Some (cool) DL applications

Emilio Serrano
2016
Technical University of Madrid

Summary

- Deep Reinforcement Learning
- (Machine vision)
- Speech recognition
  - RNNs
  - LSTM RNNs
- Natural language processing
  - Word2Vec

ML applications (reminder)

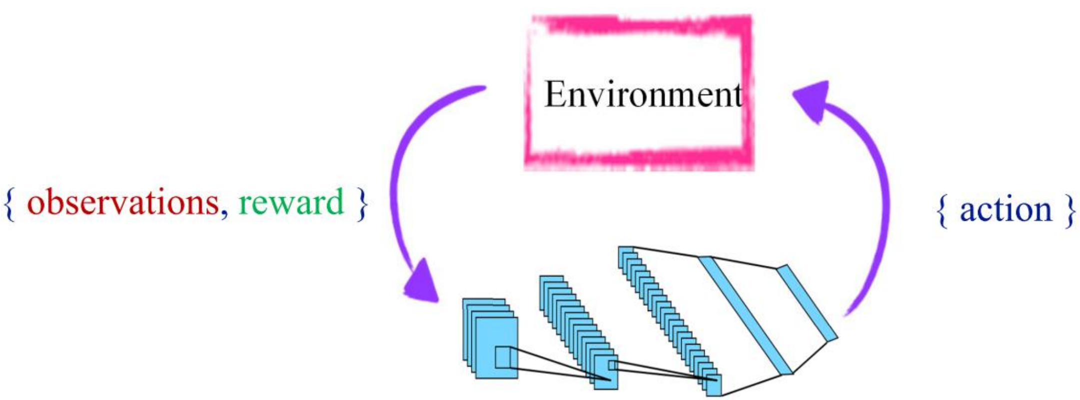
- Bioinformatics
- Classifying DNA sequences
- Computer vision
- Detecting frauds
- Medical diagnosis
- Online advertising
- Recommender systems
- Robot locomotion
- Search engines
- Speech and handwriting recognition

- Processing loan applications
- Screening images for oil slicks
- Electricity supply forecasting
- Diagnosis of machine faults
- Marketing and sales
- Separating crude oil and natural gas
- Reducing banding in rotogravure printing
- Finding appropriate technicians for telephone faults
- Scientific applications: biology, astronomy, chemistry
- Automatic selection of TV programs
- Monitoring intensive care patients

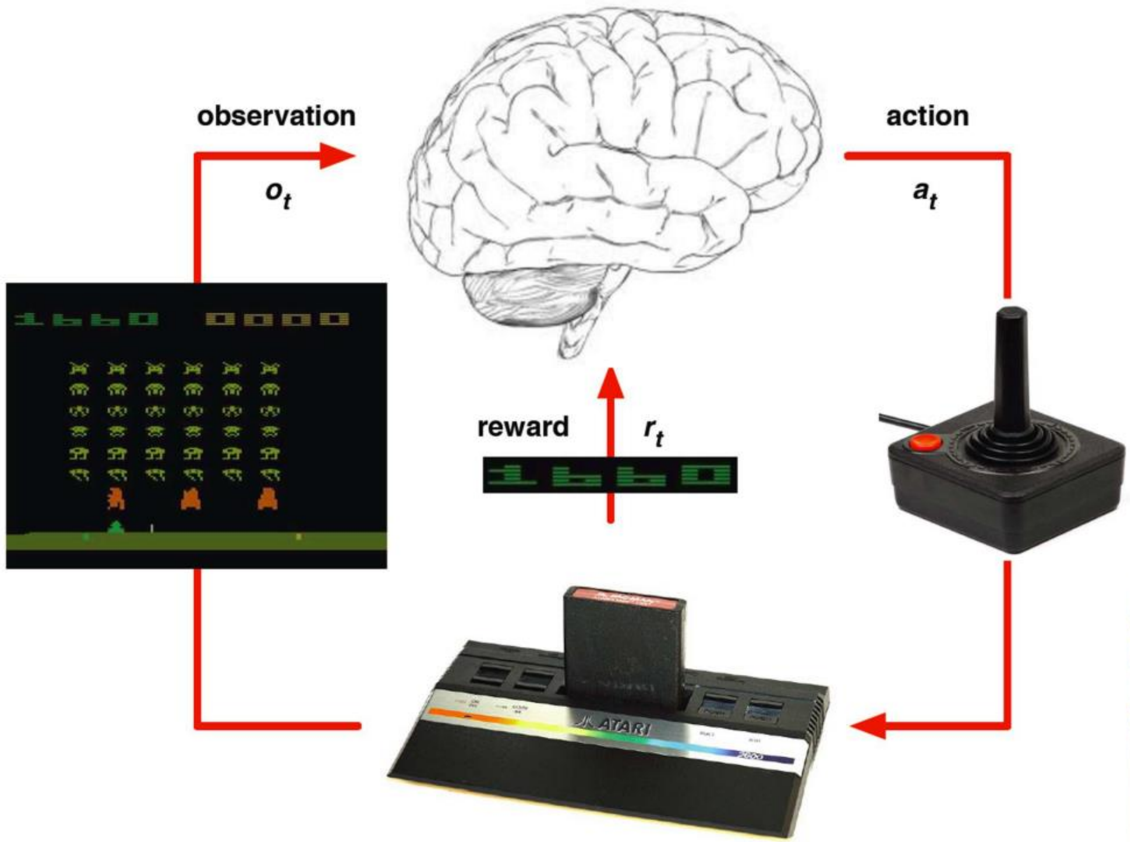
Deep Reinforcement Learning

Deep Reinforcement Learning [3]

- No models, labels, demonstrations, or any other human-provided supervision signal.
- **Representation** has been a challenge/missing in RL.
- DRL combines deep neural networks with RL
- Learn to act from high-dimensional sensory inputs.
- **Is a noisy, sparse, and delayed reward signal sufficient for training deep networks?**



Learning to play Atari [3][4]

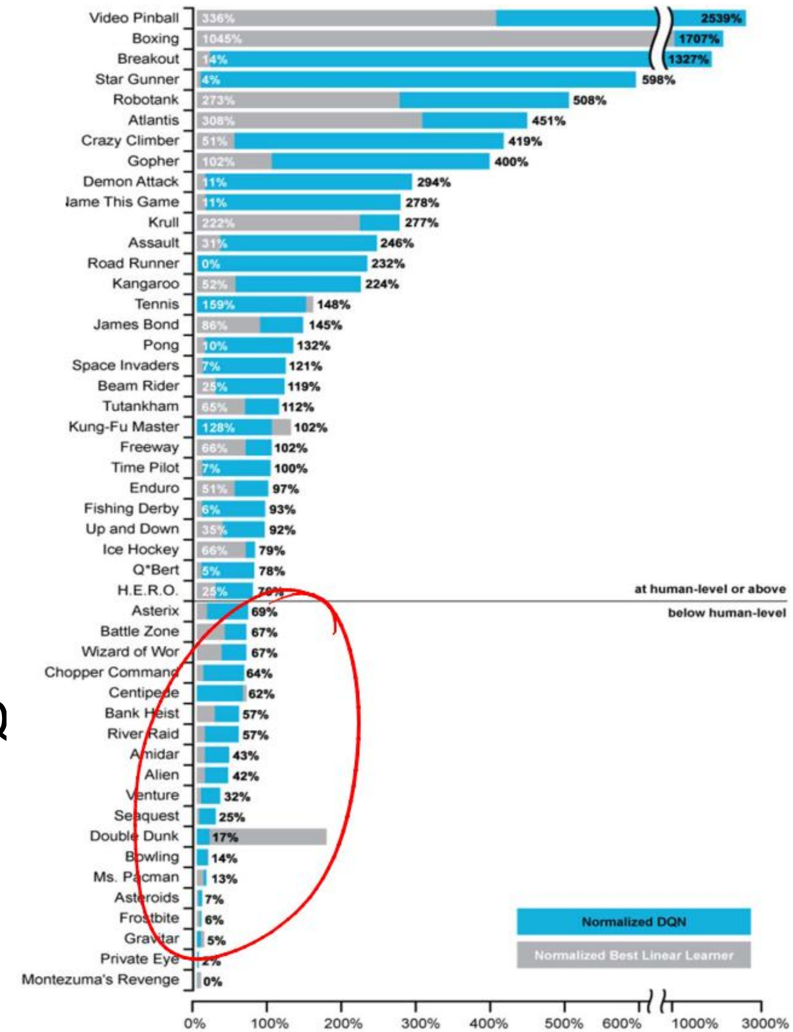


David Silver

[Google DeepMind's Deep Q-learning playing Atari Breakout]()

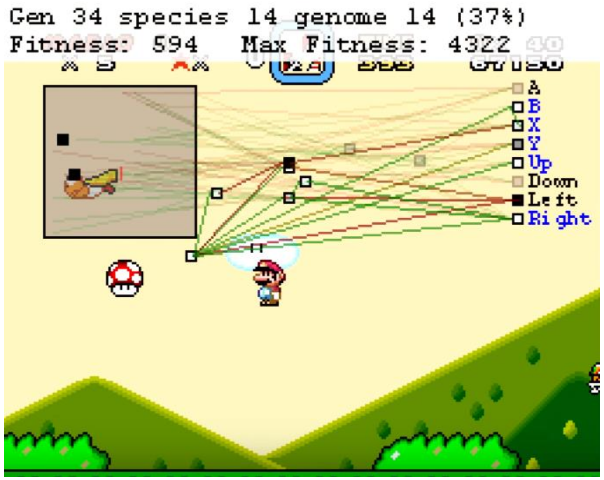
Learning to play Atari 2

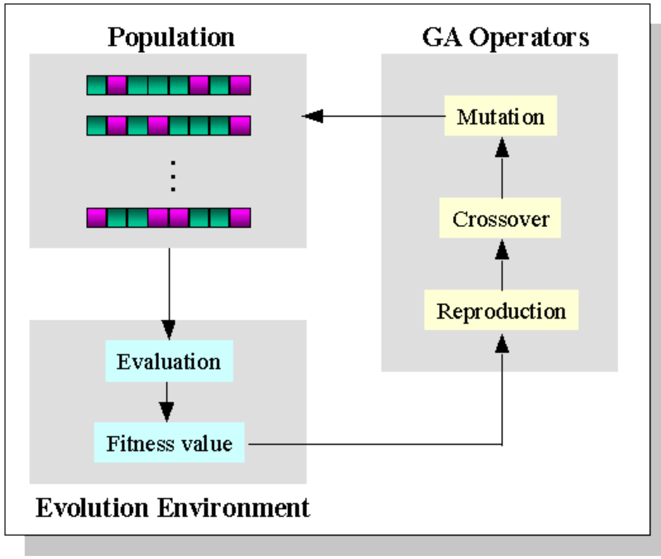
- Achieved performance comparable to or better than an experienced human on 29 out of 49 games
- The architecture and hyperparameter values were the same for all 49 games
  - No expert system or even data!
- Uses a Deep Q-network (DQN)
  - type of convolutional network
  - represent the action value function Q
  - Q(**s**, **a**) is an estimation of how valuable an action **a** at state **s**
- DQN learn Q with end-to-end RL mapping the raw pixels to action values in Atari games



MarI/O

- ❑MarI/O learned an entire level of Super Mario World in 34 tries
  - ❑Nothing assumed
  - ❑Input of the ANN: black moving position and white static positions
  - ❑Output: a console button
- ❑Based on NeuroEvolution of Augmenting Topologies (NEAT)
  - ❑a genetic algorithm for the generation of evolving ANN developed by Ken Stanley in 2002





[MarI/O - Machine Learning for Video Games](#)

AlphaGo [5]

- Go game
  - as chess, search exhaustively for the best move is unfeasible
  - unlike chess, recognizing winning/losing positions is much harder
  - it has frustrated the efforts of AI for decades
- **In March 2016** AlphaGo won 4-1 Lee Sedol: top Go player in the world over the past decade
  - the first computer program to ever beat a professional player

The computer that mastered Go

AlphaGo 2

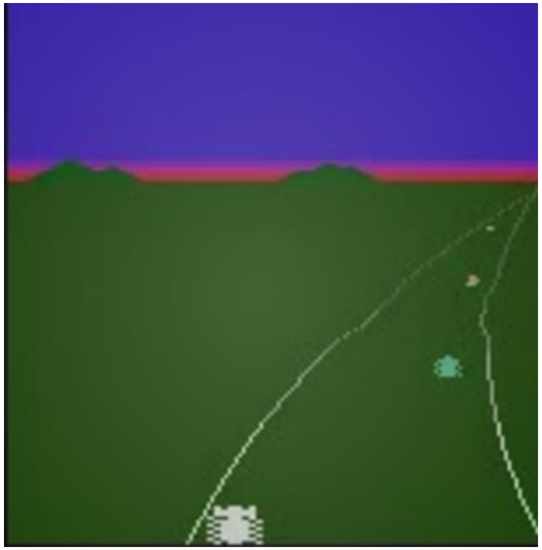
- AlphaGo
  - studied 30 million positions from expert games
    - Monte-Carlo tree search with DNN trained by supervised learning.
  - played against itself across 50 computers
    - DRL
  - Uses two DNNs, both containing many layers with millions of neuron-like connections
    - "policy network", predicts the next move, and is used to narrow the search to consider only the moves most likely to lead to a win
      - (suggests intelligent moves to play)
    - "value network", reduces the depth of the search tree estimating the winner in each position in place of searching all the way to the end of the game
      - (evaluates the position that is reached)

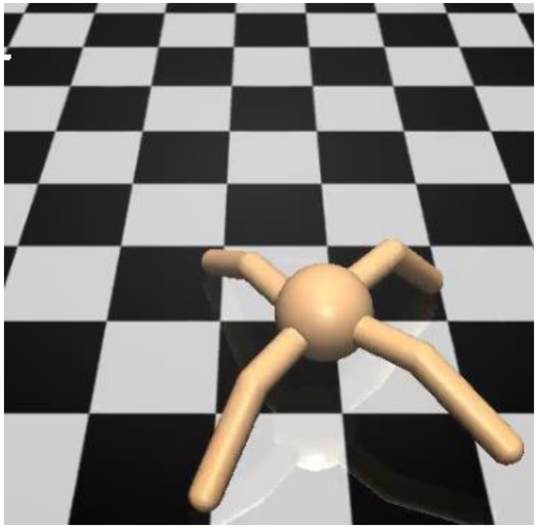
- Why is interesting a DL research on games?
  - Deep Blue beat Kasparov in 1997 being explicitly programmed to win at the game
  - AlphaGo wasn't not preprogrammed to play Go
  - Similar techniques could be applied to other AI domains that require recognition of complex patterns, long-term planning and decision-making
  - Examples are using medical images to make diagnoses or treatment plans, and improving climate-change models

