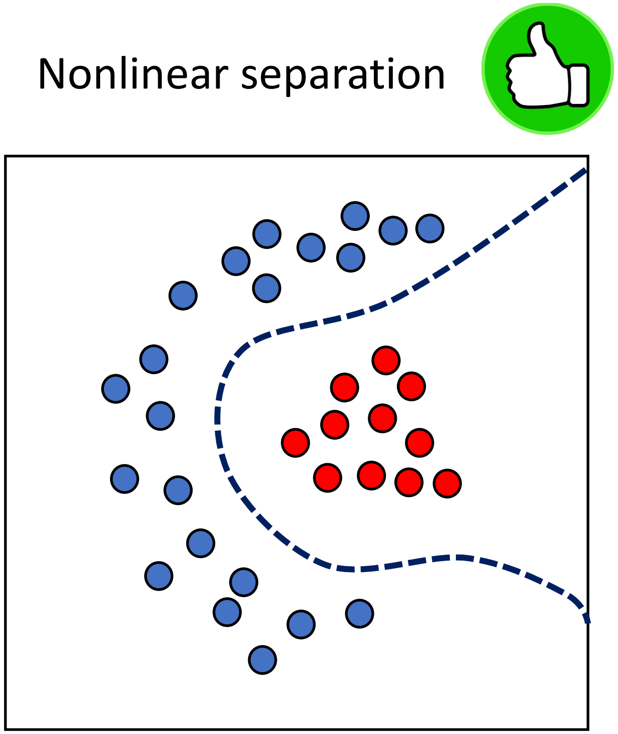


The MLP decision boundary is **nonlinear**

Linear separation



Nonlinear separation



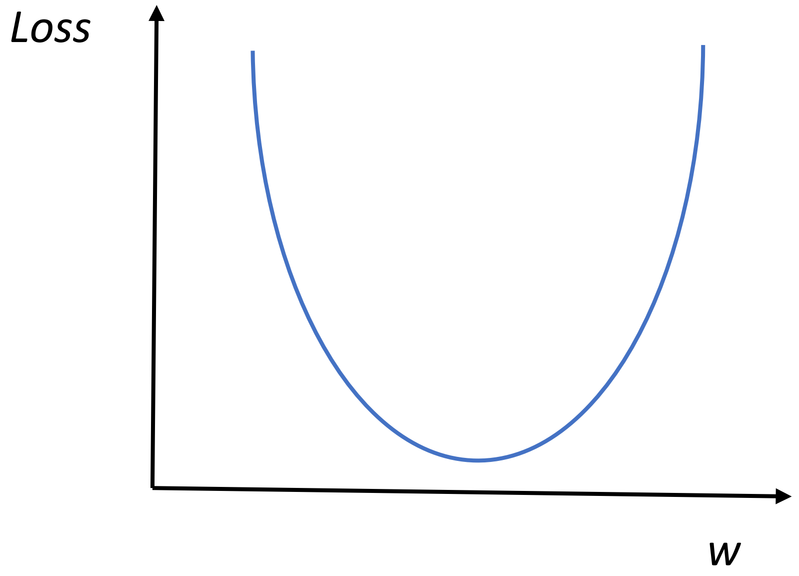
Gradient Descent

Loss function:

$$\mathcal{L}(\boldsymbol{w}) = \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}_i(\boldsymbol{w})$$

Iterative minimization:

$$\boldsymbol{w}(t+1) = \boldsymbol{w}(t) - \alpha\,\nabla\mathcal{L}(\boldsymbol{w}(t))$$



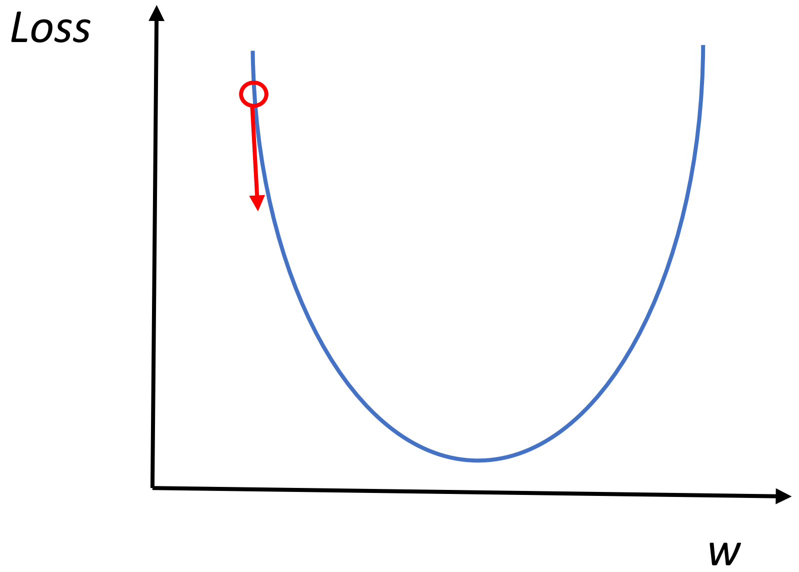
Gradient Descent

Loss function:

$$\mathcal{L}(\boldsymbol{w}) = \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}_i(\boldsymbol{w})$$

Iterative minimization:

$$\boldsymbol{w}(t+1) = \boldsymbol{w}(t) - \alpha\,\nabla\mathcal{L}(\boldsymbol{w}(t))$$



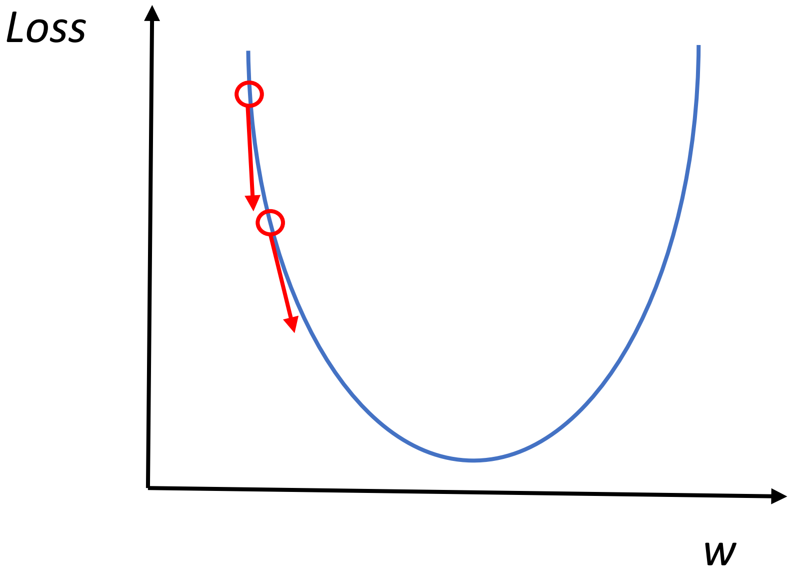
Gradient Descent

Loss function:

$$\mathcal{L}(\boldsymbol{w}) = \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}_i(\boldsymbol{w})$$

Iterative minimization:

$$\boldsymbol{w}(t+1) = \boldsymbol{w}(t) - \alpha\,\nabla\mathcal{L}(\boldsymbol{w}(t))$$



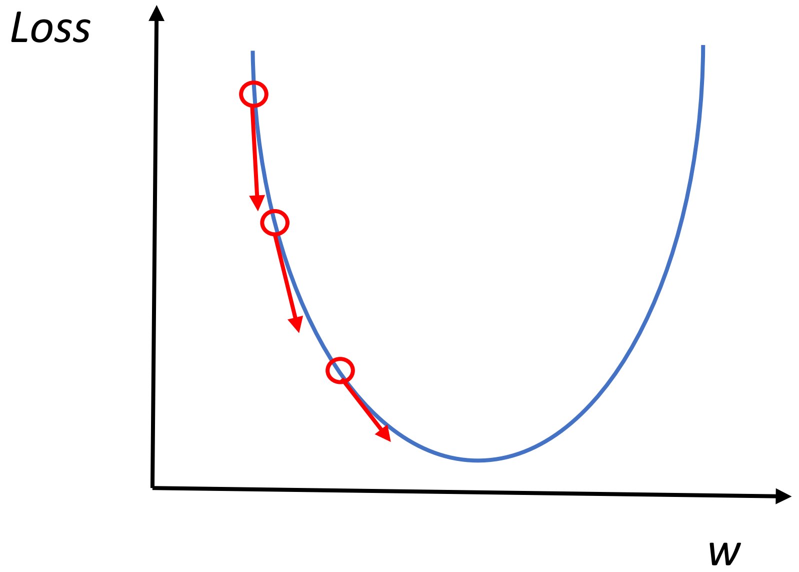
Gradient Descent

Loss function:

$$\mathcal{L}(\boldsymbol{w}) = \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}_i(\boldsymbol{w})$$

Iterative minimization:

$$\boldsymbol{w}(t+1) = \boldsymbol{w}(t) - \alpha\,\nabla\mathcal{L}(\boldsymbol{w}(t))$$



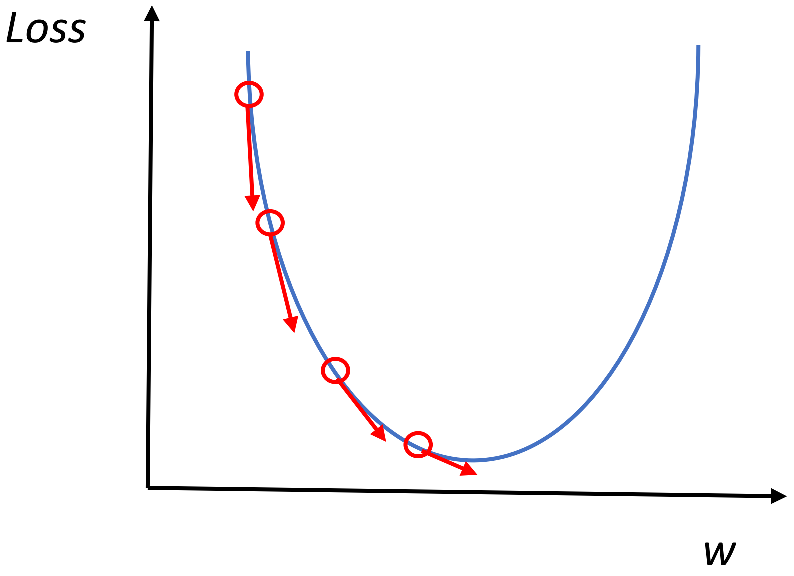
Gradient Descent

Loss function:

$$\mathcal{L}(\boldsymbol{w}) = \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}_i(\boldsymbol{w})$$

Iterative minimization:

$$\boldsymbol{w}(t+1) = \boldsymbol{w}(t) - \alpha\,\nabla\mathcal{L}(\boldsymbol{w}(t))$$



Gradient Descent

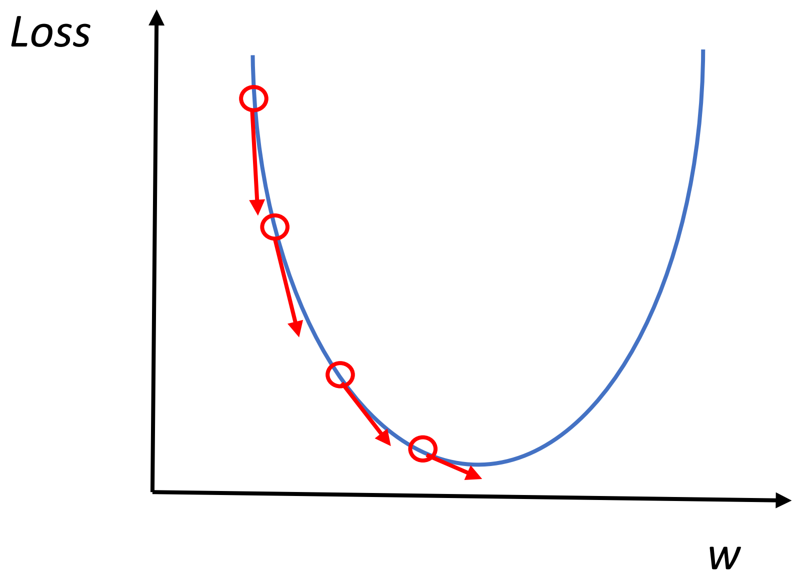
Loss function:

$$\mathcal{L}(\boldsymbol{w}) = \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}_i(\boldsymbol{w})$$