The computer that mastered Go

openai.com

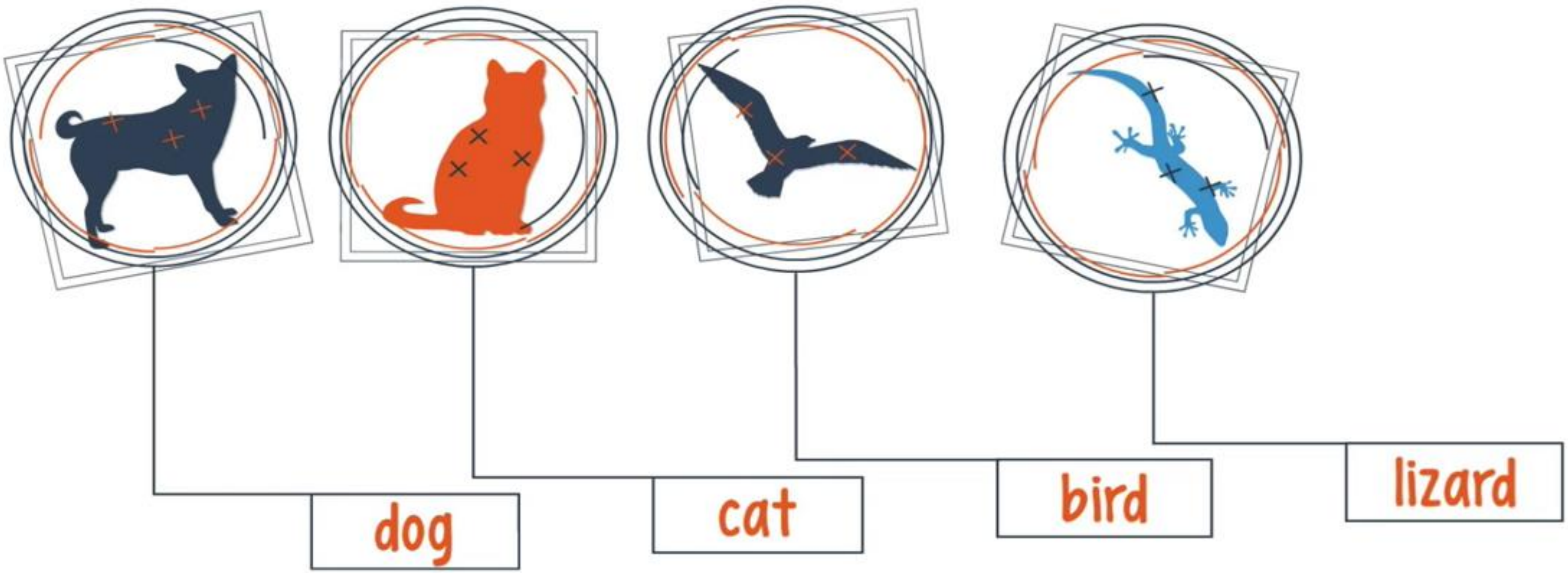
- a toolkit for developing and comparing RL algorithms
- progress has been driven by large labeled datasets like ImageNet in supervised learning
  - suite of environments



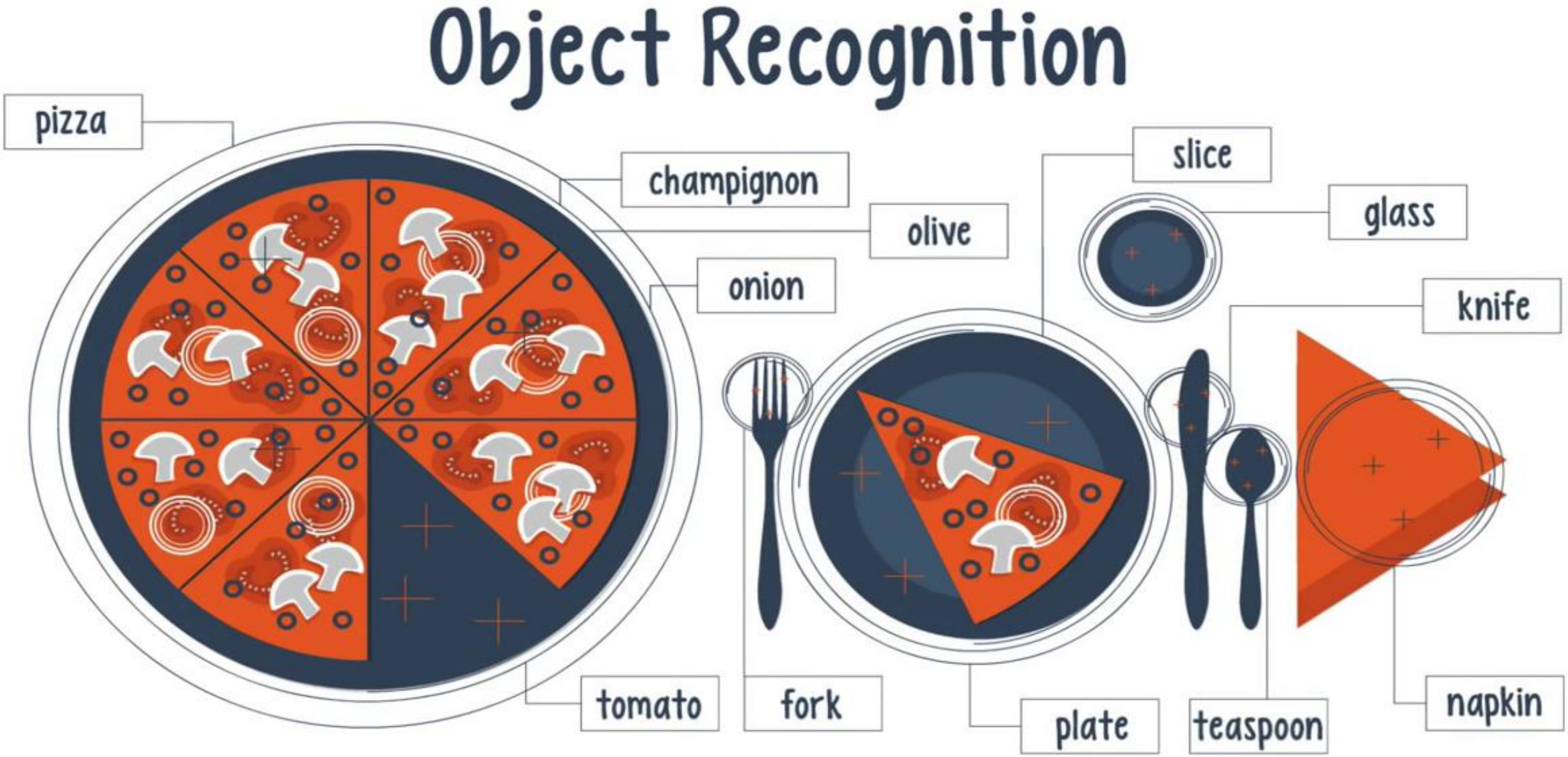


Machine vision
(teaser trailer)

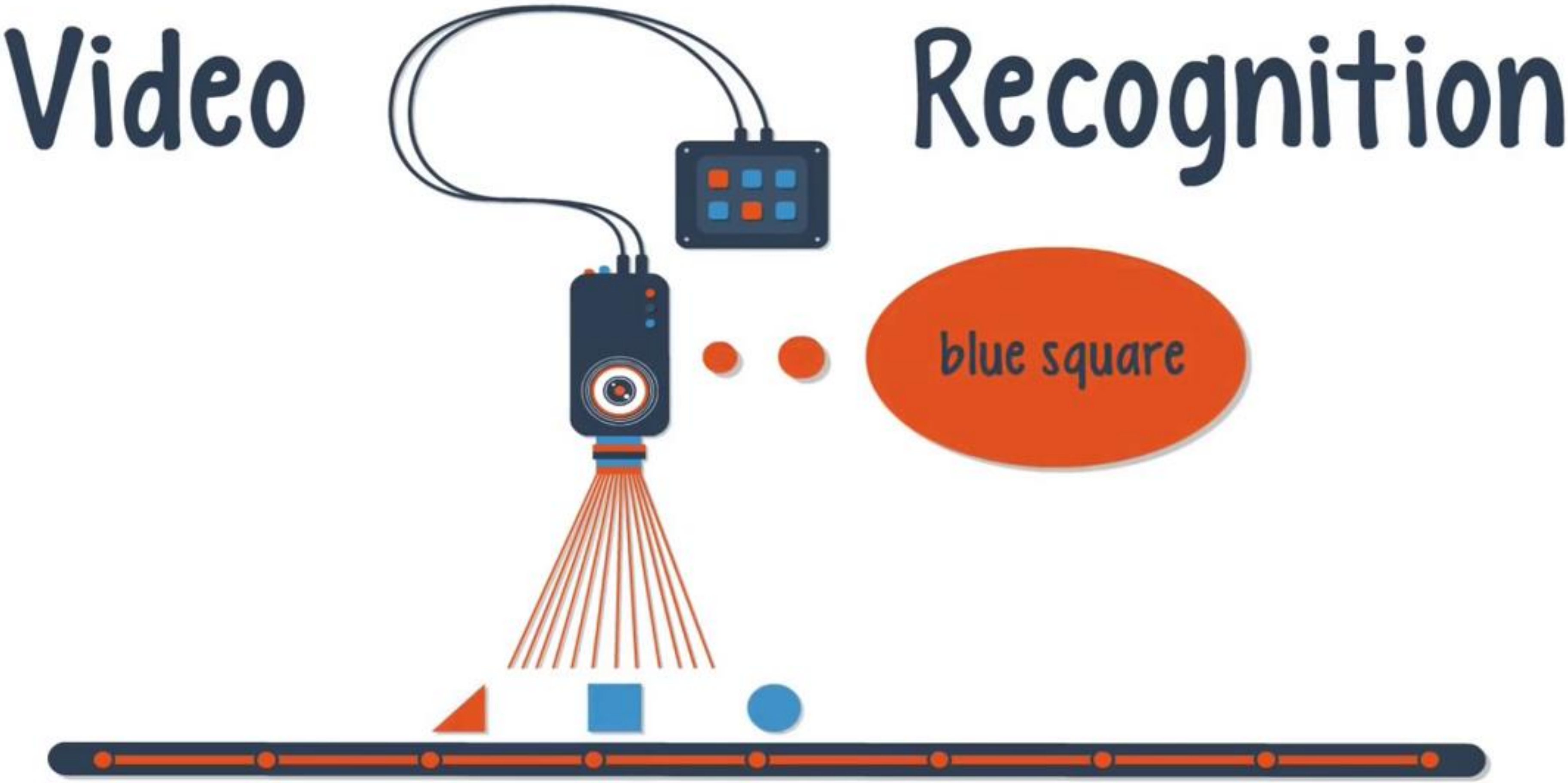
Image classification



From DeepLearning.tv



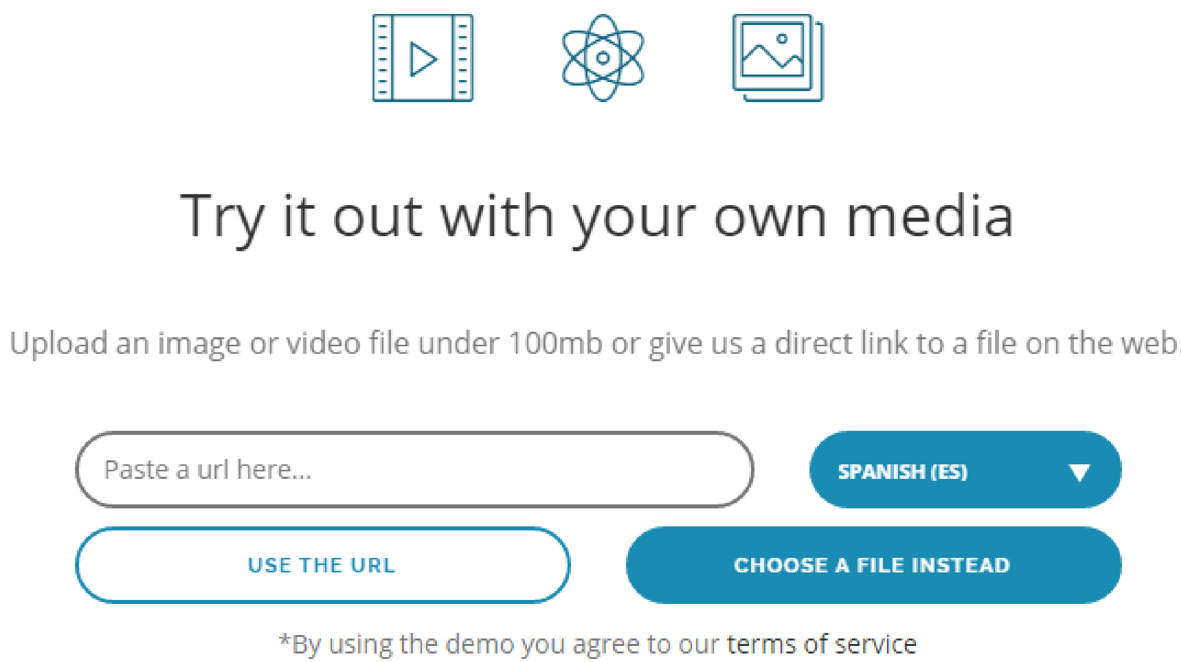
From DeepLearning.tv

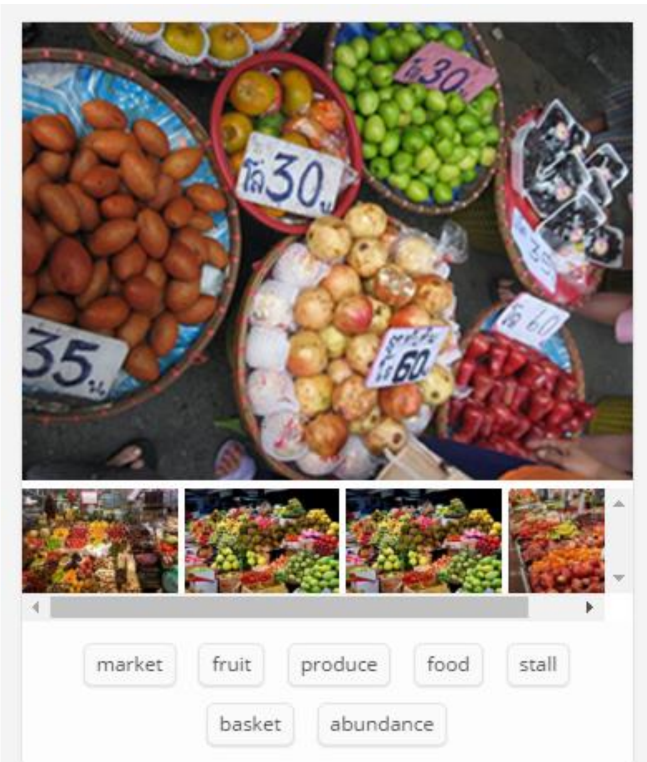


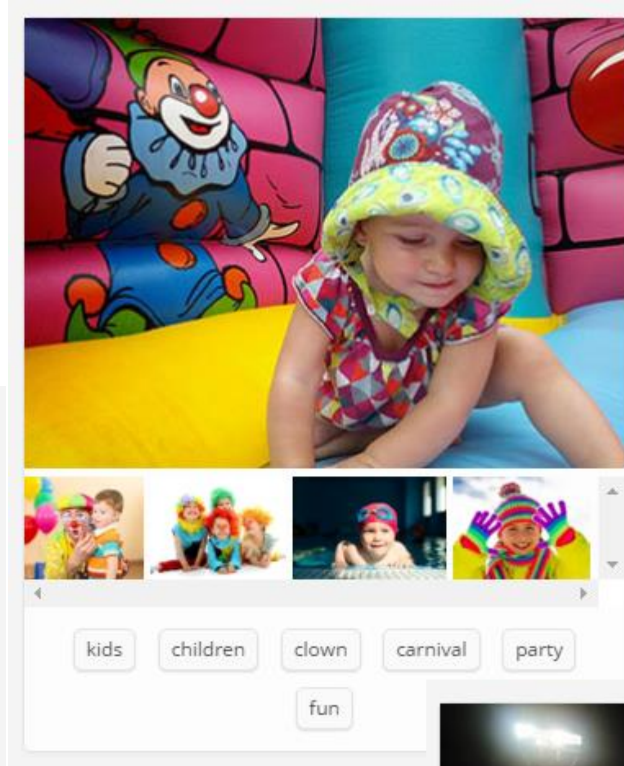
From DeepLearning.tv

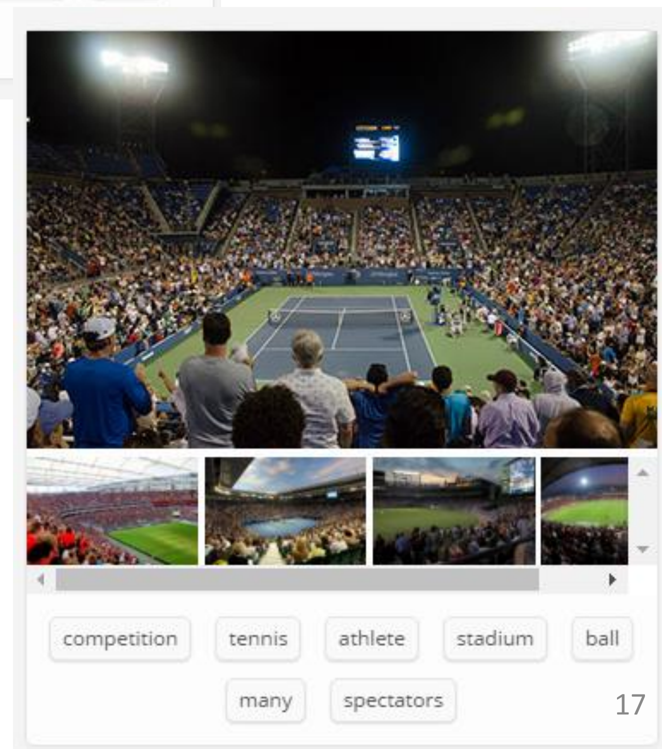
www.clarifie.com

- Automatically tag, search, and find similar images



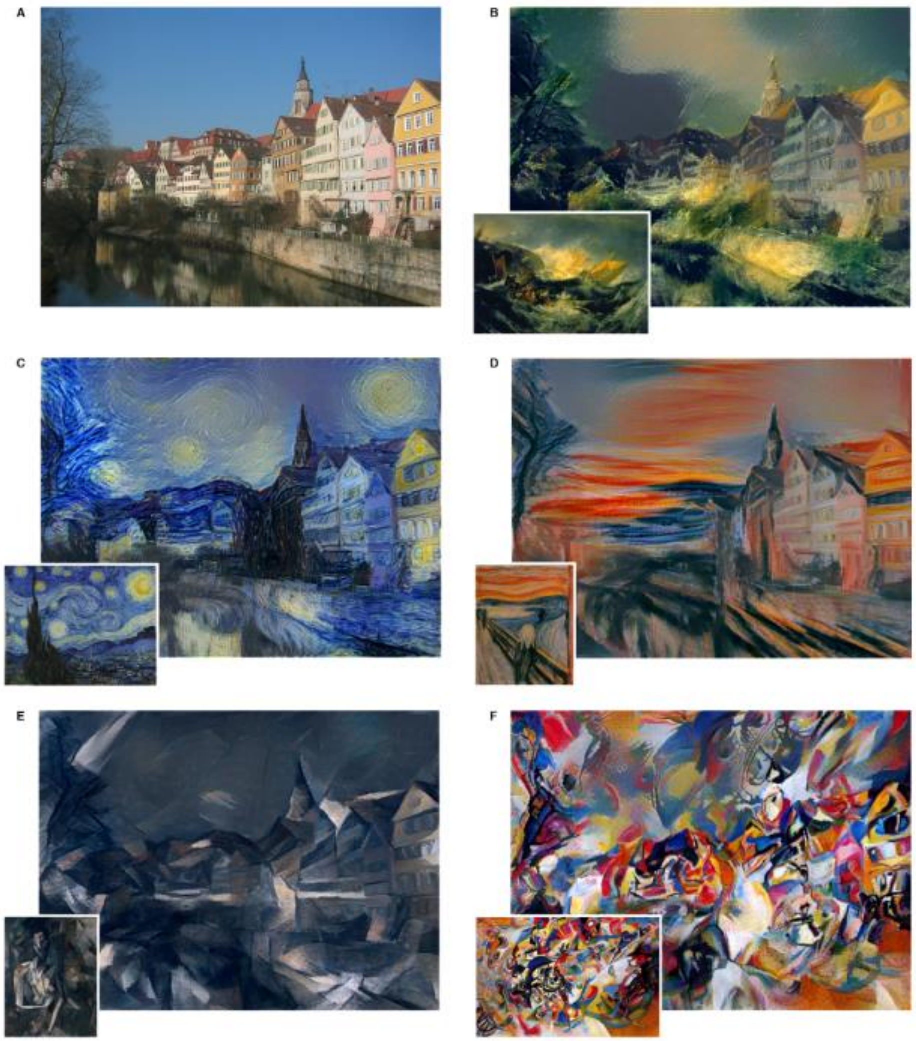

market
fruit
produce
food
stall
basket
abundance


kids
children
clown
carnival
party
fun


competition
tennis
athlete
stadium
ball
many
spectators

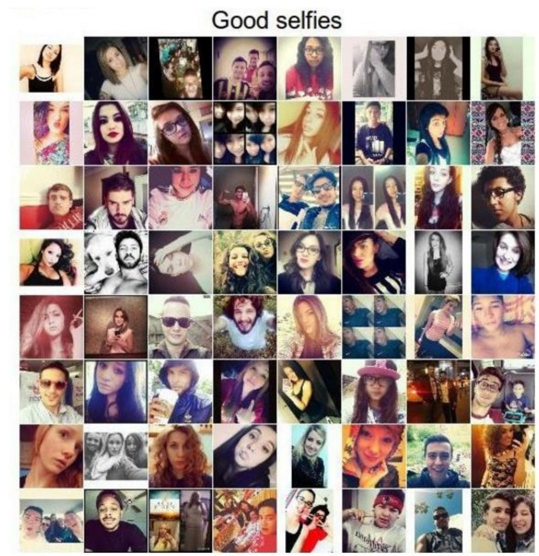
A Neural Algorithm of Artistic Style [5]

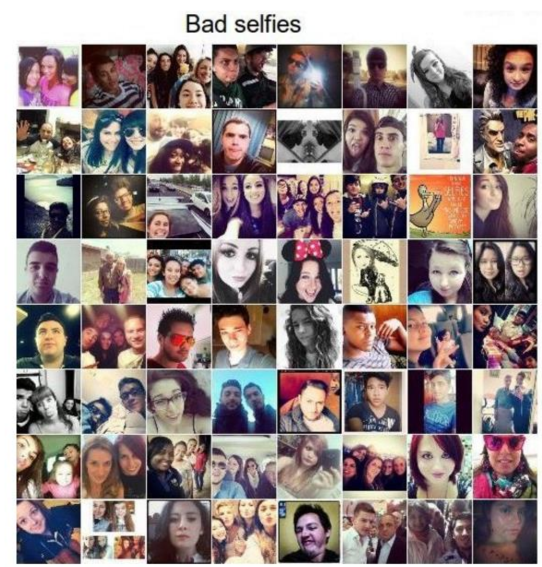
- ConvNets to separate the style and content of different images
- Recombination into one piece
- A path forward to an algorithmic understanding of how humans create and perceive artistic imagery
- (a number of github implementations in Torch, Caffe…)



What makes a good #selfie [6]

- Based on 1 million good and 1 million bad selfies
- Interesting (general) data gathering
  - script to gather **#selfie** number of likes as a function of the audience size, selfies online more than a month to ensure stable like count, filtered people with too few followers or too many follower…
- ConvNets again (with Caffe)
  - you feed them images of whatever you like (along with some labels)
- The DNN allows to rate a selfie (online)
  - Or finding the optimal crop for a selfie picture
  - Or (manually) find patterns in the best selfies
    - Be female!
    - Face = 1/3 of image
    - Don't show entire forehead (???)
    - Show long hair
    - ….



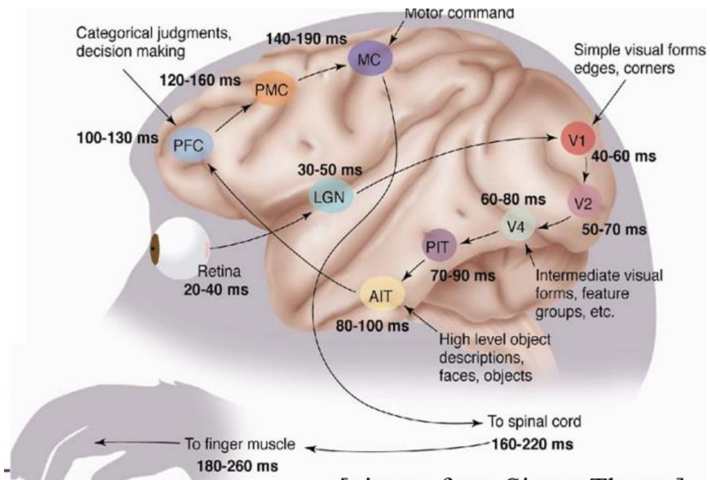


…and plenty more applications

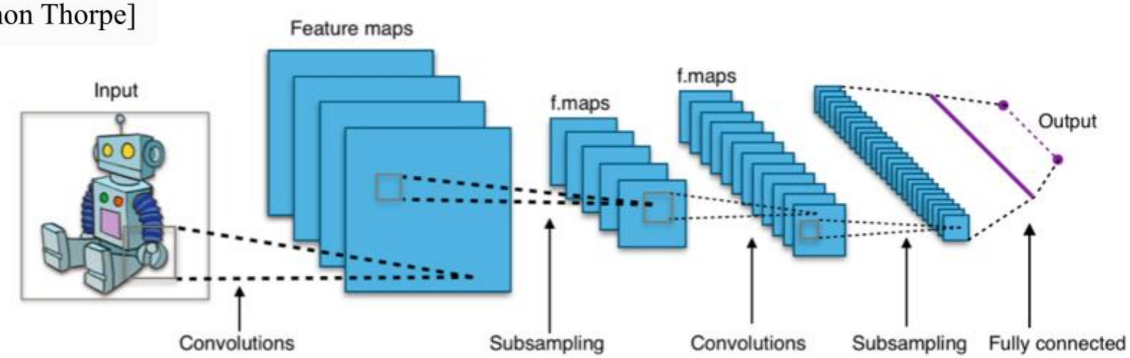
- Reconstructing complete 3D geometry from a single Kinect scan
  - ConvNets
- Improving YouTube video thumbnails with deep neural nets
  - influential convolutional network
- Automatic Colorization
  - ConvNets
- Recognizing and Localizing Endangered Right Whales from areal photos
  - Deep Residual Network (ResNet)
    - LSTM without gates
- Celebrity super-resolution
  - Theano implementation of an convolutional autoencoder

Convolutional neural networks

- inspired by the Visual Cortex
  - The ventral (recognition) pathway in the visual cortex has multiple stages " Retina - LGN - V1 - V2 - V4 - PIT - AIT ....