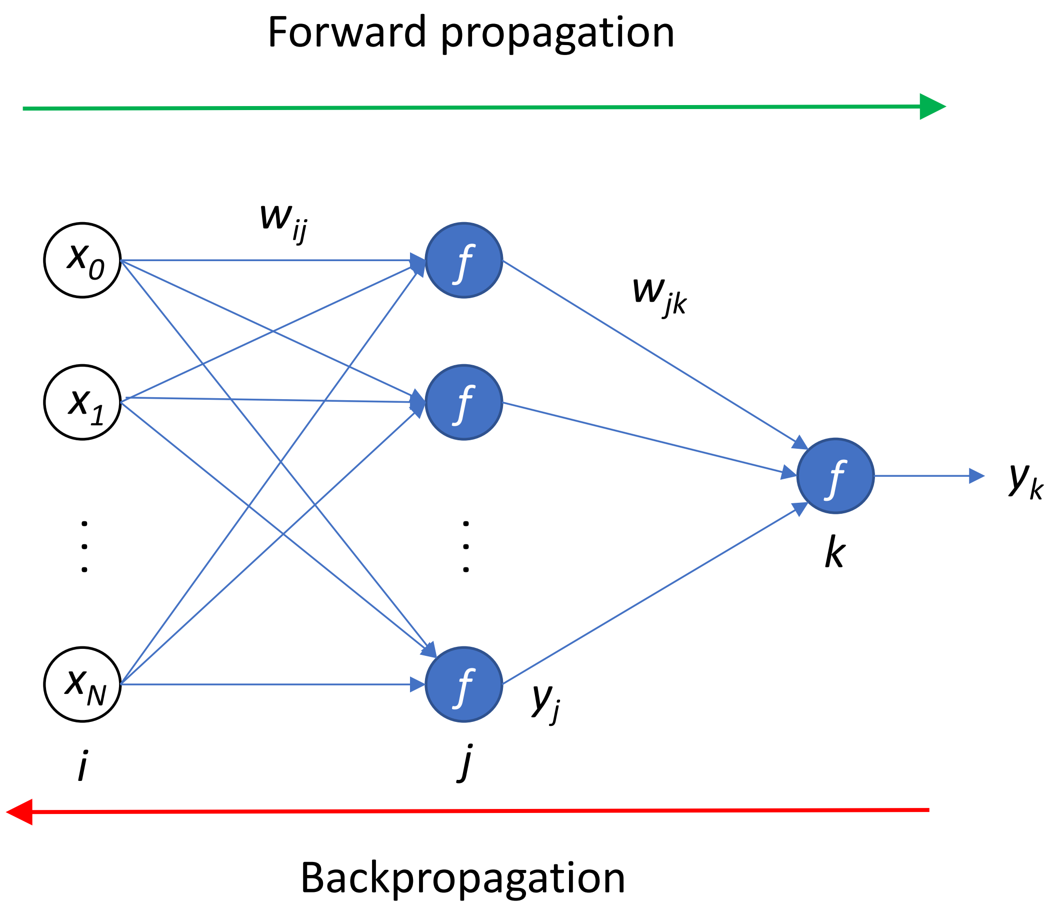
Iterative minimization:

$$\boldsymbol{w}(t+1) = \boldsymbol{w}(t) - \alpha\,\nabla\mathcal{L}(\boldsymbol{w}(t))$$

Learning rate



Backpropagation algorithm

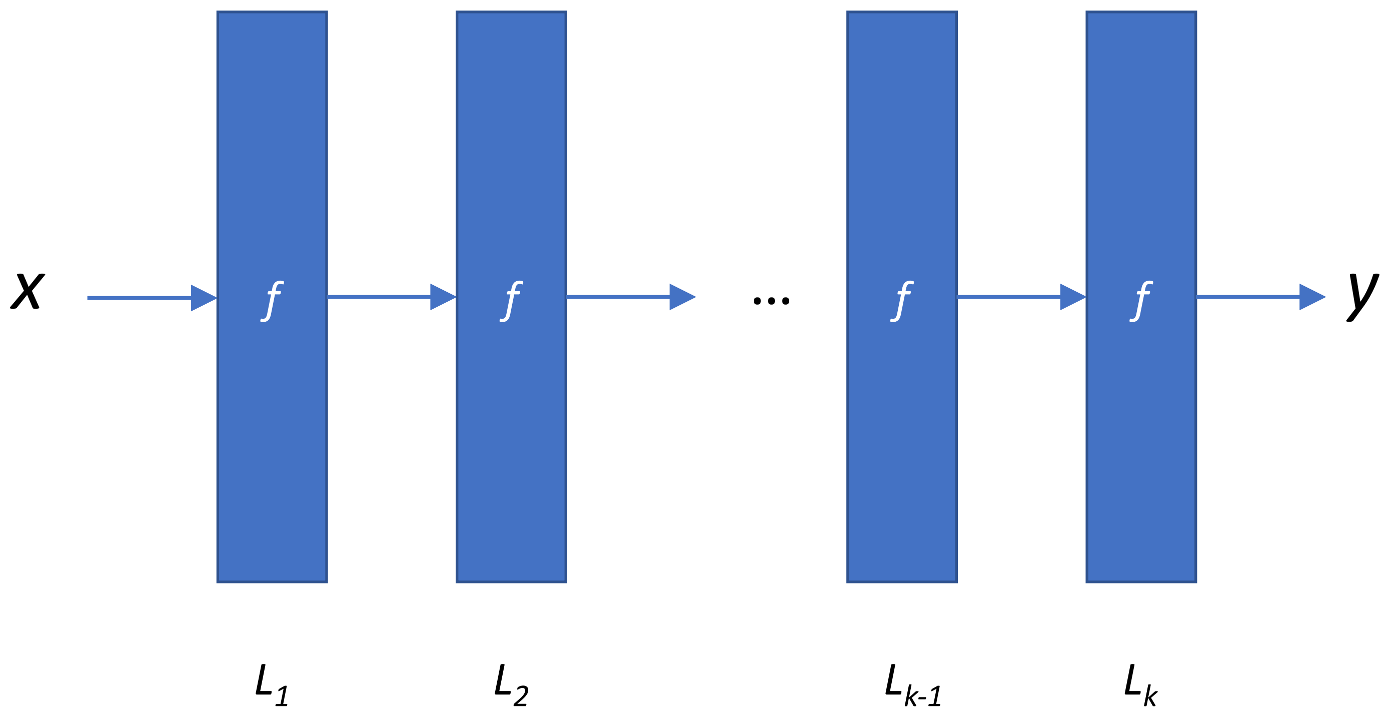


<u>Chain rule:</u>

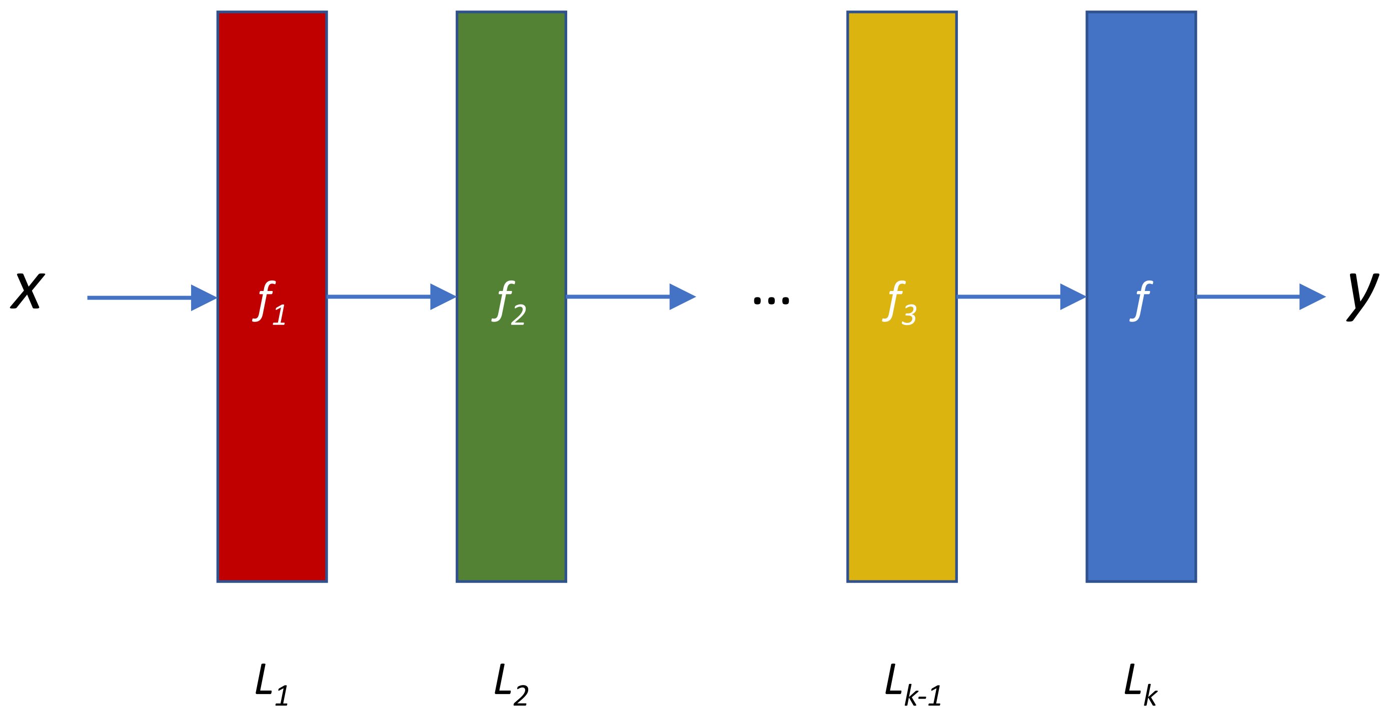
$$\frac{\partial \mathcal{L}}{\partial w_{jk}} = \frac{\partial \mathcal{L}}{\partial y_k} \cdot \frac{\partial y_k}{\partial w_{jk}}$$

$$\frac{\partial \mathcal{L}}{\partial w_{ij}} = \frac{\partial \mathcal{L}}{\partial y_k} \cdot \frac{\partial y_k}{\partial y_j} \cdot \frac{\partial y_j}{\partial w_{ij}}$$