[picture from Simon Thorpe]



Speech recognition

Speech Recognition

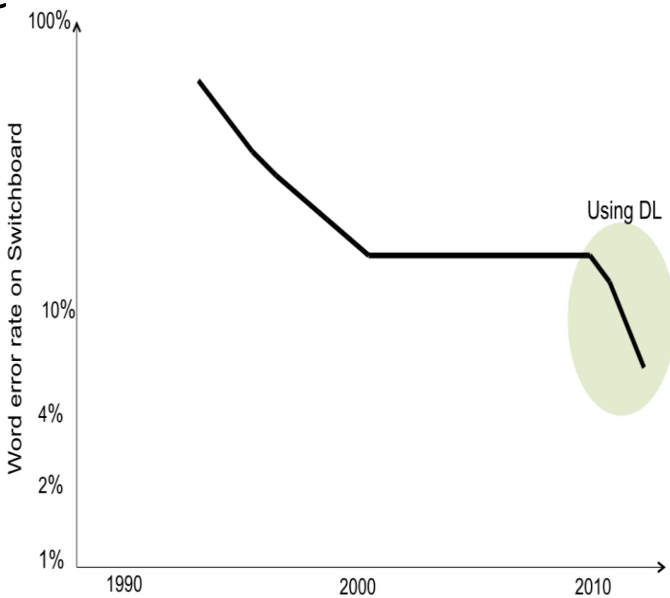
From DeepLearning.tv

State of the art

- Speech recognition has been revolutionized by DL
  - especially by Long short term memory (LSTM) trained by Connectionist Temporal Classification (CTC)
- Large-scale automatic speech recognition is the first and most convincing successful case of deep learning in the recent history
  - both in industry and academia
- All major commercial speech recognition systems are based on deep learning methods [11]
  - Microsoft Cortana, Xbox, Skype Translator, Google Now, Apple Siri, Baidu and iFlyTek voice search, Nuance speech products, etc)
- Neural networks are rapidly replacing previous technologies (as in many other complex services)

The problem

- An acoustic signal is transcribed into words or sub-word units
- More than recognizing individual sounds in the audio [7]
  - sequences of sounds need to match existing words
  - and sequences of words should make sense in the language
- Solved with language models
  - trained over very large corpora of text
  - difficult to find sources that match naturally spoken sentences
    - (Shakespeare's plays in 17th-century English won't help on voicemail)
  - difficult to deal with punctuation
    - hand-crafted rules o "grammars" can't easily take textual context into account
    - "Let's eat, grandpa" vs "Let's eat grandpa"



The dramatic impact of Deep Learning on Speech Recognition (according to Microsoft)

What is behind Google Voice? [7]

- Google Voice transcription improved using Long Short-term Memory Recurrent Neural Networks (LSTM RNNs)
- It used Gaussian Mixture Model (GMM) at the beginning (2009)
  - modeling each phonetic unit
  - each variable assumed to be distributed according to a mixture of K Gaussians Distributions
  - (also used for anomaly detection)
- The service was improved with LSTM RNNs in 2012
  - they differentiate phonetic units with "discriminative training" instead of modeling each one independently
  - LSTM RNNs use recurrent connections and memory cells
  - allow them to "remember" the data they've seen so far
  - (as we interpret the words you hear based on previous words in a sentence)
- Complex data gathering and preprocessing
  - old transcriptions could not be used (tainted with recognition errors)
  - iterative pipeline to retrain the model
    - Improve models
    - Recognize existing voicemails to get better transcriptions
    - Retrain model and repeat
  - Error rate dropped around 50%!!

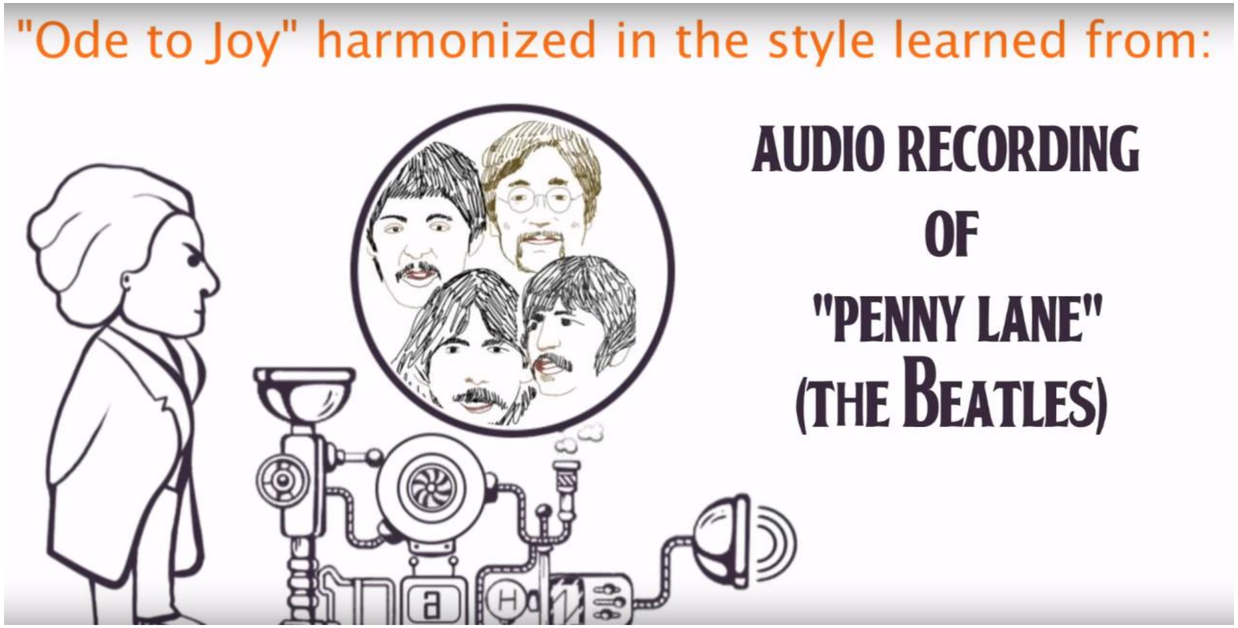
Some solutions [12]

- Results for the TIMIT public corpus
  - 630 speakers from eight major dialects of American English, where each speaker reads 10 sentences

Method	PER (%)
Randomly Initialized RNN	26.1
Bayesian Triphone GMM-HMM	25.6
Hidden Trajectory (Generative) Model	24.8
Monophone Randomly Initialized DNN	23.4
Monophone DBN-DNN	22.4
Triphone GMM-HMM with BMMI Training	21.7
Monophone DBN-DNN on fbank	20.7
Convolutional DNN	20.0
Convolutional DNN w. Heterogeneous Pooling	18.7
Ensemble DNN/CNN/RNN	18.2
Bidirectional LSTM	17.9

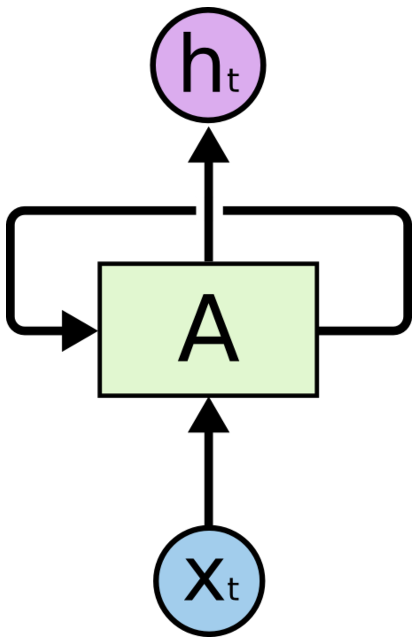
Automatic music generation

- A Joyful Ode to Automatic Orchestration with Markov constraint techniques
  - As in [5], an Artistic Style is modeled from an author work, and the style is combined with another work…
  - …but the time dimension matters now
- Music generation with LSTM RNNs



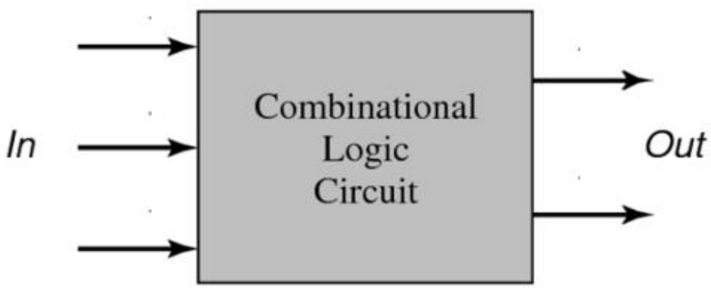
RNNs and the temporal component [8]

- How to classify what is word is a sound at every point in a speech considering previous words?
- It's unclear how a traditional NNs could use its reasoning about previous words to inform later ones
- Recurrent neural networks address this issue
    - their loops allow information to persist
    - RNNs are intimately related to sequences and lists



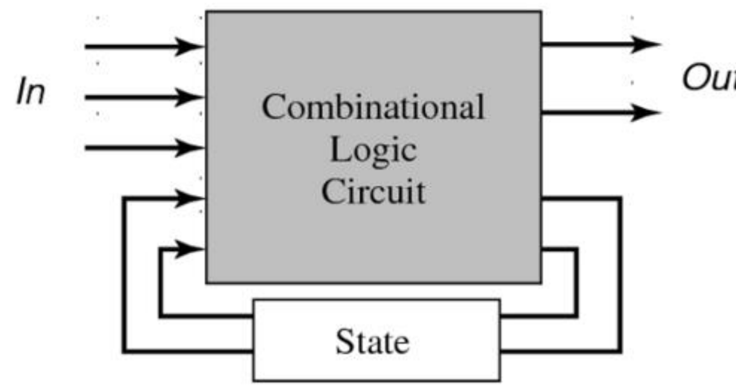
RNNs 2

- Feedforward NN vs RNN = combinational logic vs sequential logic



Combinational

Output = *f*(*In*)

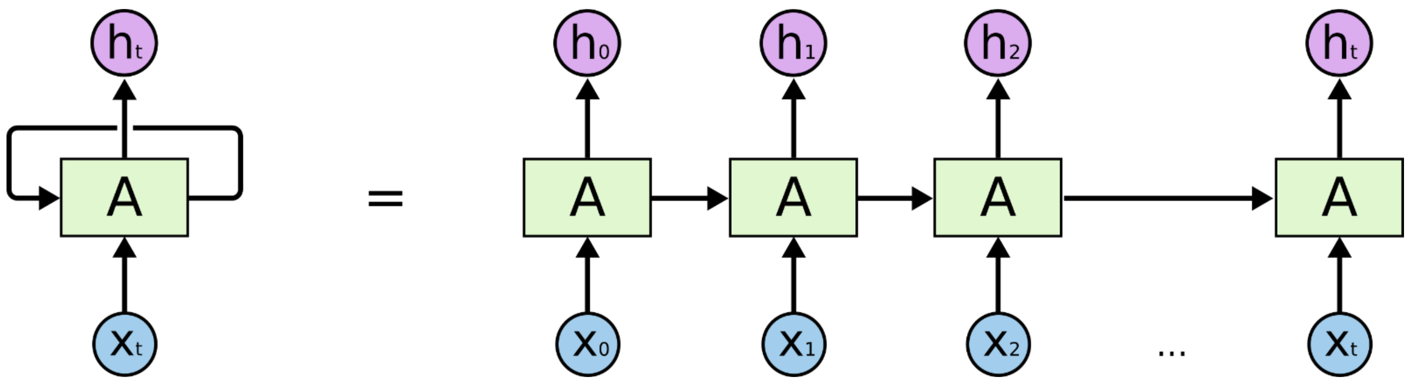


Sequential

Output = *f*(*In*, *Previous In*)

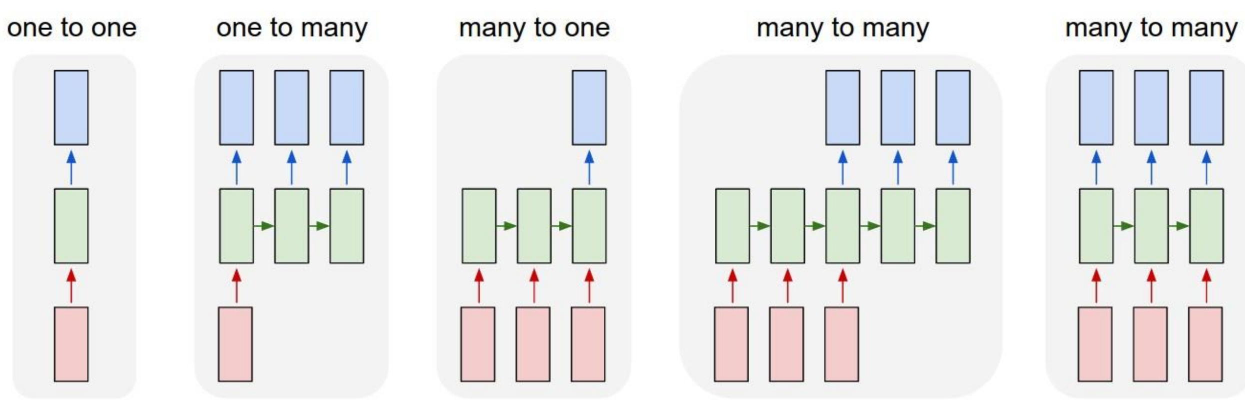
RNNs 3

- If the RNN loop is unrolled, a regular NN is got



RNNs 4 [9]