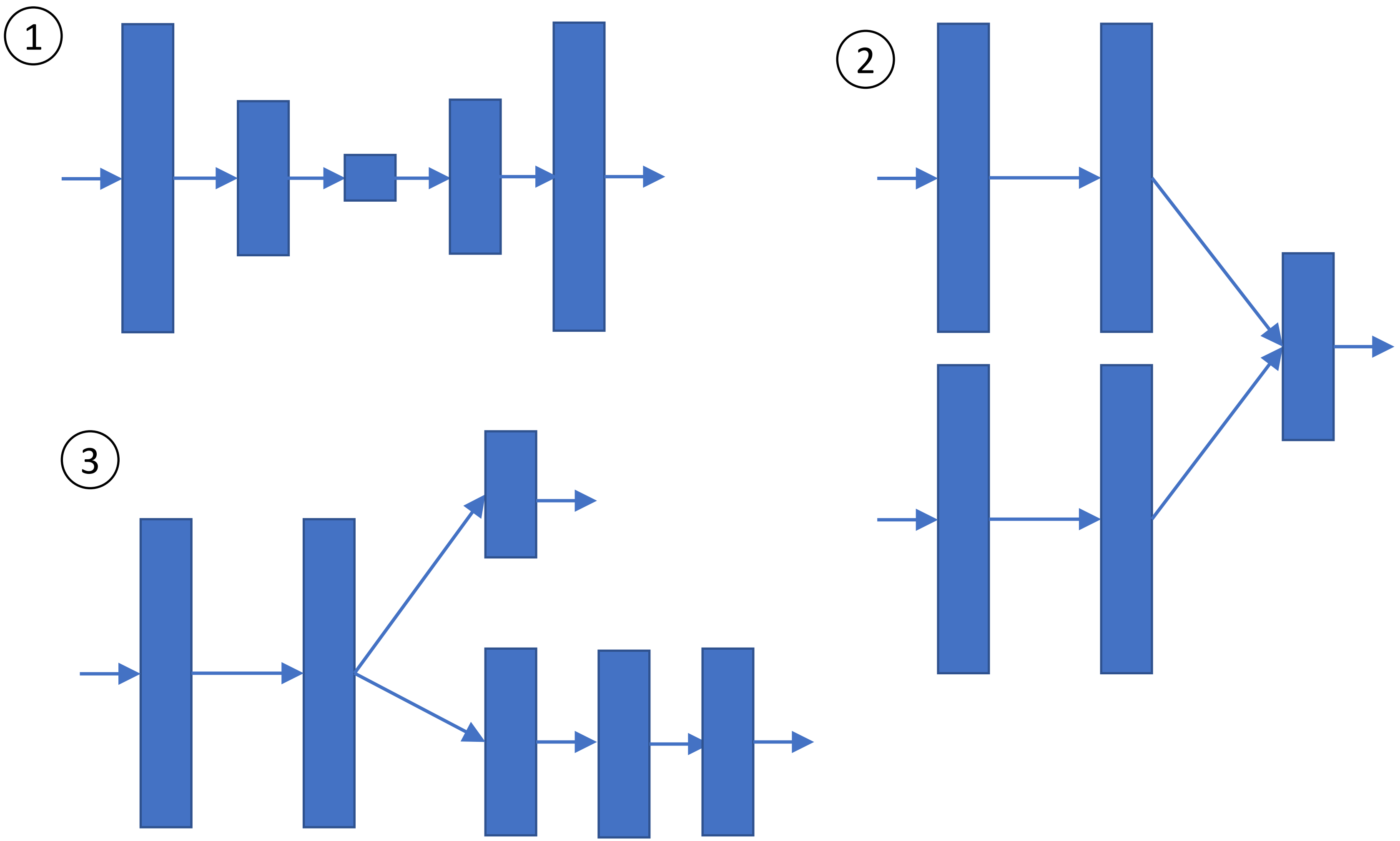
Deep learning



Deep learning



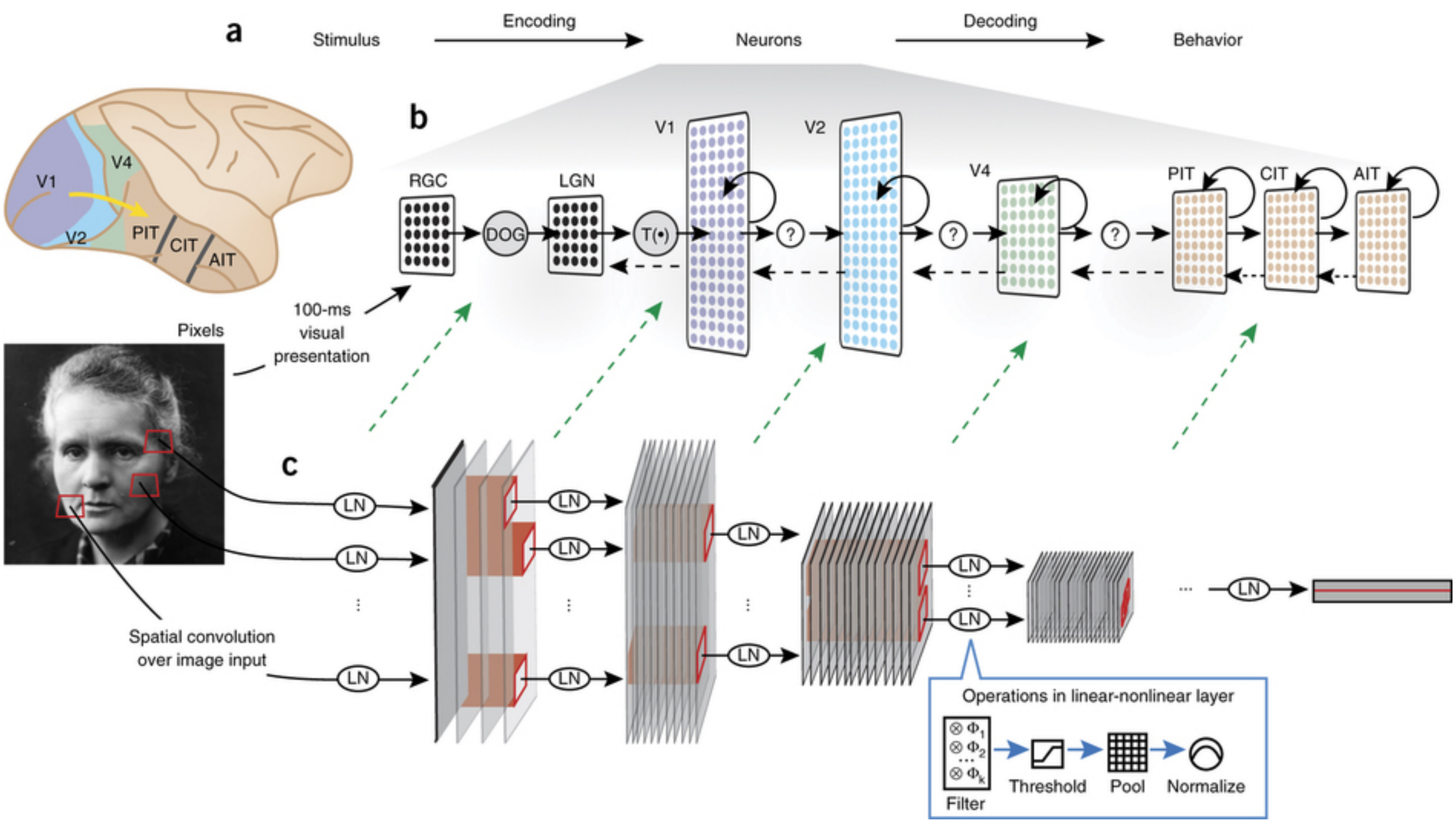
Deep learning



Why deep learning now?

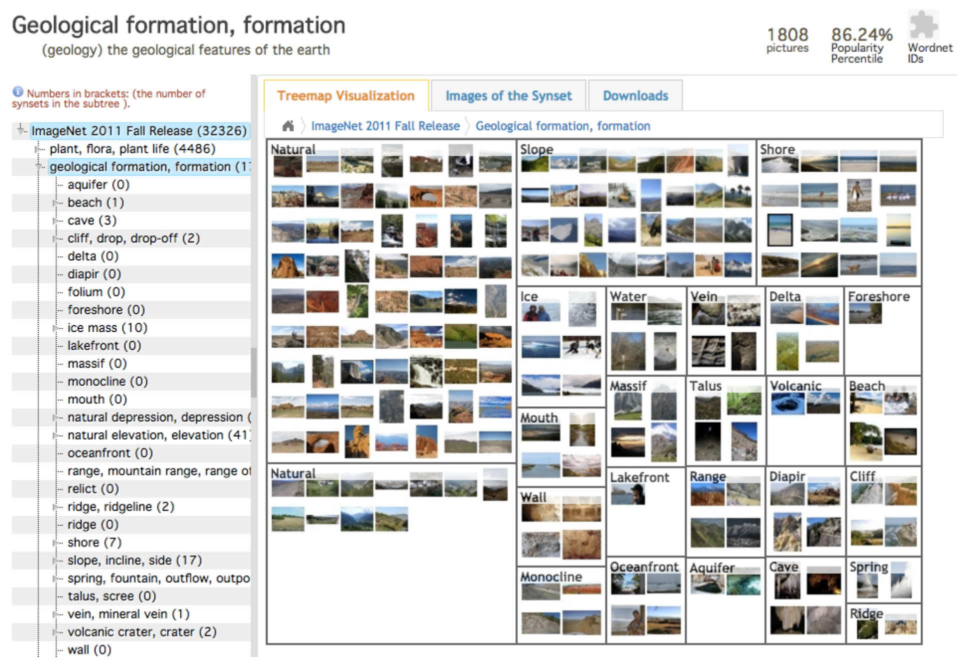
- Massive datasets
- More compute power (Faster CPUs + GPUs)
- Better understanding of regularization techniques
- Open source deep learning frameworks: PyTorch, Keras, Tensorflow, Theano, MXNet, Caffe, etc.

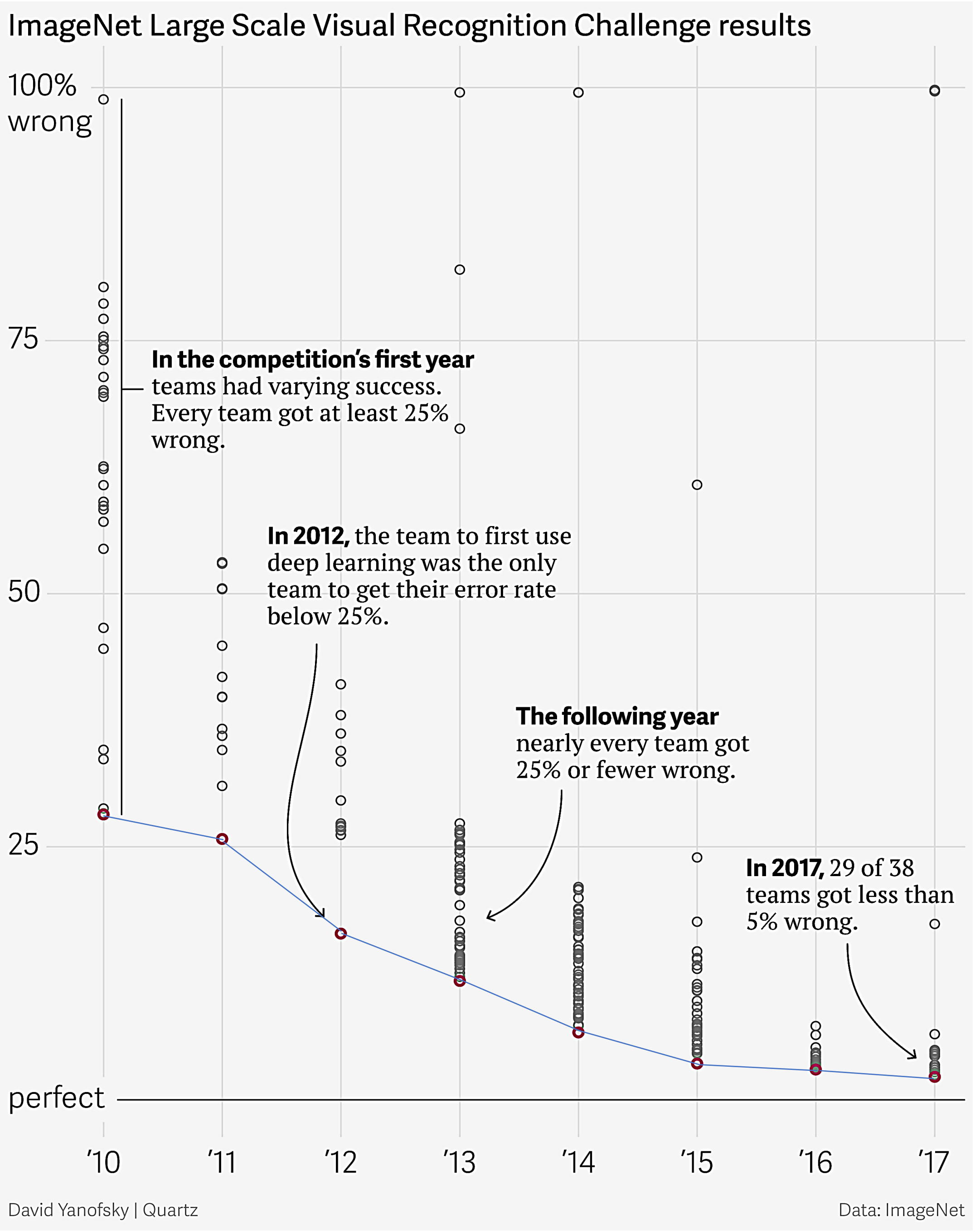
Convolutional Neural Networks



Imagenet Challenge

- 1M images
- 1000 categories





ImageNet Large Scale Visual Recognition Challenge results
100% wrong
75
50
25
perfect
'10
'11
'12
'13
'14
'15
'16
'17
In the competition's first year teams had varying success. Every team got at least 25% wrong.
In 2012, the team to first use deep learning was the only team to get their error rate below 25%.
The following year nearly every team got 25% or fewer wrong.
In 2017, 29 of 38 teams got less than 5% wrong.
David Yanofsky | Quartz
Data: ImageNet

Convolution

Image (binary)

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

1	0	1
0	1	0
1	0	1

Neuron
(convolutional filter)

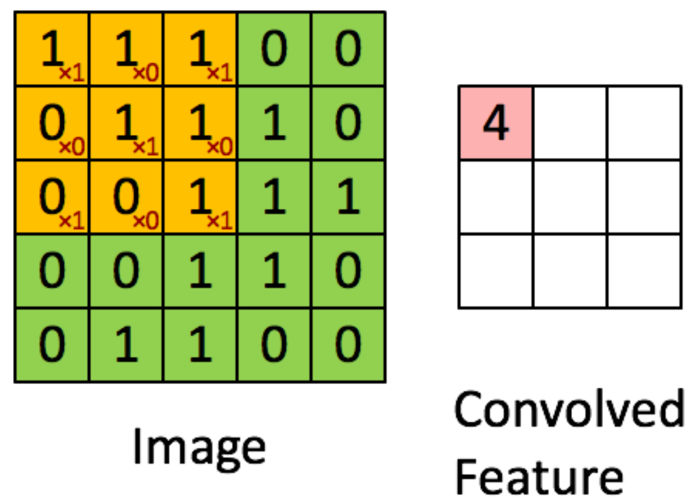


Image source:
http://deeplearning.stanford.edu/wiki/index.php/Feature_extraction_using_convolution

AlexNet

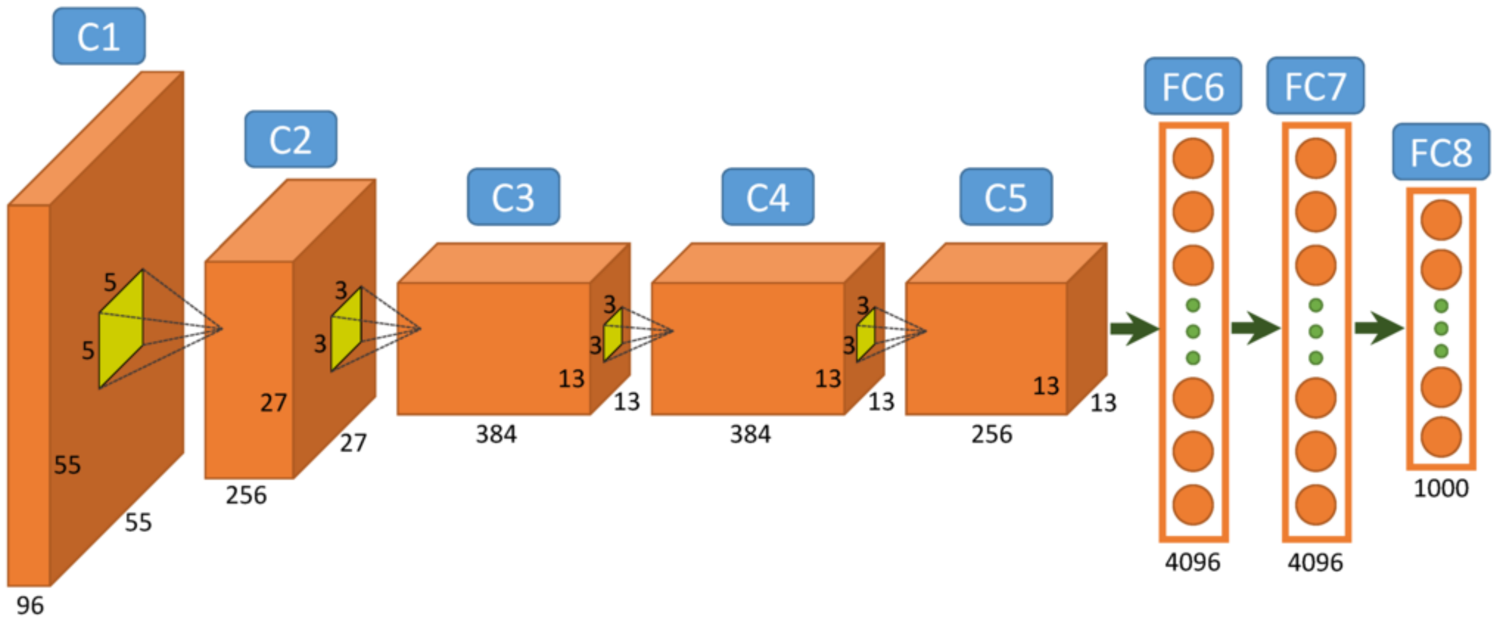
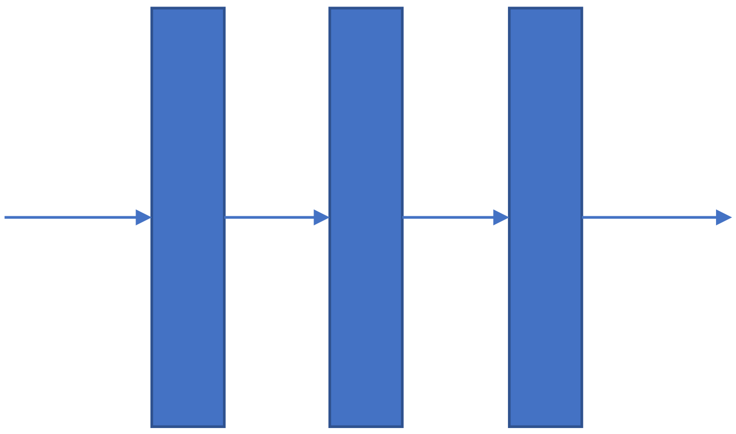


Image source: https://www.saagie.com/blog/object-detection-part1

Recurrent Neural Networks

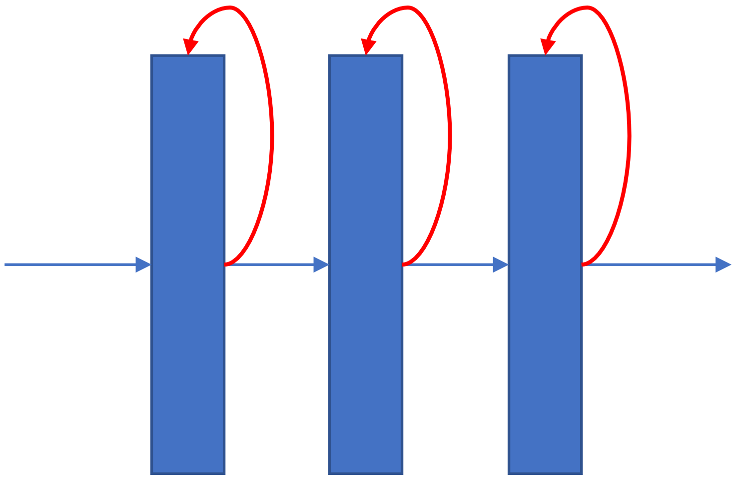
Feed-forward network

$$\boldsymbol{y} = f(W\boldsymbol{x})$$



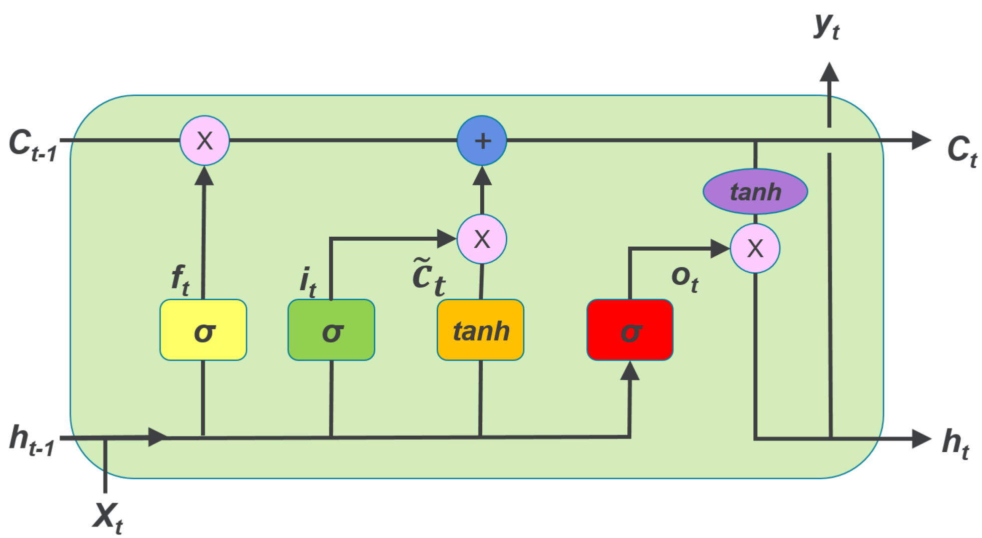
Recurrent network

$$\boldsymbol{y}_t = f(W\boldsymbol{x}_t + U\boldsymbol{y}_{t-1})$$



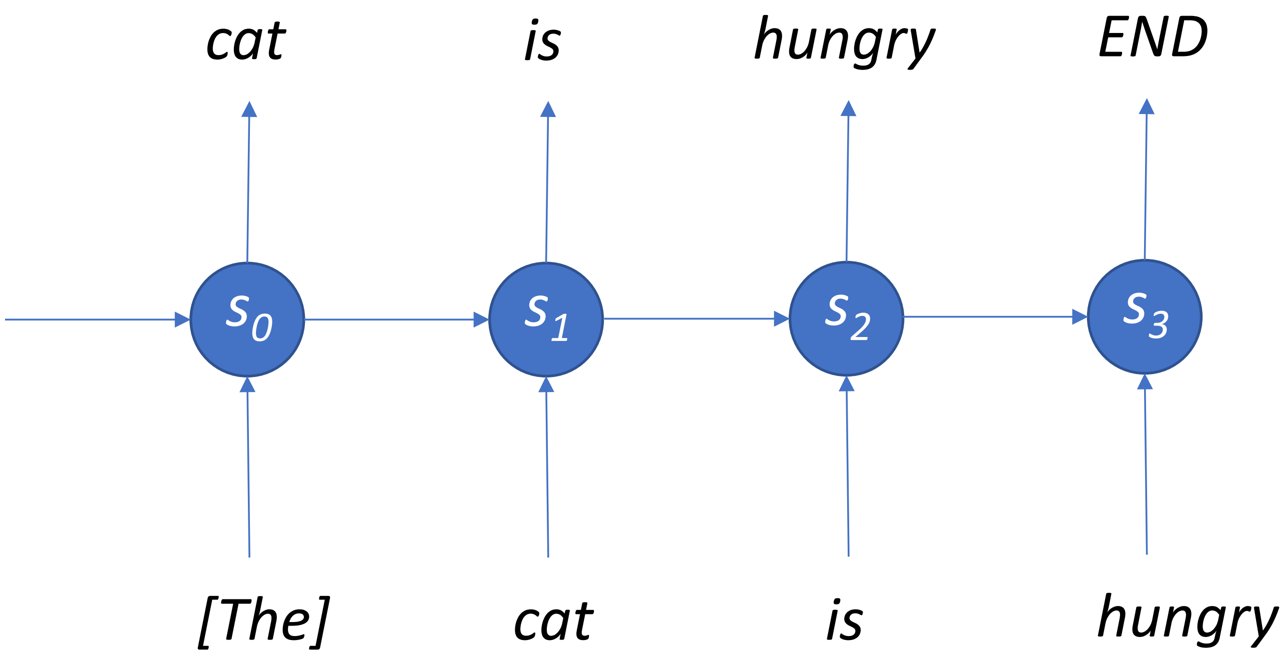
Long-short term memory (LSTM)

$$
\begin{aligned}
f_t &= \sigma_g(W_f x_t + U_f h_{t-1} + b_f) \\
i_t &= \sigma_g(W_i x_t + U_i h_{t-1} + b_i) \\
o_t &= \sigma_g(W_o x_t + U_o h_{t-1} + b_o) \\
c_t &= f_t \circ c_{t-1} + i_t \circ \sigma_c(W_c x_t + U_c h_{t-1} + b_c) \\
h_t &= o_t \circ \sigma_h(c_t)
\end{aligned}
$$



Text generation with LSTMs

- Language model: $P(w_1, \dots, w_m) = \prod_{i=1}^{m} P(w_i | w_1, \dots, w_{i-1})$



Text generation examples

[Shakespeare:](http://karpathy.github.io/2015/05/21/rnn-effectiveness/)

[KING LEAR:] O, if you were a feeble sight, the courtesy of your law, Your sight and several breath, will wear the gods With his heads, and my hands are wonder'd at the deeds, So drop upon your lordship's head, and your opinion Shall be against your honour.

(Source: http://karpathy.github.io/2015/05/21/rnn-effectiveness/)

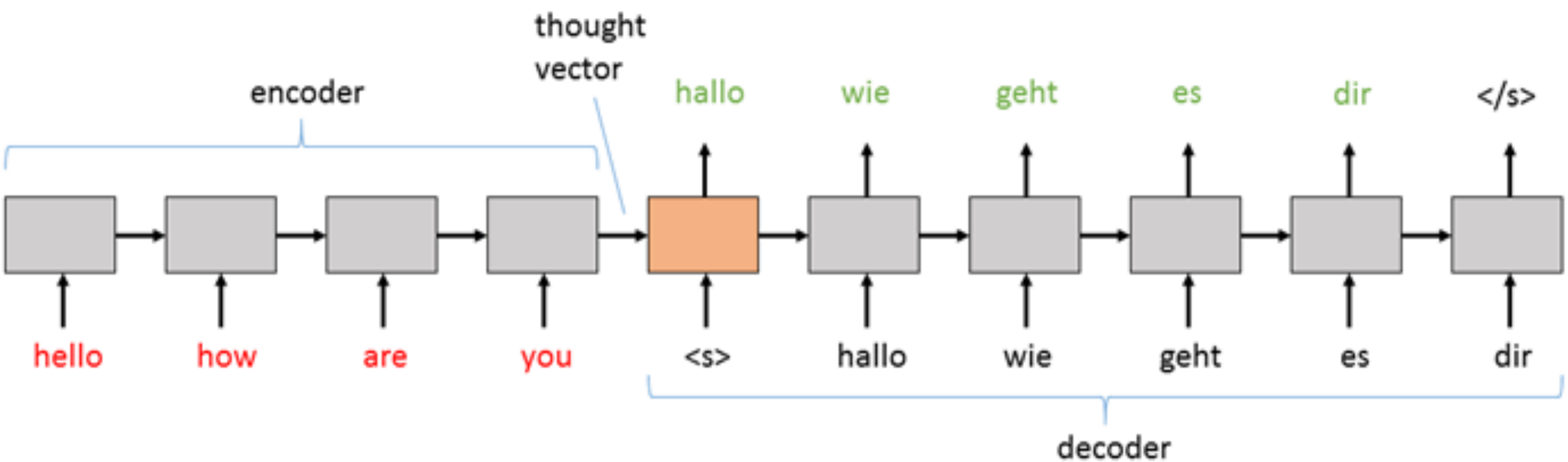
[Trump tweet bot:](https://towardsdatascience.com/yet-another-text-generation-project-5cfb59b26255)

[I, Donald J. Trump, president of the United]States of The Apprentice and the world in the world that we have to be a great trade deal and the people of the World Trade Center in the world that we have to be a great person to be a great problem.

(Source: https://towardsdatascience.com/yet-another-text-generation-project-5cfb59b26255)

Neural Machine Translation

- Sequence-to-sequence models

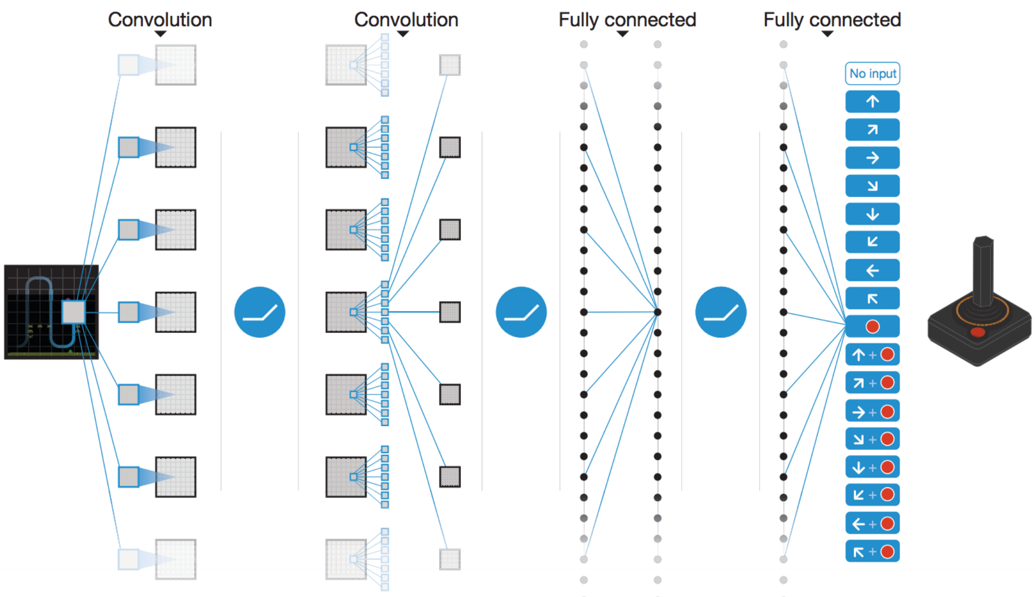


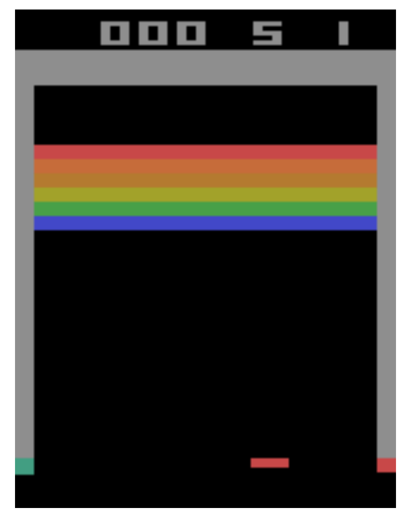
(Image source: https://cntk.ai/pythondocs/CNTK_204_Sequence_To_Sequence.html)

Deep reinforcement learning

- Deep Q-learning (Google Deepmind)

Network structure

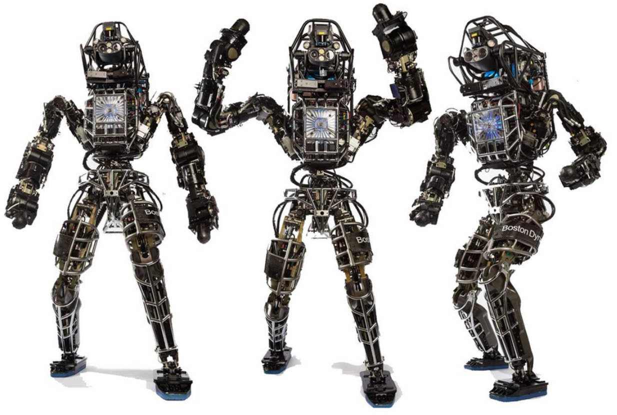


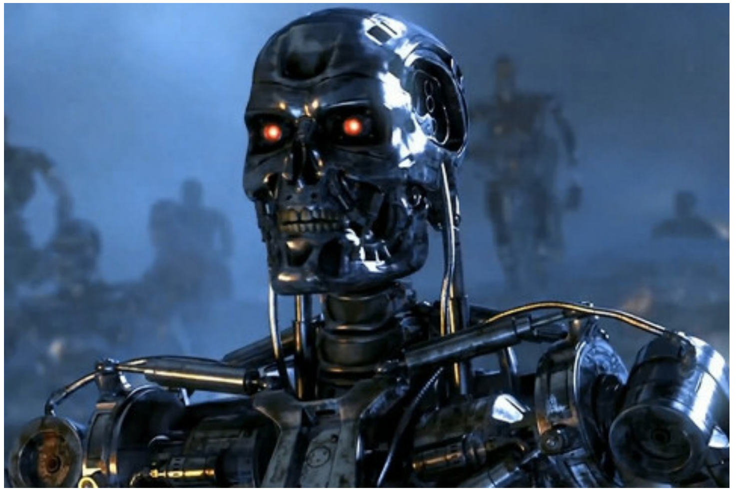


Atari game

(Image source: https://becominghuman.ai/lets-build-an-atari-ai-part-1-dqn-df57e8ff3b26)

Killer robots!