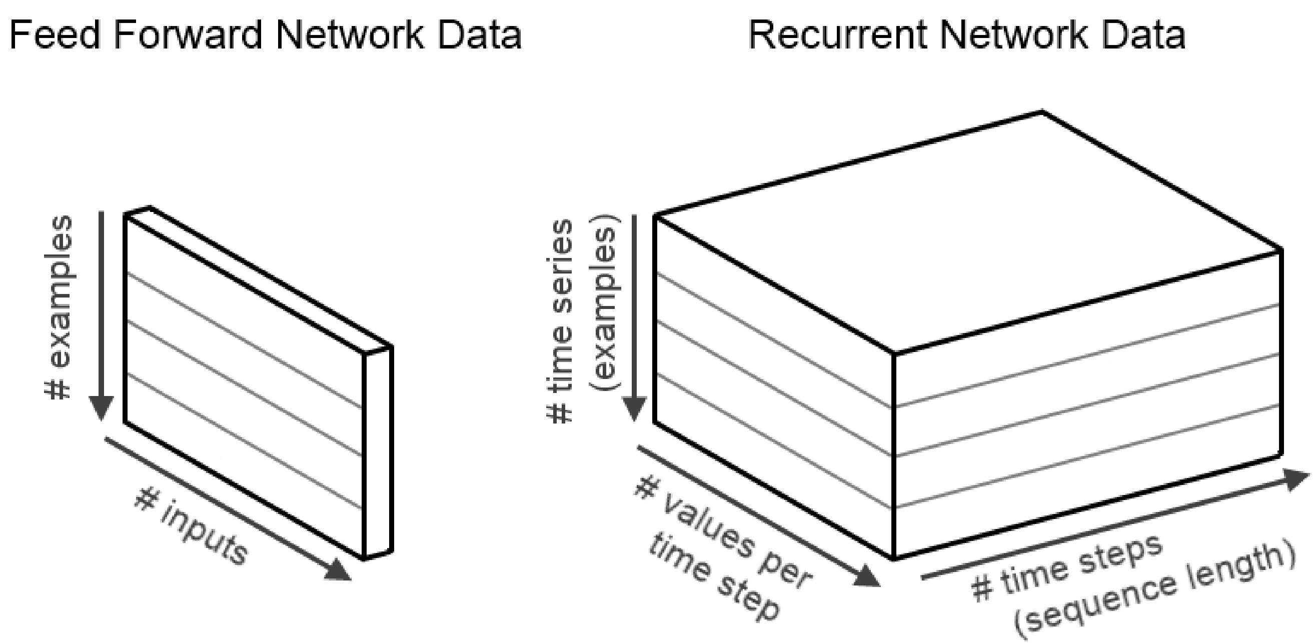
- Traditional NNs and ConvNets have an API constrained
  - input is a fixed-sized vector (e.g. an image) and output is a fixed-sized vector as output (e.g. probabilities of different classes)
- RNNs allow us to operate over *sequences* of vectors
  - Input, output or both



(1) Vanilla mode of processing without RNN, from fixed-sized input to fixed-sized output (e.g. image classification). **(2)** Sequence output (e.g. image captioning takes an image and outputs a sentence of words). **(3)** Sequence input (e.g. sentiment analysis where a given sentence is classified as expressing positive or negative sentiment). **(4)** Sequence input and sequence output (e.g. Machine Translation: an RNN reads a sentence in English and then outputs a sentence in French). **(5)** Synced sequence input and output (e.g. video classification where we wish to label each frame of the video). [9]

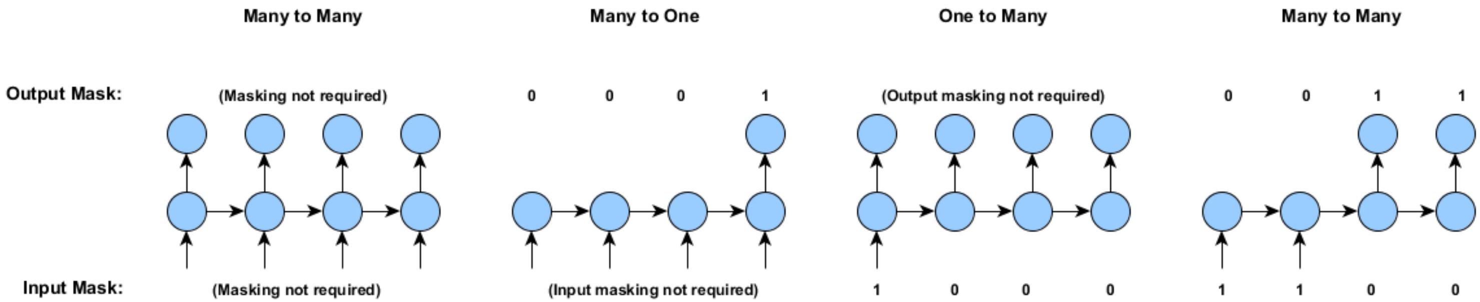
RNNs 5 [10]

- Data is more complex
  - consider a time series or sensor data as input, time gives an extra dimension
  - the value at position (i,j,k) is the jth value at the kth time step of the ith example



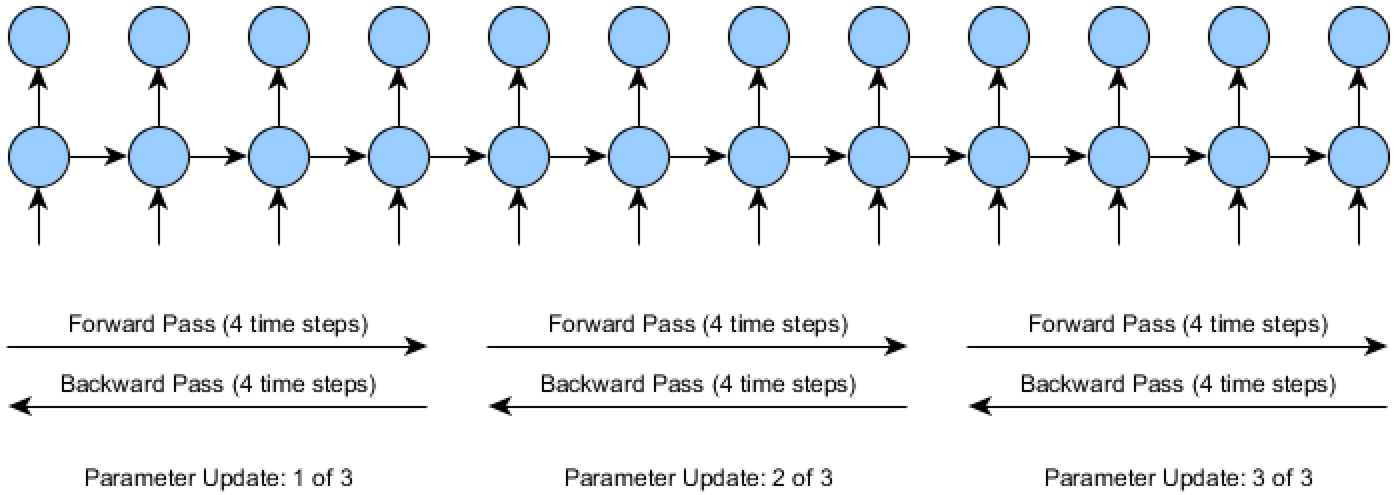
RNNs 6 [10]

- Preprocess is more complex
  - consider inputs or outputs that don't occur at every time step
    - e.g. two time series of lengths 50 and 100 time steps
- Padding and masking mechanisms can be used
  - Padding, to get training data a rectangular array, zeros are added to the shorter time series (for both input and output)
    - (in this example: 100 time steps)
  - Masking, additional arrays that record whether an input or output is actually present
    - 0 ('absent') or 1 ('present')



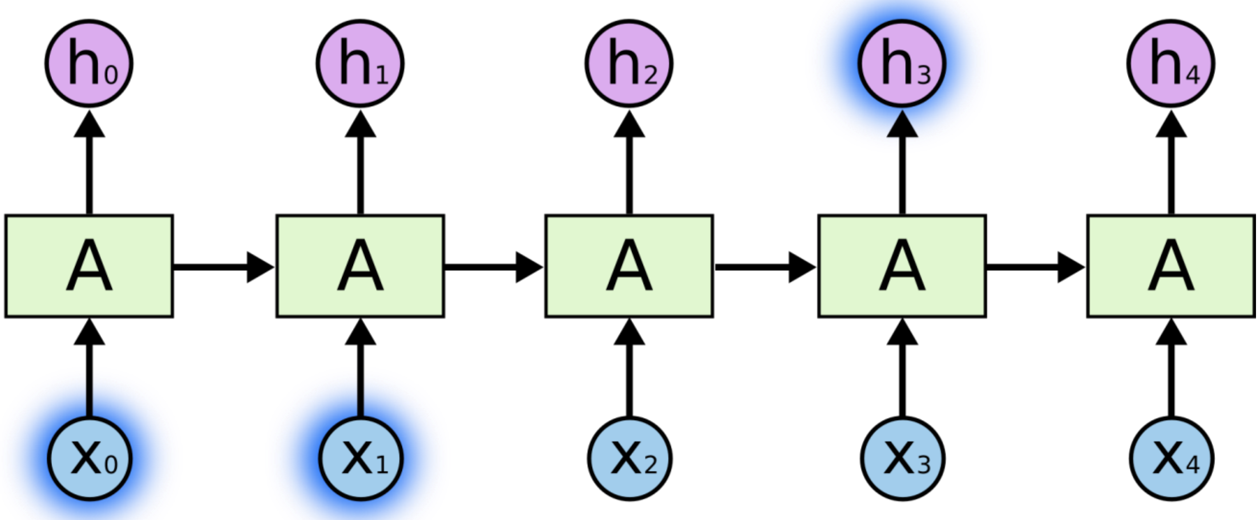
RNNs 7 [10]

- Training is more costly
  - the training of a RNN is computationally demanding
  - consider a list or time series of 10K elements
  - standard backpropagation would require 10,000 time steps for each of the forward and backward passes for each and every parameter update
  - in practice, truncated backpropagation is used
  - it splits the forward and backward passes into a set of smaller forward/backward pass operations



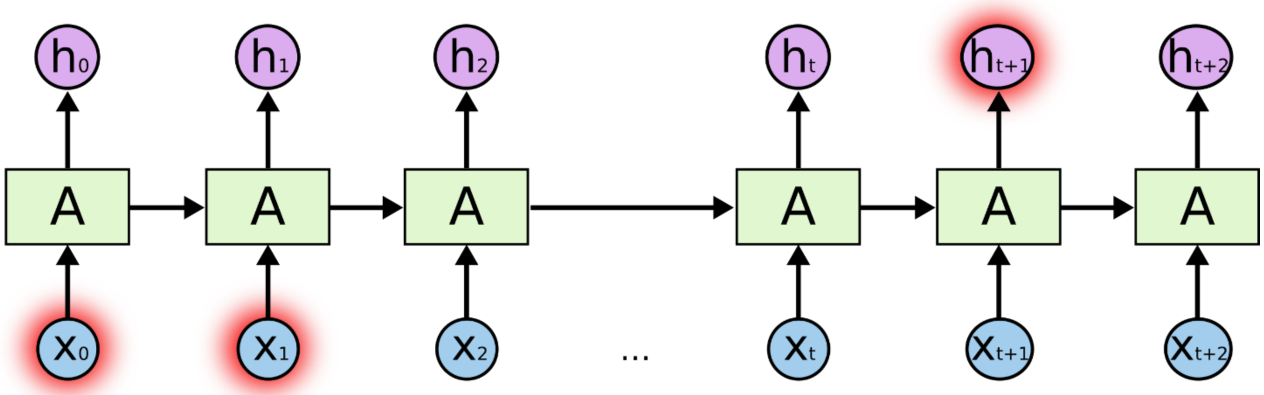
The Problem of Long-Term Dependencies [8]

- RNNs connect previous information to the present task
  - But if the gap between the relevant information and the place that it's needed is small
  - E.g., language model trying to predict the next word: "the clouds are in the *sky*"



The Problem of Long-Term Dependencies 2 [8]

- Sometimes more context is needed…
  - E.g. "I grew up in France…[long text]… I speak fluent *French*."
  - as that gap grows, RNNs become unable to learn to connect the information
  - RNNs do not handle "long-term dependencies"

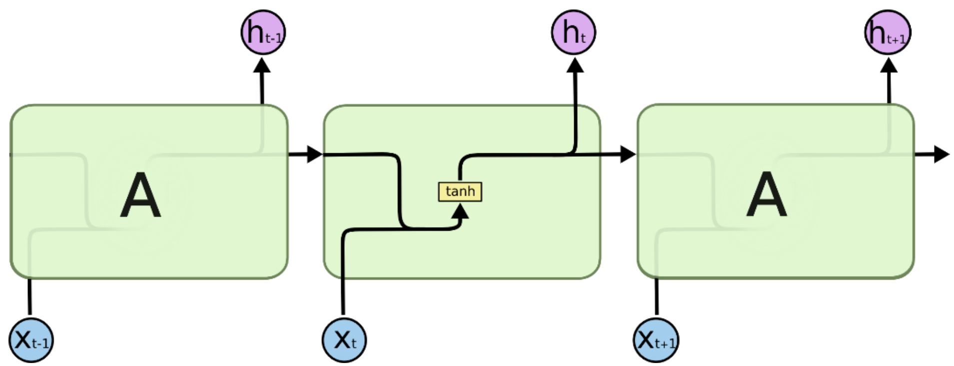


LSTMs

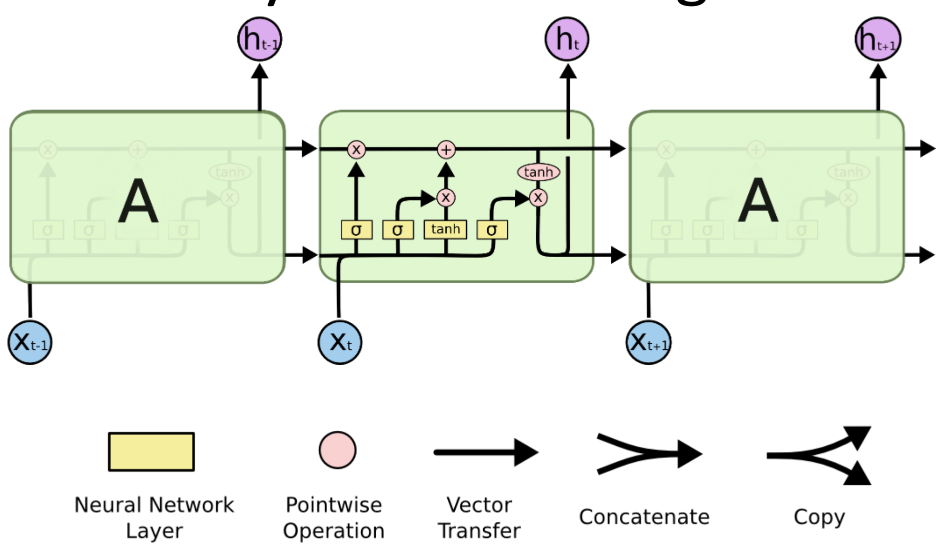
- Long Short Term Memory networks (LSTMs)
  - a special kind of RNN, capable of learning long-term dependencies
  - introduced by [Hochreiter & Schmidhuber (1997)]
  - deal with a great variety of problems
  - lots of variants (almost every paper involving LSTMs uses a slightly different version)

LSTMs 2

- The standard RNN repeating module has one layer based on a tanh (or another) activation function

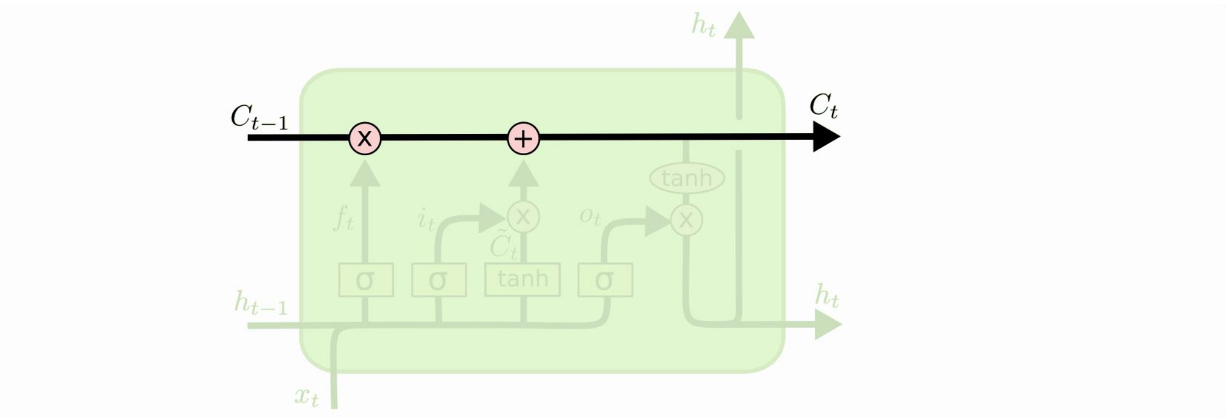


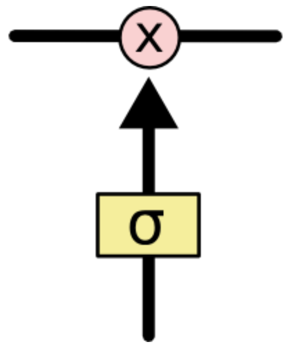
- LSTMs have four NN layers interacting in a very special way



LSTMs 3

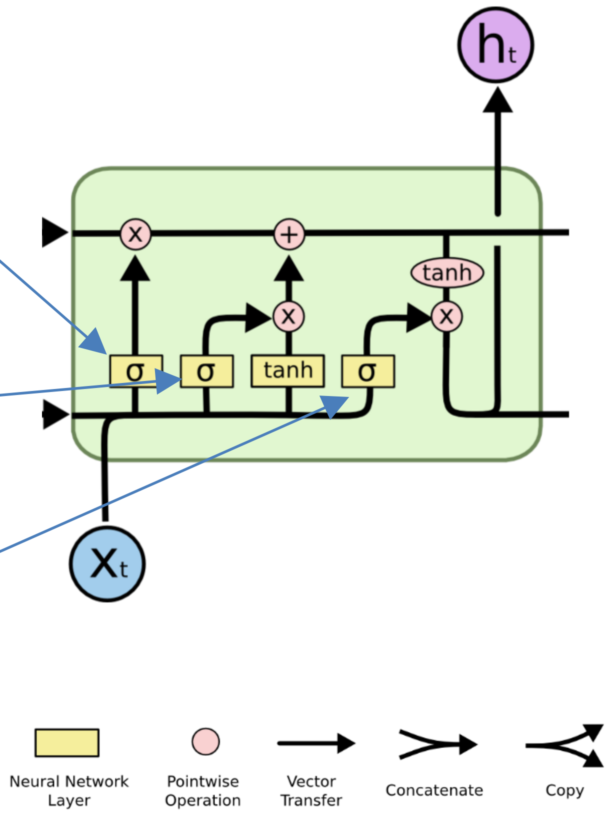
- The key of these NN is the cell state
  - acts like a conveyor belt: information easily flow along it unchanged
- LSTM can remove or add information to the cell state regulated by structures called gates
  - composed out of a sigmoid neural net layer and a pointwise multiplication operation
  - a value of zero in the gate output means “let nothing through,” while a value of one means “let everything through!”





LSTMs 4

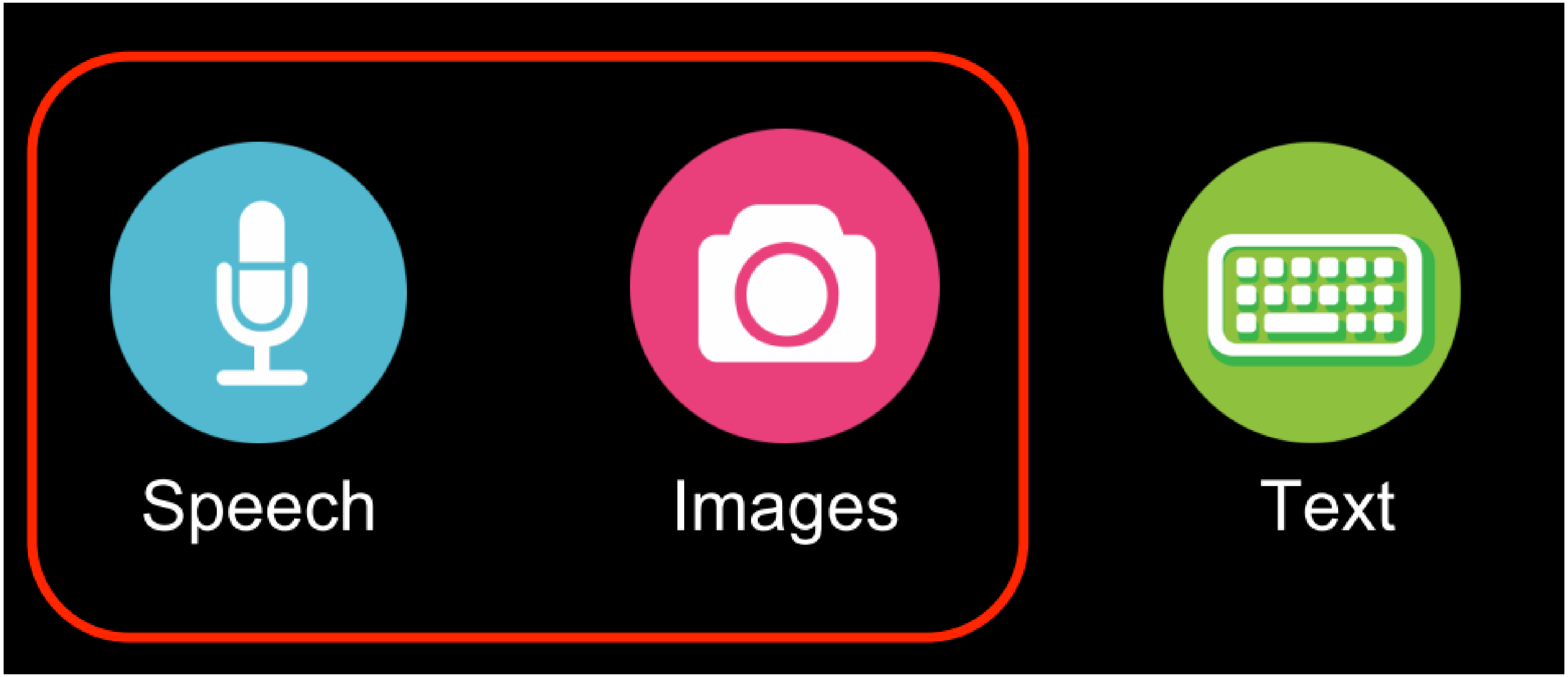
- forget gate layer
  - what information is going to be throw away from the cell state
- input gate layer
  - decides which values are updated in the cell state
  - a tanh layer creates a vector of new candidate values
  - these two outputs are combined to create an update to the state
- output gate layer
  - the output for the repeating module, a filtered version of the cell state



LSTMs 5

- Implementations of RNNs and LSTMs:
  - Recurrent Neural Networks in DL4J
  - Recurrent Neural Networks in TensorFlow

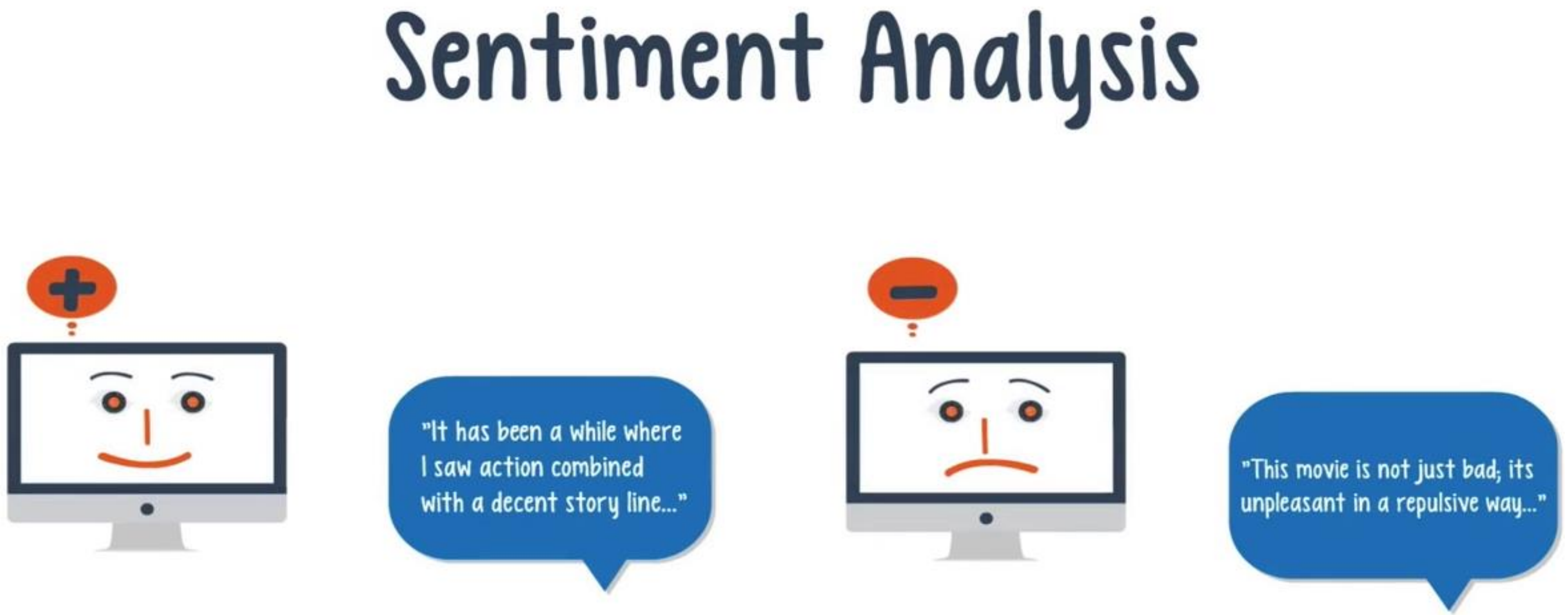
DL will transform the internet



Natural language processing

NLP

- Interactions between computers and human (natural) languages
- Challenges:
  - natural language understanding
    - enabling computers to derive meaning from human or NL input
  - natural language generation
- Classical NLP involve hand coding of large sets of rules
- NLP based on machine learning and DL
  - general learning algorithms generate these rules automatically
  - they use large corpora of typical real-world examples
    - (documents or sentences annotated with the correct values to be learned)
- Recurrent neural networks, especially LSTM, are most appropriate for sequential data such as language



- Socher, Richard (2013). "Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank"(PDF). *EMNLP 2013*.
- MetaMind Api for sentiment analysis from twitter

From DeepLearning.tv

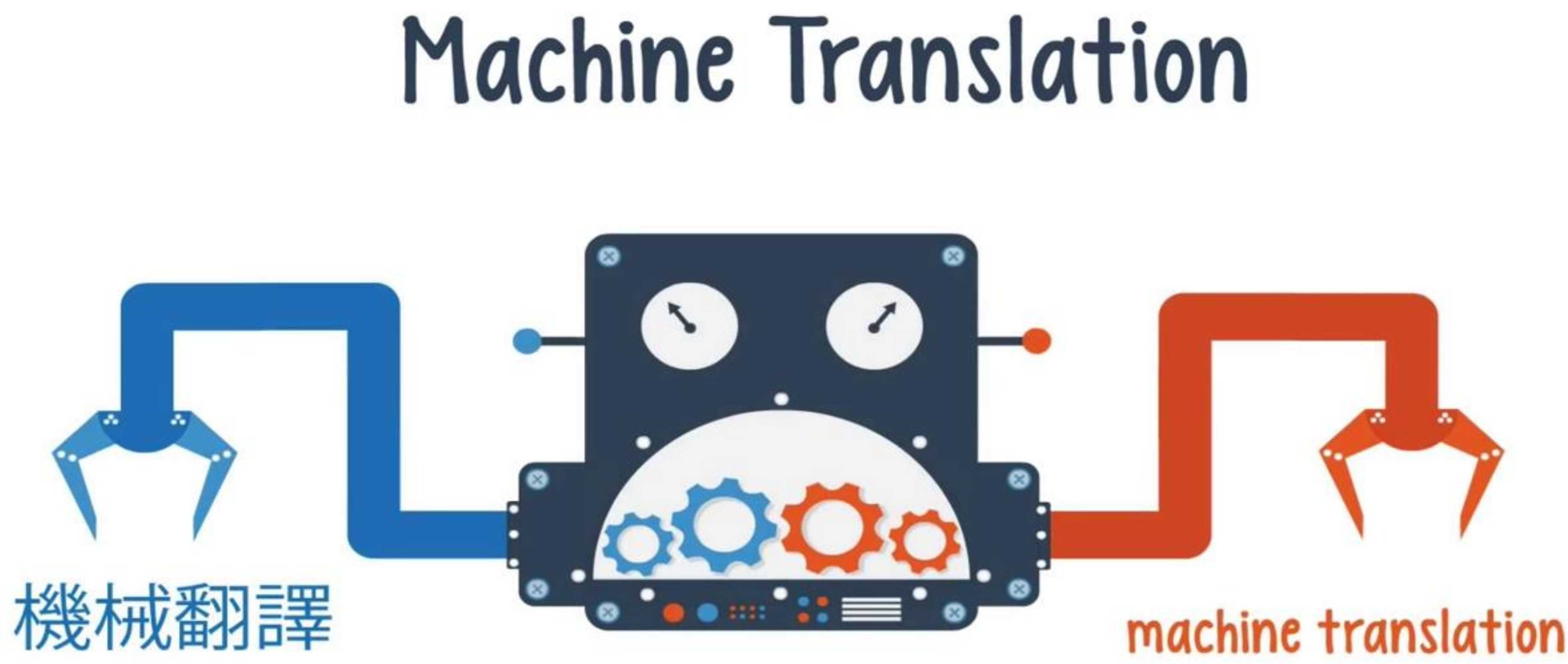
Fact Extraction

President Barack completed his tour of Asia, met with leaders, and returned to the US.