Boston Dynamics

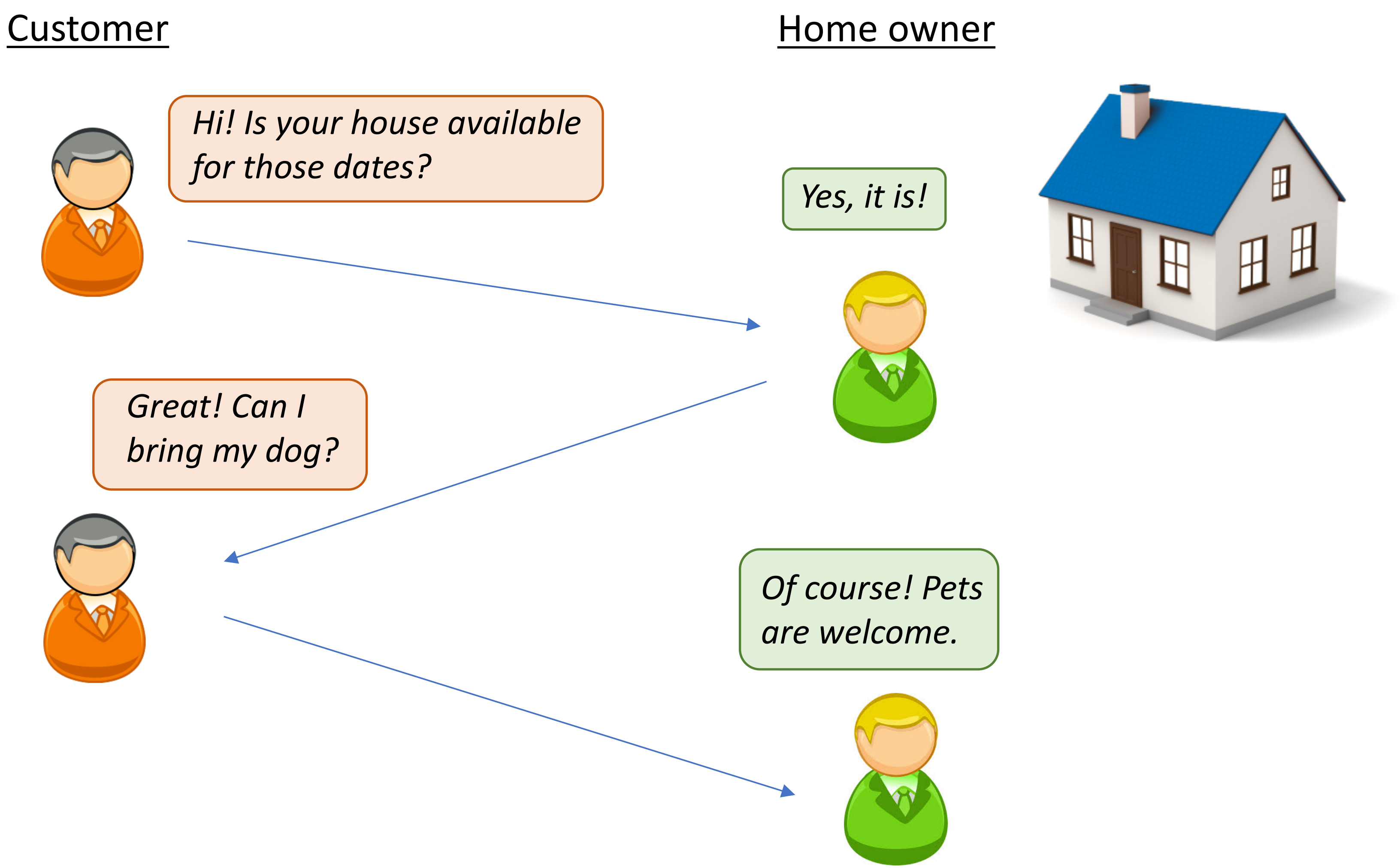
The real danger of "AI"

- <u>Algorithmic bias:</u>

  "when a computer system behaves in ways that reflects the implicit values of humans involved in that data collection, selection, or use." – Wikipedia

- Increased danger with widespread automation

Short-term home rentals business



Fraud detection with deep learning

Customer: “Hi, I’d like to stay at you place, it’s perfect! Do you accept pets?”

Owner 1: “We’d love to have you stay with us. Yes, we do accept pets.”

Fraud detection with deep learning

Customer: “Hi, I’d like to stay at you place, it’s perfect! Do you accept pets?”

Owner 1: “We’d love to have you stay with us. Yes, we do accept pets.”

Owner 2: “Please contact me at badowner@gmail.com”

Fraud detection with deep learning

Customer: “Hi, I’d like to stay at you place, it’s perfect! Do you accept pets?”

Owner 1: “We’d love to have you stay with us. Yes, we do accept pets.”

Owner 2: “Please contact me at badowner@gmail.com”

Owner 3: “Find me at *****, I can make you a better deal.”

More examples

- “Search for ‘Paradise house’ on google and you’ll find our website.”
- “Call me at five five five two seven six three.”
- “Send a message to janelle at hotmail dot com”
- “please c-o-n-t-a-c-t me on mmartin-a-t-g-m-a-i-l”

Naïve approach: Regular expressions

- Regular expressions for phone numbers, websites, email, keywords, etc.
  - *Websites*: (?:(?:(?:https?|ftps?)://)|(?:www?|m|ftp)\.)(?:[a-z0-9-]+\.)+[a-z]+(?::[0-9]+)?(?:(/|\?).*)?
  - *Emails*: (?:[a-z0-9._%+-]| +dot +)+(?: *@ *| +at +)(?:[a-z0-9.-]| +dot +)+(?:\.| +dot +)[a-z]{2,6}
- Over 2,000 regexes
- Manual process
- Generates many false positives

Using a model: bag-of-words + regex

- Logistic regression classifier using bag of words
- Uses existing regular expressions as features (1 if there is a match, 0 otherwise)
- Reduces false positives by 2-4x!
- However…
  - Regular expressions are easy to fool
  - Cannot learn new patterns on its own (requires new regexes)
  - BOW misses character-level features
  - Does not capture context

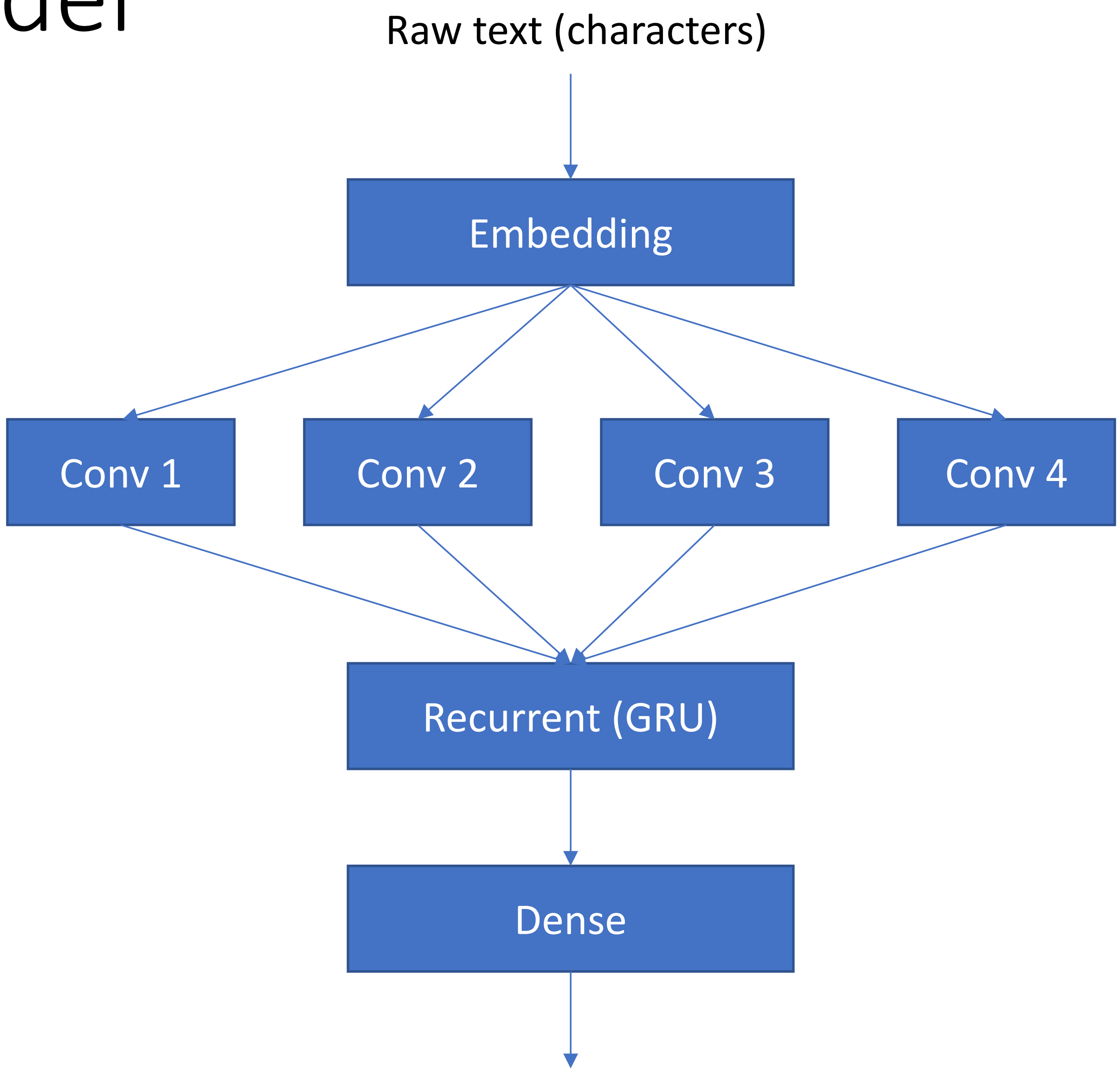
Examples where traditional approaches fail

- Please call me on *407*410*9301
- u can 91 reach me 49 on this 27 number anytime 86 you want 54. I hope you understand the message
- I can confirm the availability and the price. I kindly invite you to proceed with your booking. Please do not hesitate to contact me if you need further information. Thank you. (*False positive*)
- Please note the owner address contact him directly: ernesto＠gardener. ｃｏｍ
- you may r. ea ch us at नौ तीन नौ शून्य आठ सात आठ एक शून्य दो

Deep learning solution

- Learns the features (at the character level)
- Interprets contextual meaning (through recurrent layers)
- Captures high-level relationships and patterns
- Goals:
  - Eliminate the need for regular expressions
  - Improve accuracy
  - Catch more circumvention

Model


Raw text (characters)
Embedding
Conv 1
Conv 2
Conv 3
Conv 4
Recurrent (GRU)
Dense

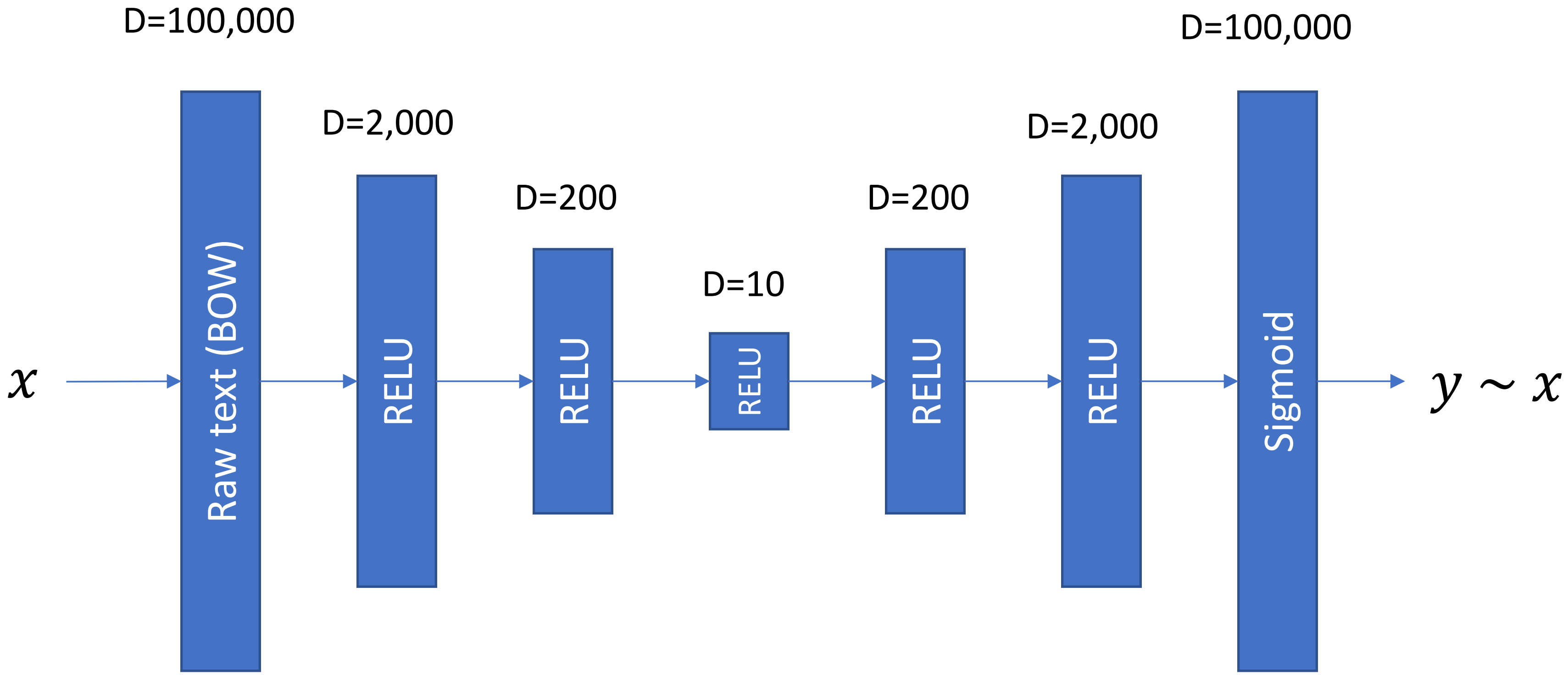
Results

- Please call me on *407*412*9391
  - Proba: 0.998
- u can 90 reach me 49 on this 17 number anytime 86 you want 54. I hope you understand the message
  - Proba: 0.710
- I can confirm the availability and the price. I kindly invite you to proceed with your booking. Please do not hesitate to contact me if you need further information. Thank you. (False positive)
  - Proba: 0.016
- Please note the owner address contact him directly: alvarez@ gardener. c o m
  - Proba: 0.97
- you may r. ea ch us at नौ नौ तीन शून्य आठ सात एक आठ शून्य दो
  - Proba: 0.99

Recap (part 2)

- Neural networks are powerful non-linear methods for solving supervised problems.
- Deep learning takes neural networks to the next level: more layers, activation functions, architectures
- Convolutional neural networks changed the game on image classification tasks.
- Recurrent neural networks enable a wide variety of natural language processing tasks.
- Killer robots are not the real danger of AI; algorithmic bias is.

Deep auto-encoder



AI as a complex system

- Backpropagation
- Non-linear dynamics
- Fractal (hierarchical-modular) network structure
- Domain-specific
- Emergence

Word Embeddings

- Finding dense representations for sparse high-dimensional data

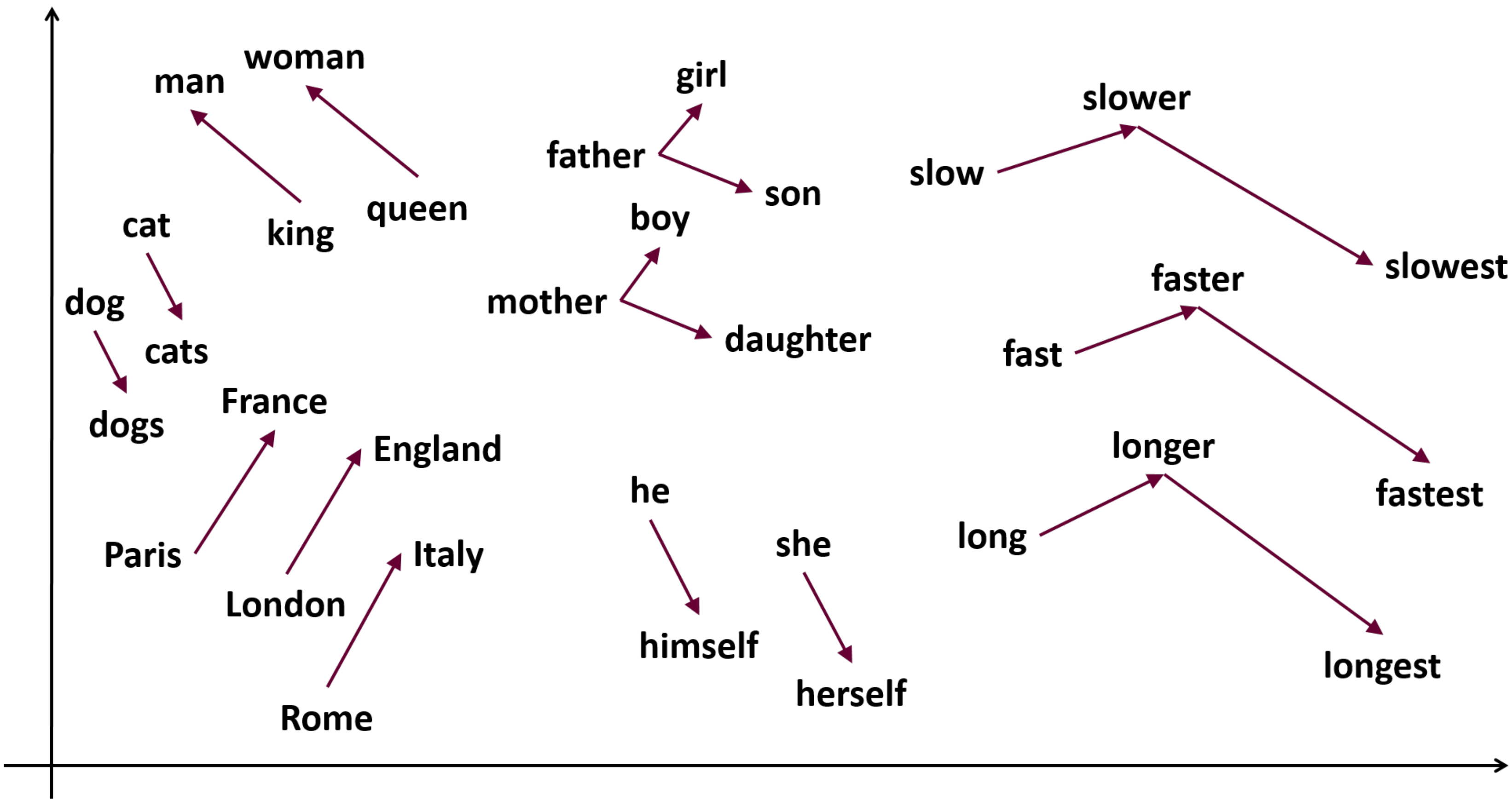
Bag of words (100,000 dimensions)

Car	0	0	…	0	1	0	…	0	0	0
Bike	0	0	…	0	0	0	…	0	1	0

Embedding (100 dimensions)

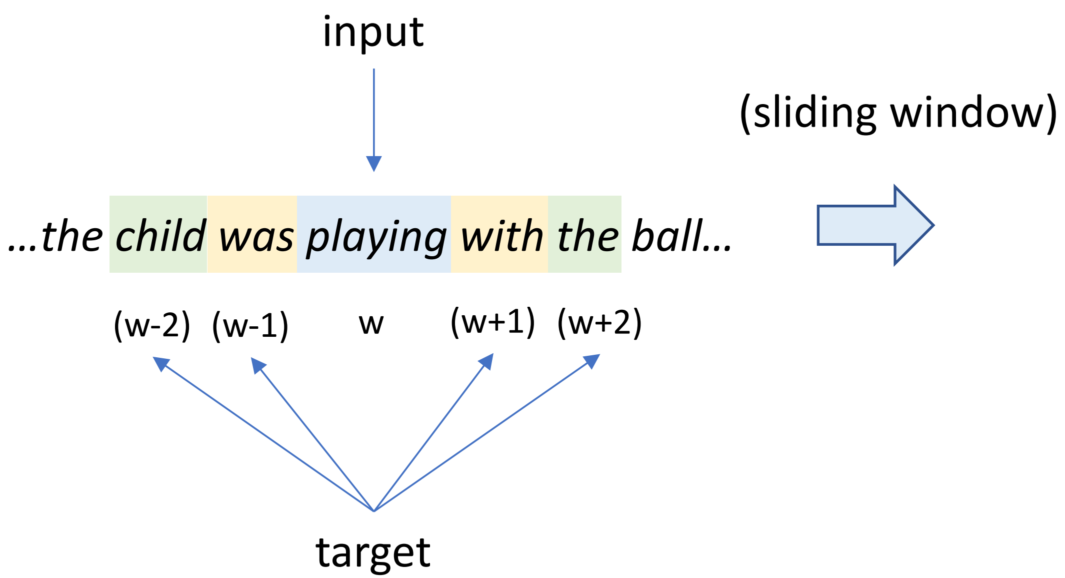
Car	0.1	0.5	-0.3	…	-0.7	0.4	0.2
Bike	0.1	0.6	-0.3	…	-0.7	0.5	0.2