FACTS:

Obama is the president of the US.
Obama met with leaders.
Asia has leaders.

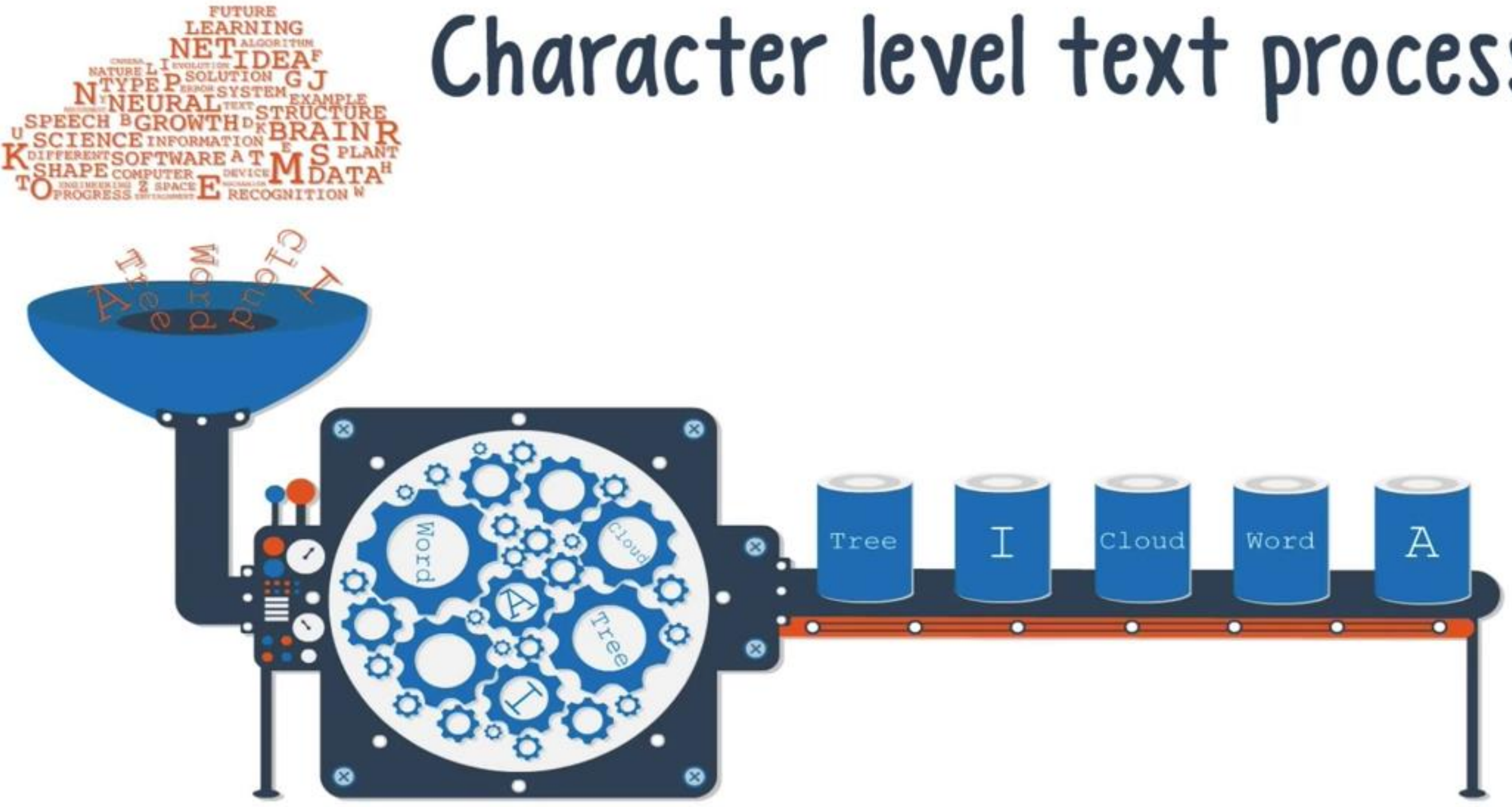
From DeepLearning.tv



1. Sutskever, O. Vinyals, Q. Le (2014) "Sequence to Sequence Learning with Neural Networks," Proc. NIPS.
2. J. Gao, X. He, W. Yih, and L. Deng(2014) "Learning Continuous Phrase Representations for Translation Modeling," Proc. ACL.

From DeepLearning.tv

Character level text processing

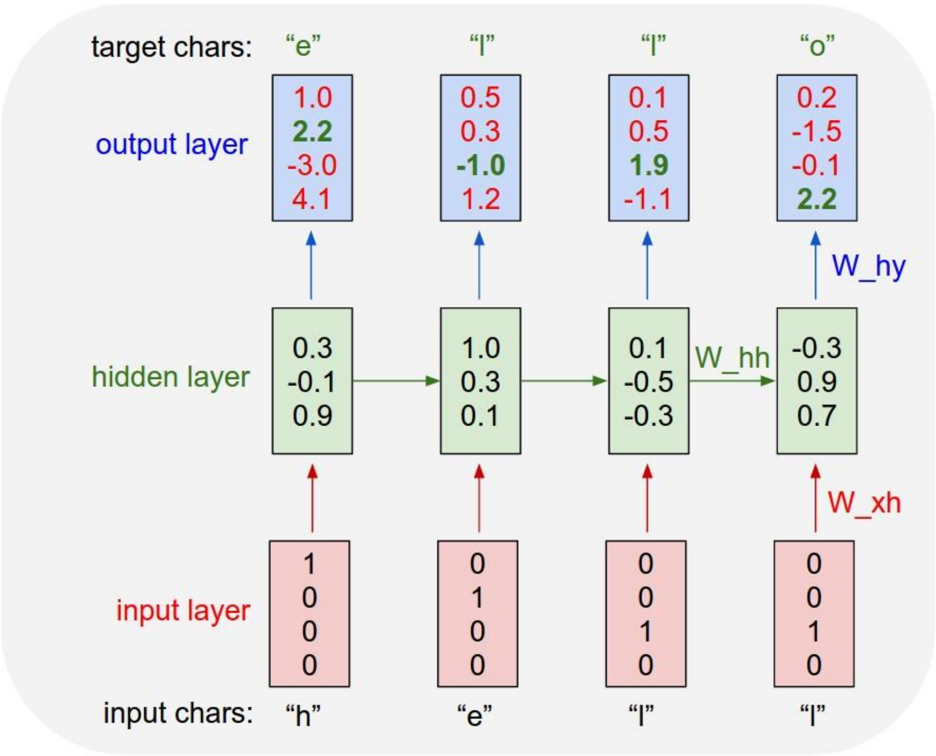


(Next slide)

From [DeepLearning.tv]

Character-level Language model example [9]

- Consider a vocabulary of 4 possible letters "helo”
- A RNN can be trained to get the probability distribution of the next character in the sequence given a sequence of previous characters
- This allow us to generate new text one character at a time



RNN with 4-dimensional input and output layers, and a hidden layer of 3 units (neurons).
This diagram shows the activations in the forward pass when the RNN is fed the characters "hell" as input.
The output layer contains confidences the RNN assigns for the next character (vocabulary is "h,e,l,o");
We want the green numbers to be high and red numbers to be low.

Shakespeare text generation [9]

- 3-layer RNN with 512 hidden nodes
- Input: All the works of Shakespeare and concatenated them into a file
- Few hours of training
- And we obtained samples such as:

PANDARUS:
Alas, I think he shall be come approached and the day
When little srain would be attain'd into being never fed,
And who is but a chain and subjects of his death,
I should not sleep.

Second Senator:
They are away this miseries, produced upon my soul,
Breaking and strongly should be buried, when I perish
The earth and thoughts of many states.

DUKE VINCENTIO:
Well, your wit is in the care of side and that.

Second Lord:
They would be ruled after this chamber, and
my fair nues begun out of the fact, to be conveyed,
Whose noble souls I'll have the heart of the wars.

Clown:
Come, sir, I will make did behold your worship.

VIOLA:
I'll drink it.

George R.R. Martin text generation

GAME OF THRONES

- *He leaned close and onions, barefoot from his shoulder. "I am not a purple girl," he said as he stood over him. "The sight of you sell your father with you a little choice."*
- *"I say to swear up his sea or a boy of stone and heart, down," Lord Tywin said. "I love your word or her to me."*
- *"Lord Tywin would have worn them and see my commands." He took the cloth of dragons for his brother to take the crow to the other side of the gallery and the wailing helps her maidenhead of the only trunk of the dead men loved by the time they had come to rest. The red priest was pleased to be done with a smile to show them from the beginning lands and red and strong, and the stables she had been king on the water, and the child will be lost in the night before his father's son. Even his brothers made her way for him and the king to take the swift of the king. The first time will take me for our command," Ned said. "Before an inn, the Sons of the Lord of Light, we must be good beard."*

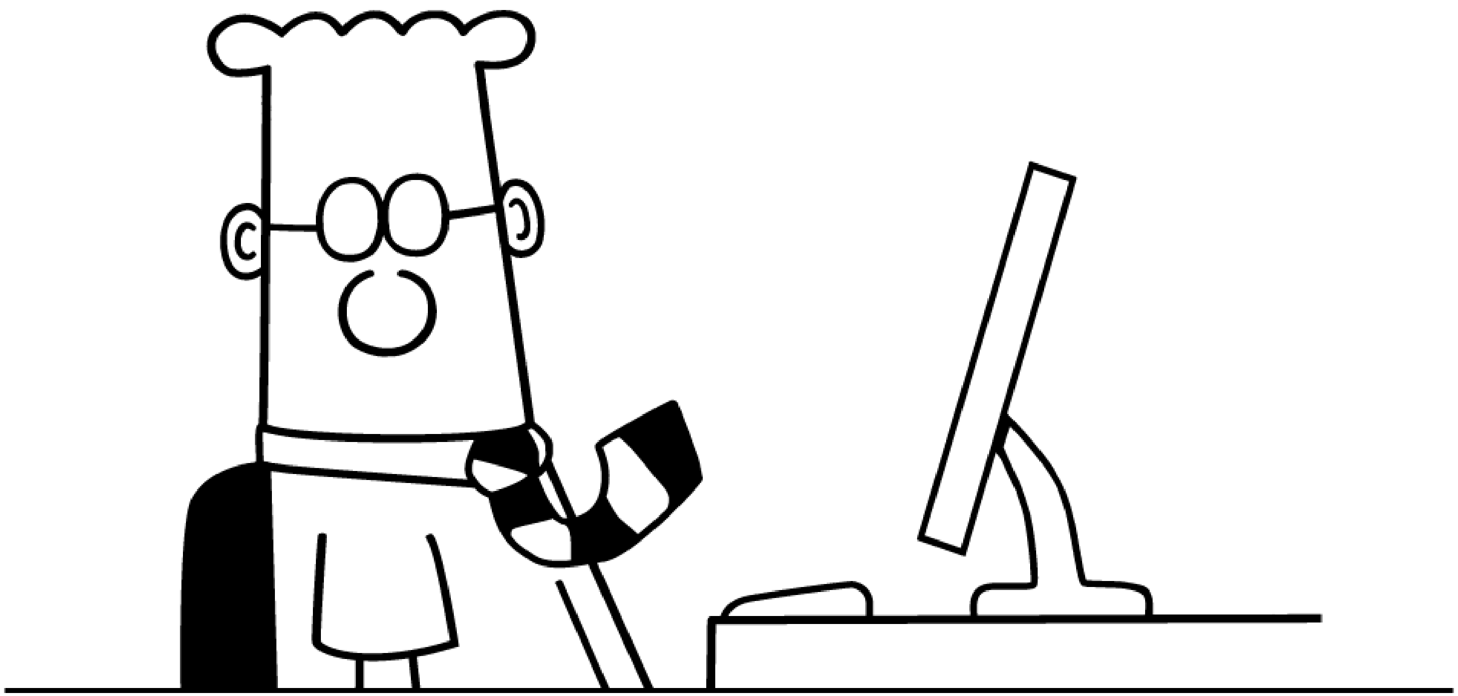
http://pjreddie.com/darknet/rnns-in-darknet/ *(also* Shakespeare, Tolstoy, and Kant style text)

BabyNames [9]

- RNN trained from 8000 baby names

Rudi Levette Berice Lussa Hany Mareanne Chrestina Carissy Marylen Hammine Janye Marlise Jacacrie Hendred Romand Charienna Nenotto Ette Dorane Wallen Marly Darine Salina Elvyn Ersia Maralena Minoria Ellia Charmin Antley Nerille Chelon Walmor Evena Jeryly Stachon Charisa Allisa Anatha Cathanie Geetra Alexie Jerin Cassen Herbett Cossie Velen Daurenge Robester Shermond Terisa Licia Roselen Ferine Jayn Lusine Charyanne Sales Sanny Resa Wallon Martine Merus Jelen Candica Wallin Tel Rachene Tarine Ozila Ketia Shanne Arnande Karella Roselina Alessia Chasty Deland Berther Geamar Jackein Mellisand Sagdy Nenc Lessie Rasemy Guen Gavi Milea Anneda Margoris Janin Rodelin Zeanna Elyne Janah Ferzina Susta Pey Castina

Creativity is allowing yourself to make mistakes. Art is knowing which ones to keep. ~Scott Adams



Wikipedia markdown syntax [9]

- LSTM trained with Wikipedia markdown code
- the model learns to open and close the parenthesis correctly
- it creates headings, lists, etc.
- (of course it makes up URLs and others)
- also for latex and Linux source code

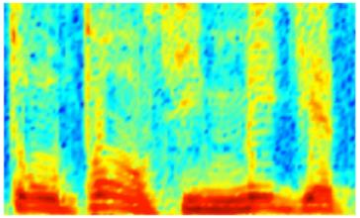
…and plenty more applications

- learning and guessing passwords
  - https://github.com/gehaxelt/RNN-Passwords
- automated grading of student-written essays
  - https://www.kaggle.com/c/asap-aes
- sentence similarity and paraphrasing detection
  - Socher, Richard; Manning, Christopher. "Deep Learning for NLP" Retrieved 26 October 2014
- constituency parsing
  - (associate grammar rules with a probability)
  - Socher, Richard; Bauer, John; Manning, Christopher; Ng, Andrew (2013). "Parsing With Compositional Vector Grammars" *Proceedings of the ACL 2013 conference*.
- information retrieval
  - (obtaining resources relevant to an information need from a collection of information resources)
  - Y. Shen, X. He, J. Gao, L. Deng, and G. Mesnil (2014) " A Latent Semantic Model with Convolutional-Pooling Structure for Information Retrieval," Proc. CIKM.