Word Embeddings in 2D

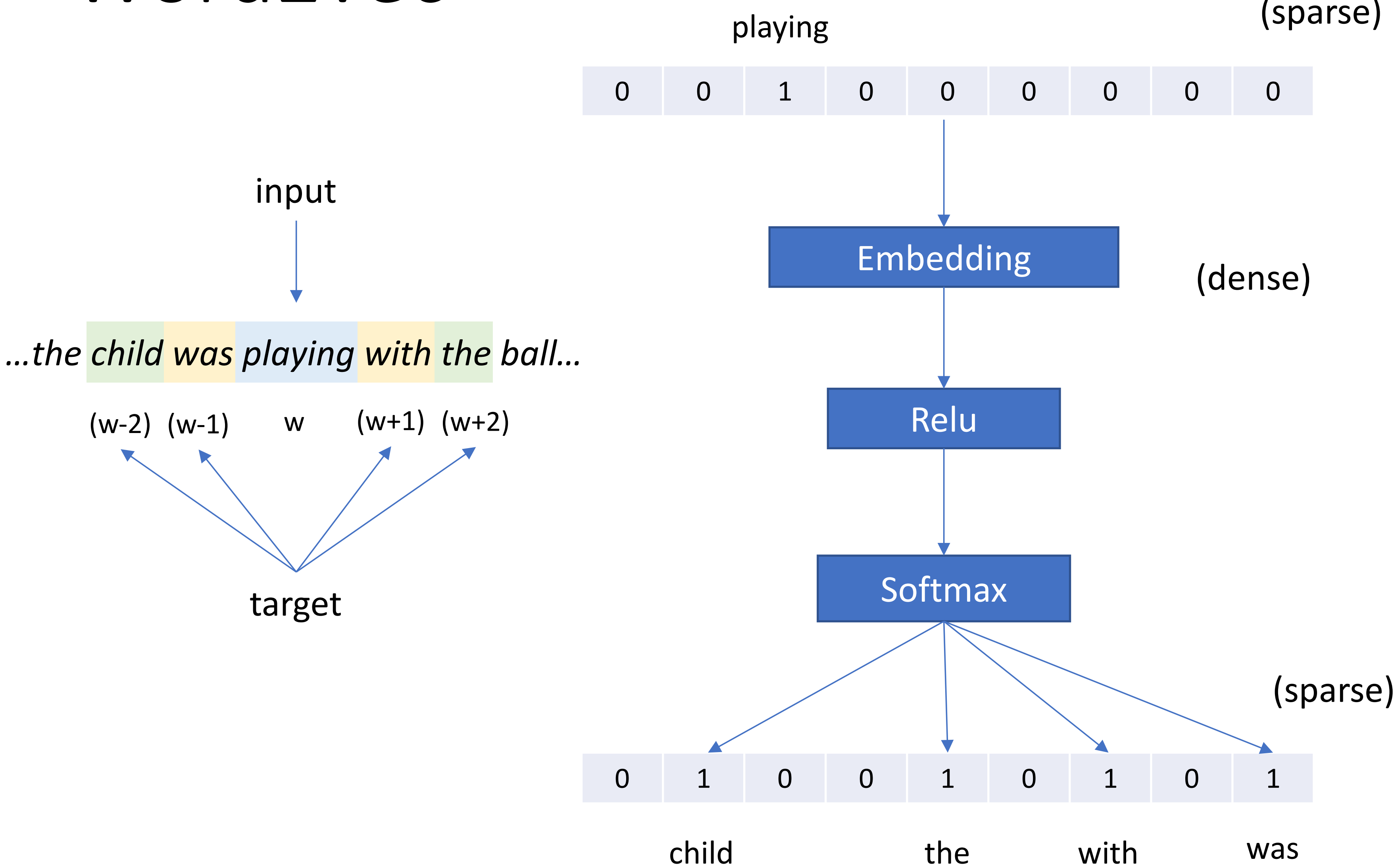


Word2vec

- Assumption: words that appear in the same context are similar to each other

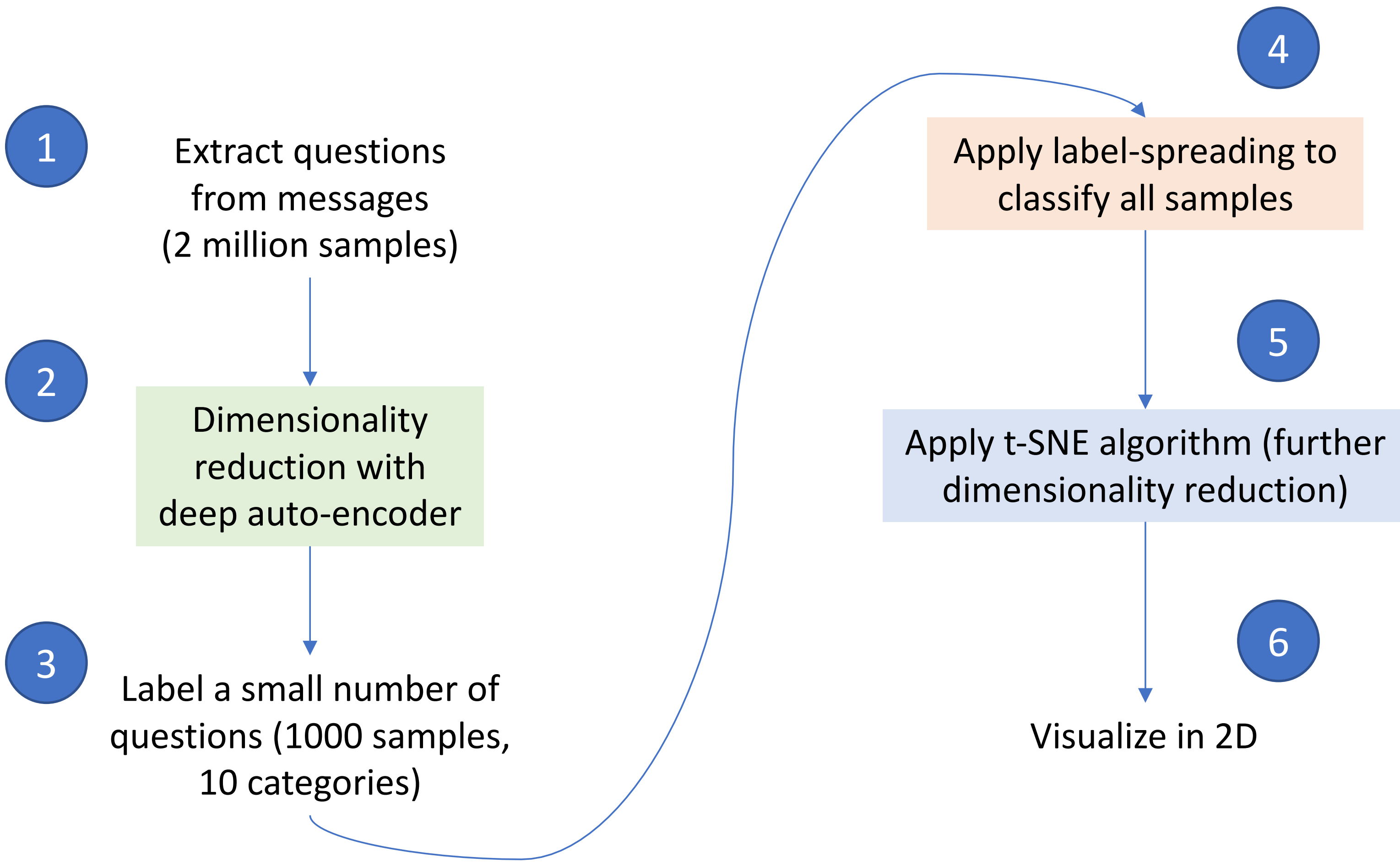


Word2vec



input
...the child was playing with the ball...
(w-2) (w-1) w (w+1) (w+2)
target
playing
(sparse)
0 0 1 0 0 0 0 0 0
Embedding
(dense)
Relu
Softmax
(sparse)
0 1 0 0 1 0 1 0 1
child
the
with
was

Topic modeling with auto-encoders + semi-supervised learning



AlexNet

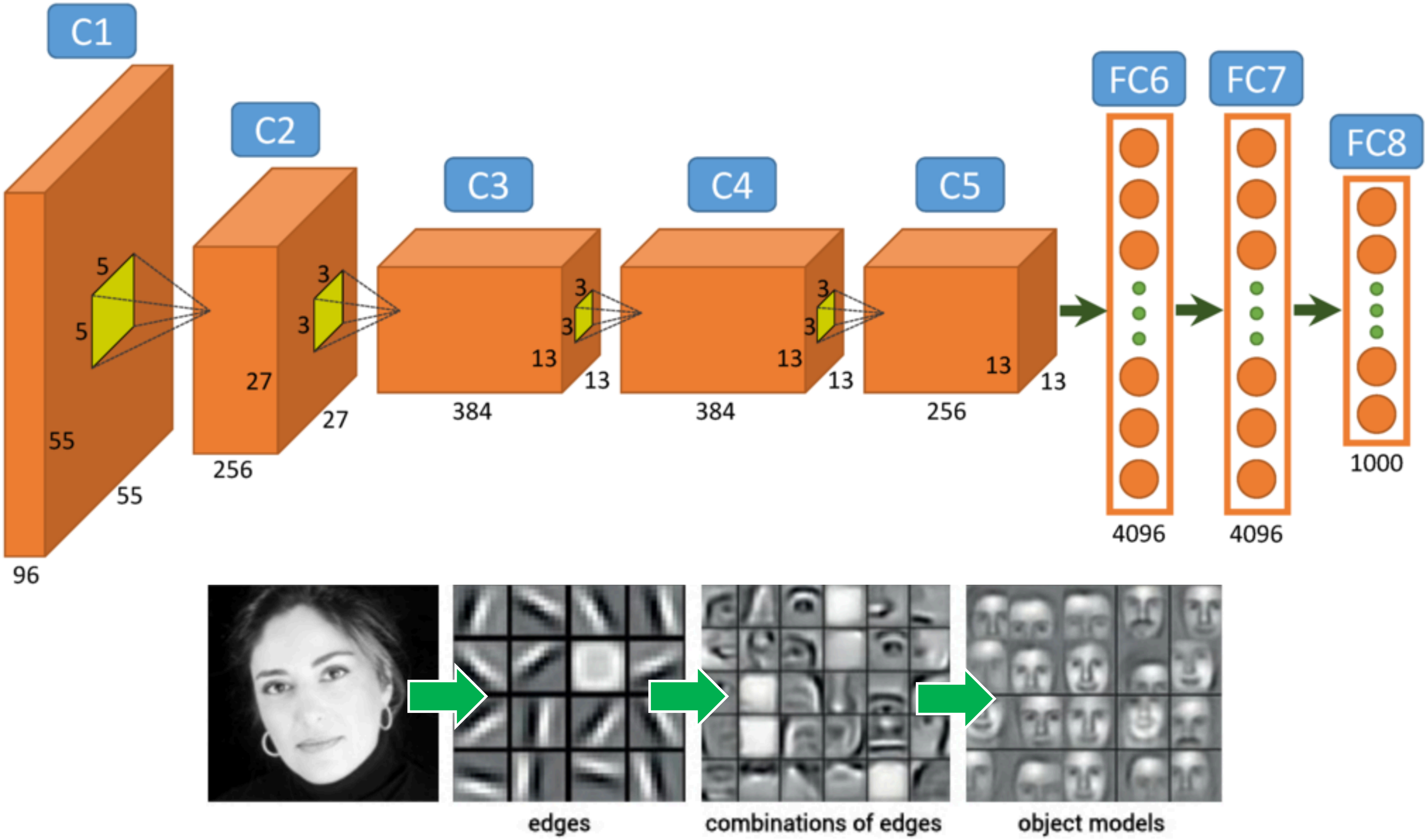
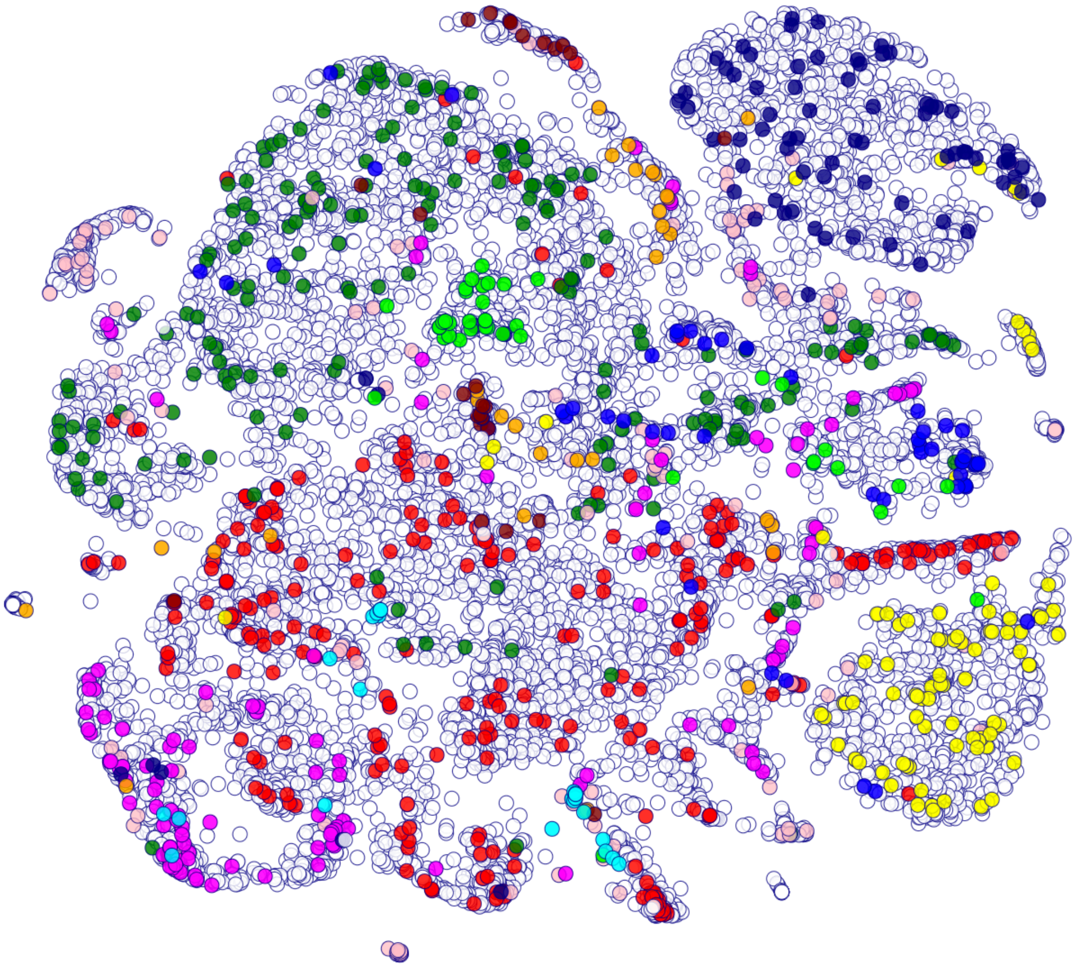
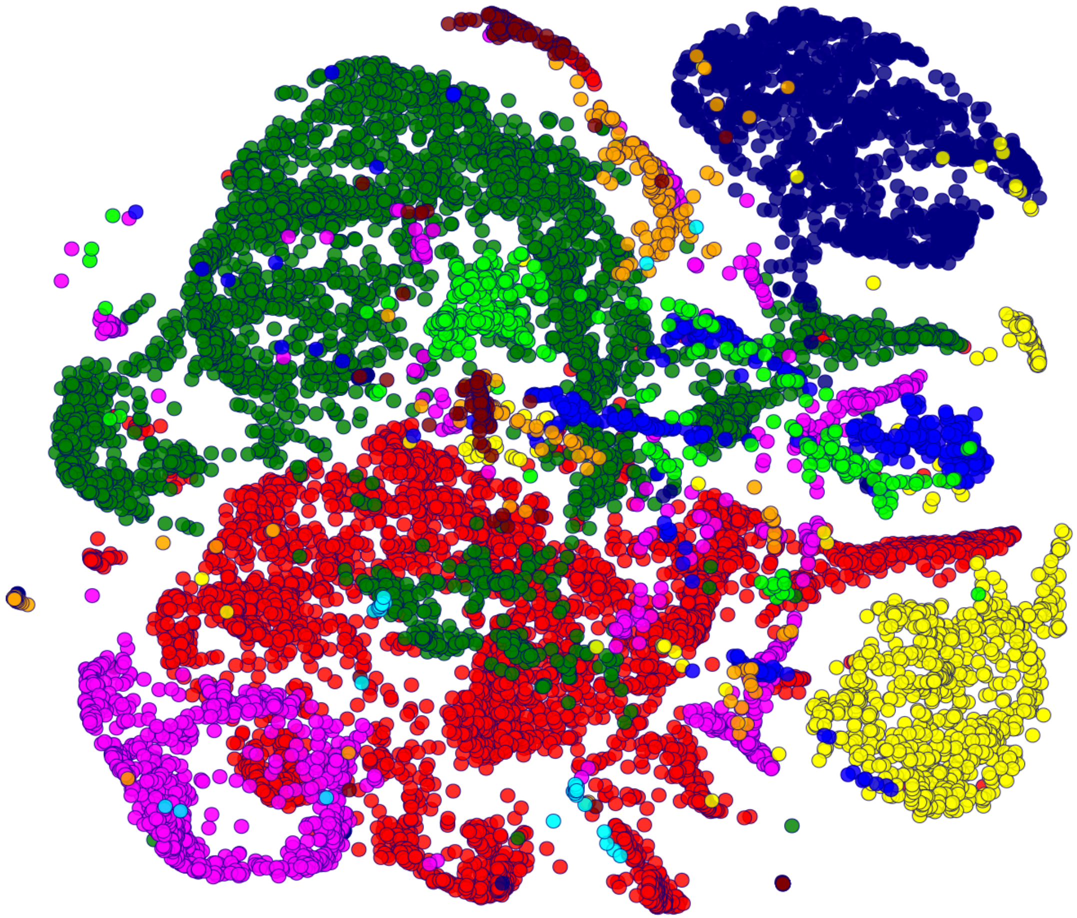


Image source: https://www.saagie.com/blog/object-detection-part1

Auto-encoder + 1000 labeled samples

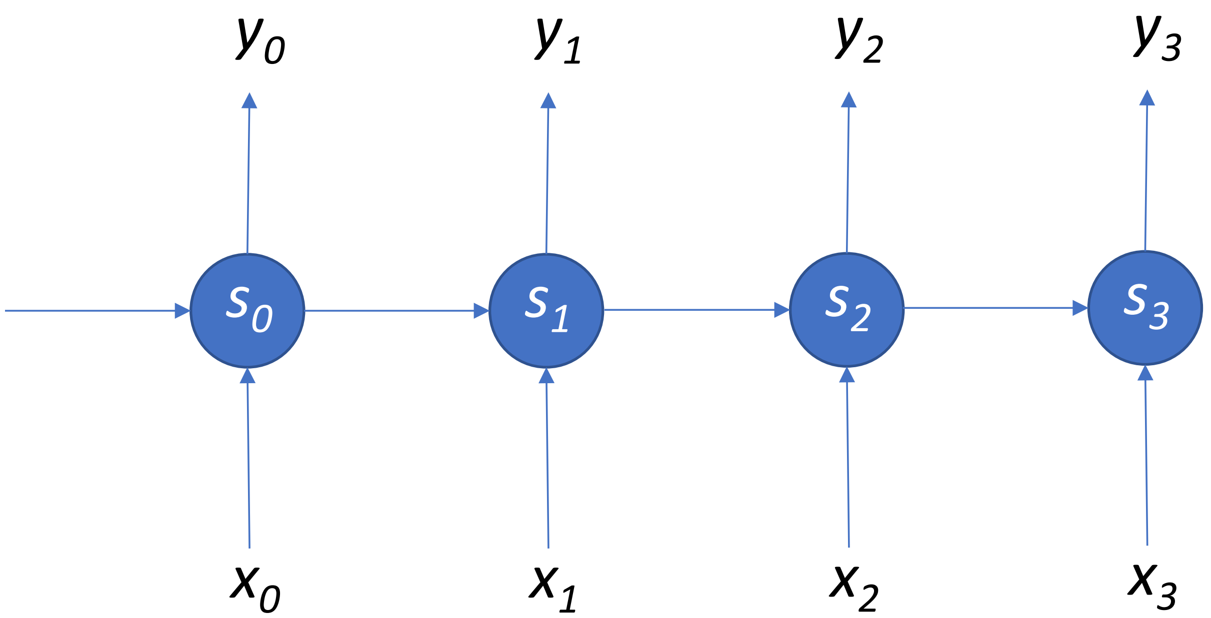


Auto-coder + Label Spreading

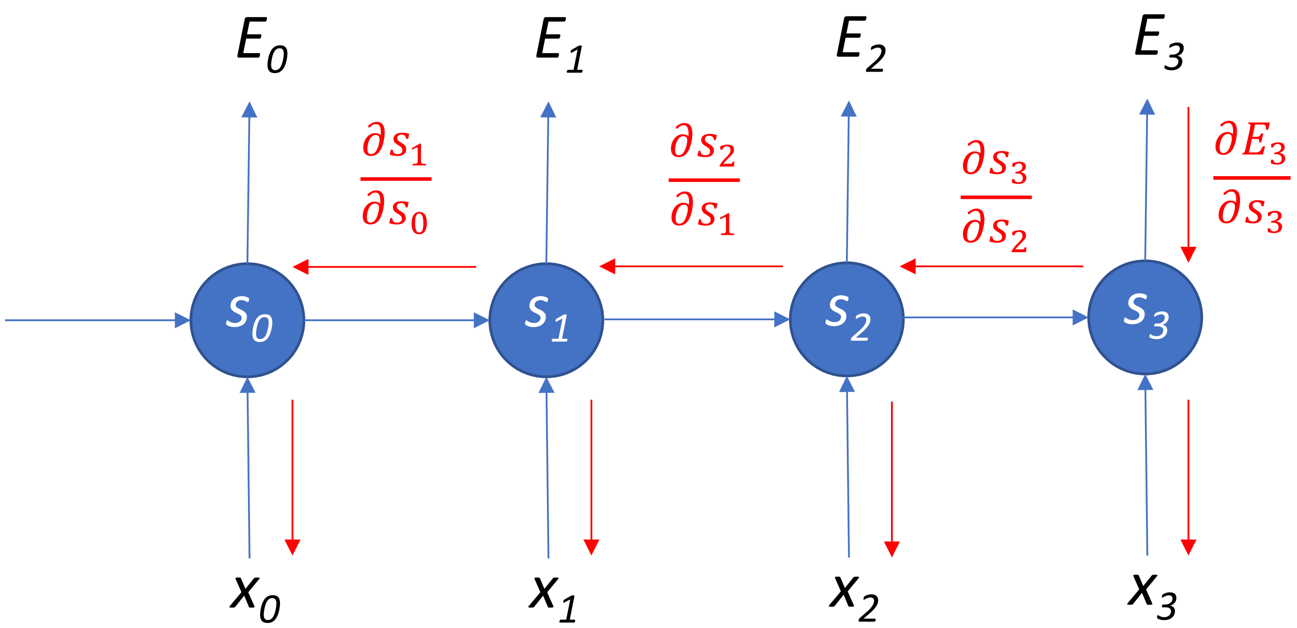


Recurrent Neural Networks

"Unrolled" network

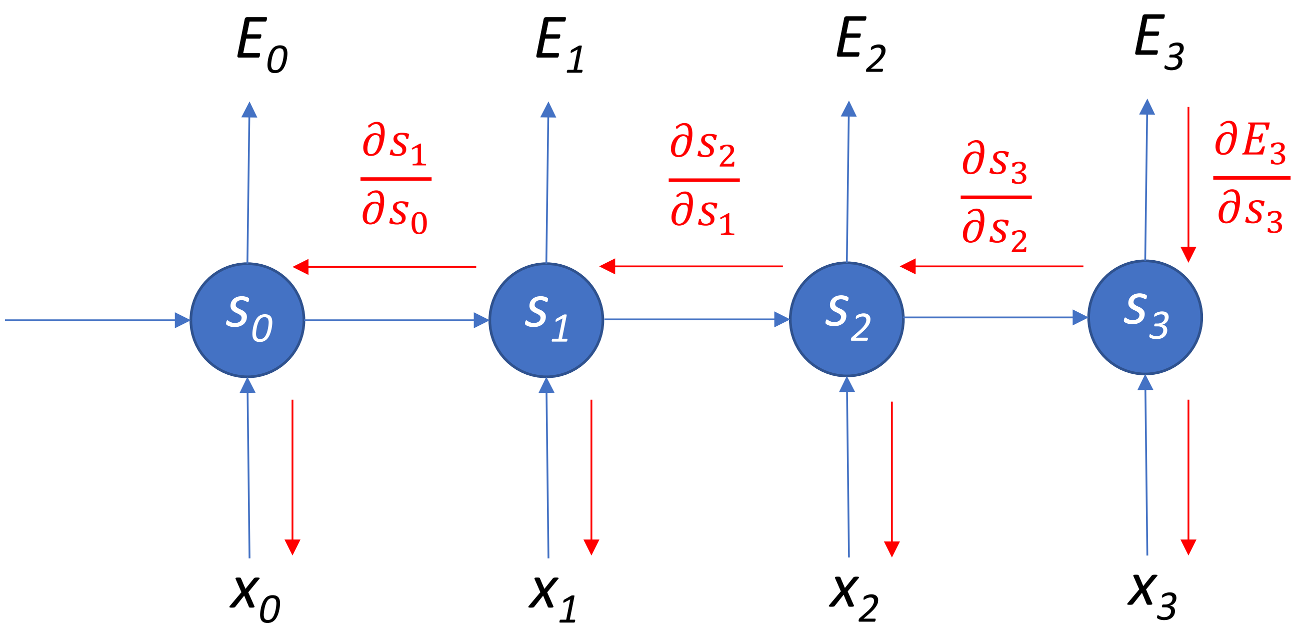


Backpropagation Through Time (BPTT)



Backpropagation Through Time (BPTT)

Vanishing gradient problem!



Structured vs Unstructured data

Structured

Results | Messages

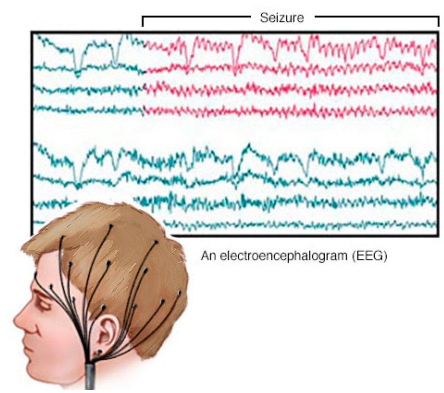
	Id	Address	Date	Id	Name	Date
1	1	Bangalore	2011-06-09 15:23:37.873	1	shirsendu	2011-06-09 15:17:58.130
2	1	Bangalore	2011-06-09 15:23:37.873	1	shirsendu	2011-06-09 15:19:30.087
3	1	Bangalore	2011-06-09 15:23:37.873	2	Sarnali	2011-06-09 15:19:30.087
4	1	Bangalore	2011-06-09 15:23:37.873	3	Mrinal	2011-06-09 15:19:30.087
5	2	Bangkok	2011-06-09 15:23:37.890	1	shirsendu	2011-06-09 15:17:58.130
6	2	Bangkok	2011-06-09 15:23:37.890	1	shirsendu	2011-06-09 15:19:30.087
7	2	Bangkok	2011-06-09 15:23:37.890	2	Sarnali	2011-06-09 15:19:30.087
8	2	Bangkok	2011-06-09 15:23:37.890	3	Mrinal	2011-06-09 15:19:30.087
9	3	CAlcutta	2011-06-09 15:23:37.890	1	shirsendu	2011-06-09 15:17:58.130
10	3	CAlcutta	2011-06-09 15:23:37.890	1	shirsendu	2011-06-09 15:19:30.087
11	3	CAlcutta	2011-06-09 15:23:37.890	2	Sarnali	2011-06-09 15:19:30.087
12	3	CAlcutta	2011-06-09 15:23:37.890	3	Mrinal	2011-06-09 15:19:30.087

PROCLIB.PAYLIST Table

IdNum	Gender	Jobcode	Salary	Birth	Hired
1639	F	TA1	42260	26JUN70	28JAN91
1065	M	ME3	38090	26JAN54	07JAN92
1400	M	ME1	29769	05NOV67	16OCT90
1561	M		36514	30NOV63	07OCT87
1221	F	FA3	.	22SEP63	04OCT94

Unstructured

Dies ist ein Blindtext. An ihm lässt sich vieles über die Schrift ablesen, in der er gesetzt ist. Auf den ersten Blick wird der Grauwert der Schriftfläche sichtbar. Dann kann man prüfen, wie gut die Schrift zu lesen ist und wie sie auf den Leser wirkt. Dies ist ein Blindtext. An ihm lässt sich vieles über die Schrift ablesen, in der er gesetzt ist. Auf den ersten Blick wird der Grauwert der Schriftfläche sichtbar. Dann kann man prüfen, wie gut die Schrift zu lesen ist und wie sie auf den Leser wirkt.



Structured vs Unstructured data

Structured

Results | Messages

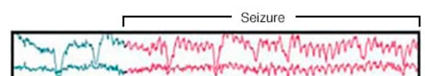
	Id	Address	Date	Id	Name	Date
1	1	Bangalore	2011-06-09 15:23:37.873	1	shirsendu	2011-06-09 15:17:58.130
2	1	Bangalore	2011-06-09 15:23:37.873	1	shirsendu	2011-06-09 15:19:30.087
3	1	Bangalore	2011-06-09 15:23:37.873	2	Samali	2011-06-09 15:19:30.087
4	1	Bangalore	2011-06-09 15:23:37.873	3	Mrinal	2011-06-09 15:19:30.087
5	2	Bangkok	2011-06-09 15:23:37.890	1	shirsendu	2011-06-09 15:17:58.130
6	2	Bangkok	2011-06-09 15:23:37.890	1	shirsendu	2011-06-09 15:19:30.087
7	2	Bangkok	2011-06-09 15:23:37.890	2	Samali	2011-06-09 15:19:30.087
8	2	Bangkok	2011-06-09 15:23:37.890	3	Mrinal	2011-06-09 15:19:30.087
9	3	CAlcutta	2011-06-09 15:23:37.890	1	shirsendu	2011-06-09 15:17:58.130
10	3	CAlcutta	2011-06-09 15:23:37.890	1	shirsendu	2011-06-09 15:19:30.087
11	3	CAlcutta	2011-06-09 15:23:37.890	2	Samali	2011-06-09 15:19:30.087
12	3	CAlcutta	2011-06-09 15:23:37.890	3	Mrinal	2011-06-09 15:19:30.087

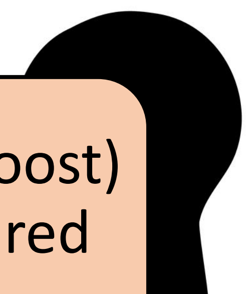
PROCLIB.PAYLIST Table

Id					
16					
10					
14					
15					
1221	F	FA3	.	22SEP63	04OCT94

Unstructured

Dies ist ein Blindtext. An ihm lässt sich vieles über die Schrift ablesen, in der er gesetzt ist. Auf den ersten Blick wird der Grauwert der Schriftfläche sichtbar. Dann kann man prüfen, wie gut die Schrift zu lesen ist und wie sie auf den Leser wirkt. Dies ist ein Blindtext. An ihm lässt sich vieles über die Schrift ablesen, in der er gesetzt ist. Auf den ersten Blick wird der Grauwert der Schriftfläche sichtbar. Dann kann man prüfen, wie gut die Schrift zu lesen ist und wie sie auf den Leser wirkt.





Practitioner's tip: Use tree-based methods (e.g., XGBoost) for structured data and neural networks for unstructured data.