Word2Vec, motivation

- Image and audio learning systems inputs are encoded as vectors
  - all the information required is econded in the raw data
  - e.g. raw pixel-intensities (image), or power spectral density coefficients (audio)
- Natural language representations are arbitrary
  - 'cat' may be Id537 and 'dog'=Id143
  - no useful information regarding the relationships
  - when learning about dogs, DL can't use anything learnt about cats (they are both animals, four-legged, pets…)

AUDIO



Audio Spectrogram

DENSE

IMAGES

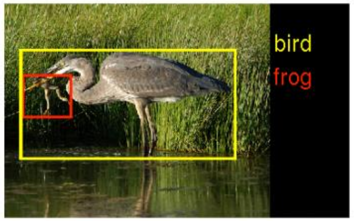


Image pixels

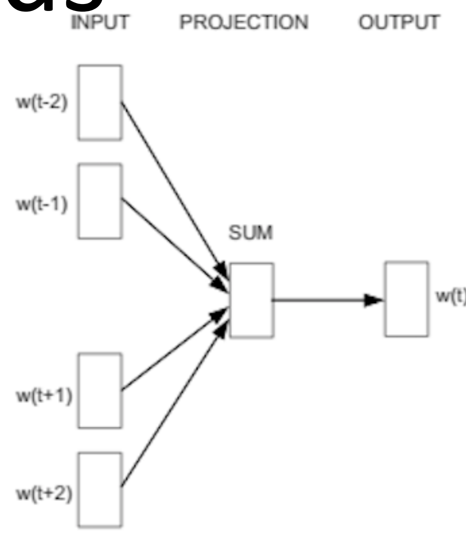
DENSE

Word2Vec [13]

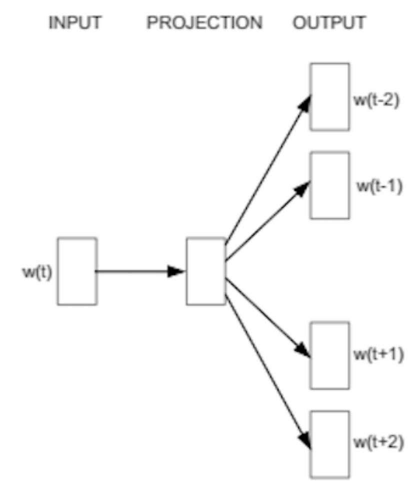
- Vector space models (VSMs) represent (embed) words in a continuous vector space
  - semantically similar words are mapped to nearby points
  - all methods relies on distributional Hypothesis
    - words that appear in the same contexts share semantic meaning.
- Word2vec is a two-layer neural net that processes text
  - input: text corpus
  - output: vocabulary in which each item has a vector attached to it
    - can be fed into a DNN
    - or queried to detect relationships between words
- Not a DNN, but allows DNNs to understand text
  - can be seen as a representational layer in DL
  - not just for sentences: genes, code, playlists, social media graphs and other verbal or symbolic series

Word2Vec, methods

- ❑ Word2Vec is similar to an autoencoder
  - ❑ but not trained against the input words for reconstruction…
  - ❑ it trains words against other words that neighbor them in the input corpus
  - ❑ well trained if similar words are close to each other in that space
- ❑ 2 methods or flavors:
  - ❑ Continuous bag of words (CBOW)
    - ❑ use context to predict a target word
    - ❑ from ('the cat sits on the'), predicts 'mat'
    - ❑ good for smaller datasets (entire context as one observation)
  - ❑ Skip-gram
    - ❑ use word to predict a target context
    - ❑ from(mat'), predicts 'the cat sits on the'
    - ❑ good for larger datasets (each context-target pair as a new observation)



CBOW



Skip-gram

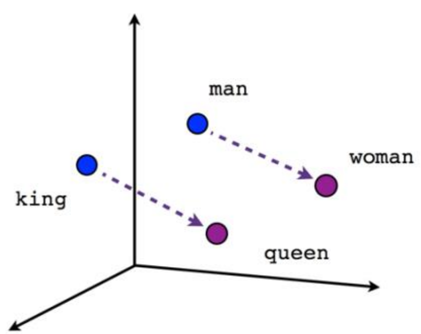
Word2Vec, similarity

- Cosine similarity can give as similarity rankings
  - Parallel unit vectors have an inner product of 1 (completely similar)
  - Perpendicular unit vectors of 0 (completely alike)
- Words associated with "Sweden":

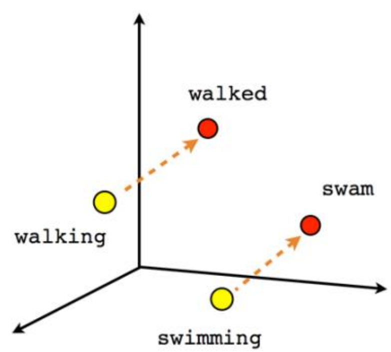
Word	Cosine distance
norway	0.760124
denmark	0.715460
finland	0.620022
switzerland	0.588132
belgium	0.585835
netherlands	0.574631
iceland	0.562368
estonia	0.547621
slovenia	0.531408

Word2Vec, more relationships

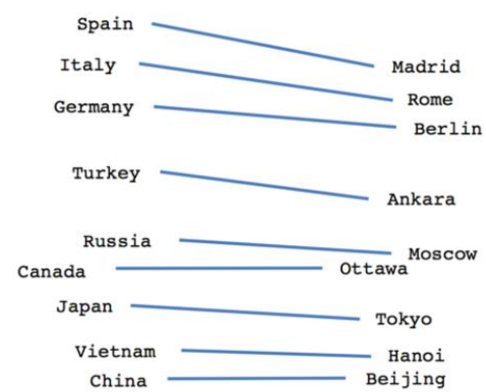
- similarity is the basis of many associations that Word2vec can learn
- the learned vectors can be visualized after dimensionality reduction (as t-SNE)
- certain directions in the induced vector space specialize towards certain semantic relationships
  - e.g. male-female, gender and even country-capital relationships



Male-Female



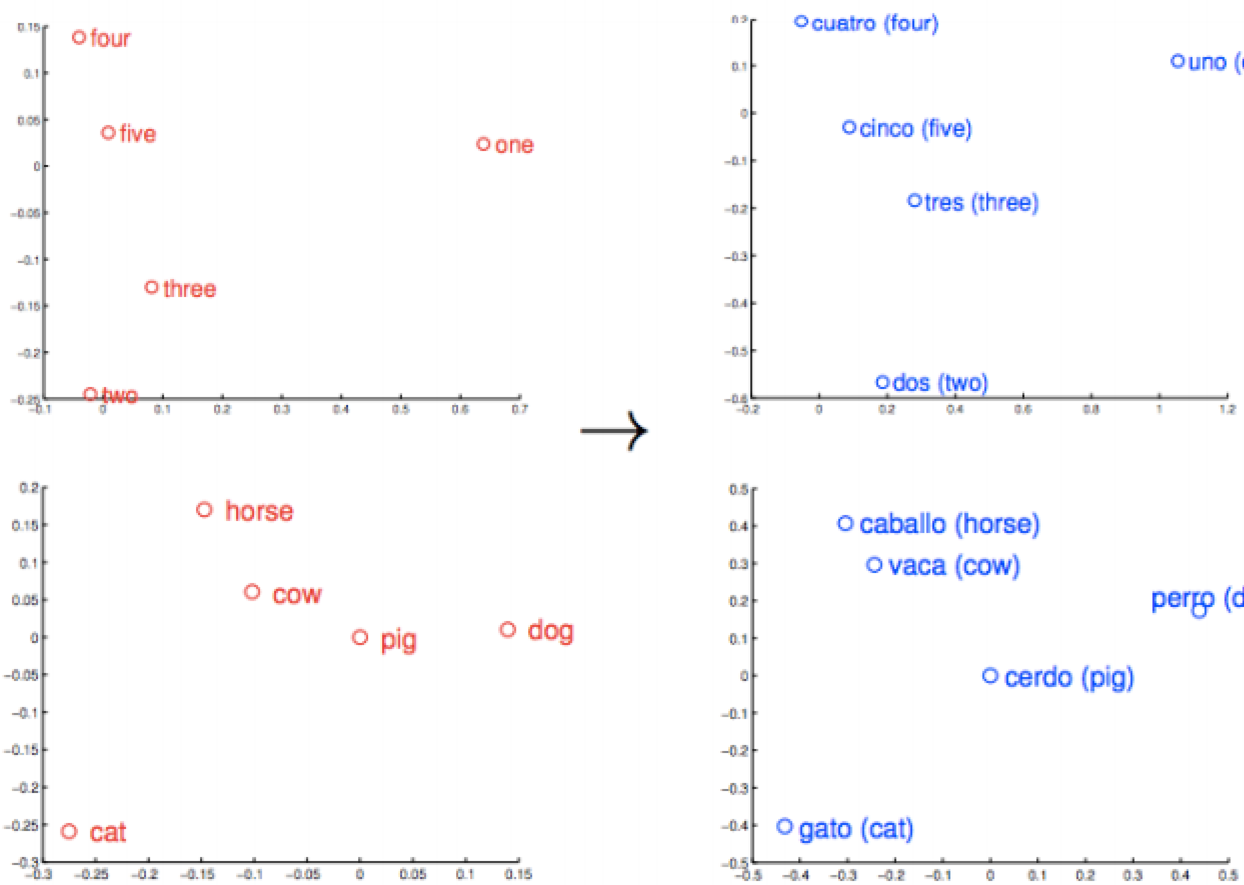
Verb tense



Country-Capital

Word2Vec, more relationships 2

- Mapping relations between words of two languages



Word2Vec, querying [13]

- Model trained on the Google News vocab
  - Query syntax: "Rome is to Italy as China is to Beijing" = Rome:Italy::Beijing:China

  - `love:indifference::fear:[apathy, callousness, timidity, helplessness, inaction]`
  - `king:queen::man:[woman, Attempted abduction, teenager, girl]`
  - `China:Taiwan::Russia:[Ukraine, Moscow, Moldova, Armenia]`
  - `house:roof::castle:[dome, bell_tower, spire, crenellations, turrets]`
  - `knee:leg::elbow:[forearm, arm, ulna_bone]`
  - `Donald Trump:Republican::Barack Obama:[Democratic, GOP, Democrats, McCain]`
  - `monkey:human::dinosaur:[fossil, fossilized, Ice_Age_mammals, fossilization]`

Word2vec algorithm has never been taught a single rule of English syntax!
It comes to the Google News documents as a blank slate!

Word2Vec implementations

- Apache 2.0 license and patented by Google in 2015
  - Word2Vec in DL4J
  - Word2Vec in TensorFlow

Conclusion

- DL is replacing previous approaches in many fields
- Same applications as ML
  - the general questions ML can solve are essential
- There is not a single and accepted taxonomy for DL applications
- …but speech recognition and vision are the most extended fields
- ConvNets are widely used for vision and LSTM RNN for speech recognition and NLP
- Word2Vec is a popular representation layer in NLP problems
- DL will transform internet: from text based queries to images/sounds queries

Recommended videos

- [Two+ Minute Papers - 9 Cool Deep Learning Applications]
- [Two Minute Papers - 10 More Cool Deep Learning Applications]
- [Use Cases - Ep. 12 (Deep Learning SIMPLIFIED)]
- [Geoffrey Hinton: "Some Applications of Deep Learning"]

Recommended online tools

- A Neural Network playground:
  - http://playground.tensorflow.org/
- ConvNet visualization
  - http://scs.ryerson.ca/~aharley/vis/conv/
- Visual recognition:
  - www.clarifie.com

References

[1] What questions can data science answer?. By Brandon Rohrer, Microsoft.

[2] Which Algorithm Family Can Answer My Question?. By Brandon Rohrer, Microsoft.

[3] Machine Learning: 2014-2015. University of Oxford. Lecture 15: Reinforcement learning with direct policy search. Nando de Freitas.

[4] Human-level control through deep reinforcement learning. Volodymyr Mnih et al. Nature 518, 529–533 (26 February 2015)

[5] A Neural Algorithm of Artistic Style. Leon A. Gatys, Alexander S. Ecker, Matthias Bethge

[6] What a Deep Neural Network thinks about your #selfie. Andrej Karpathy blog.

[7] The neural networks behind Google Voice transcription.Posted by Françoise Beaufays, Research Scientist.

[8] Understanding LSTM Networks. Christopher Olah.

[9] The Unreasonable Effectiveness of Recurrent Neural Networks. Andrej Karpathy.

[10] Recurrent Neural Networks in DL4J. link

[11] Deng, L.; Yu, D. (2014). "Deep Learning: Methods and Applications" (PDF). Foundations and Trends in Signal Processing 7: 3–4. doi:10.1561/2000000039.

[12] Deep Learning. Wikipedia. Link

[13] Word2Vec in DL4J. link