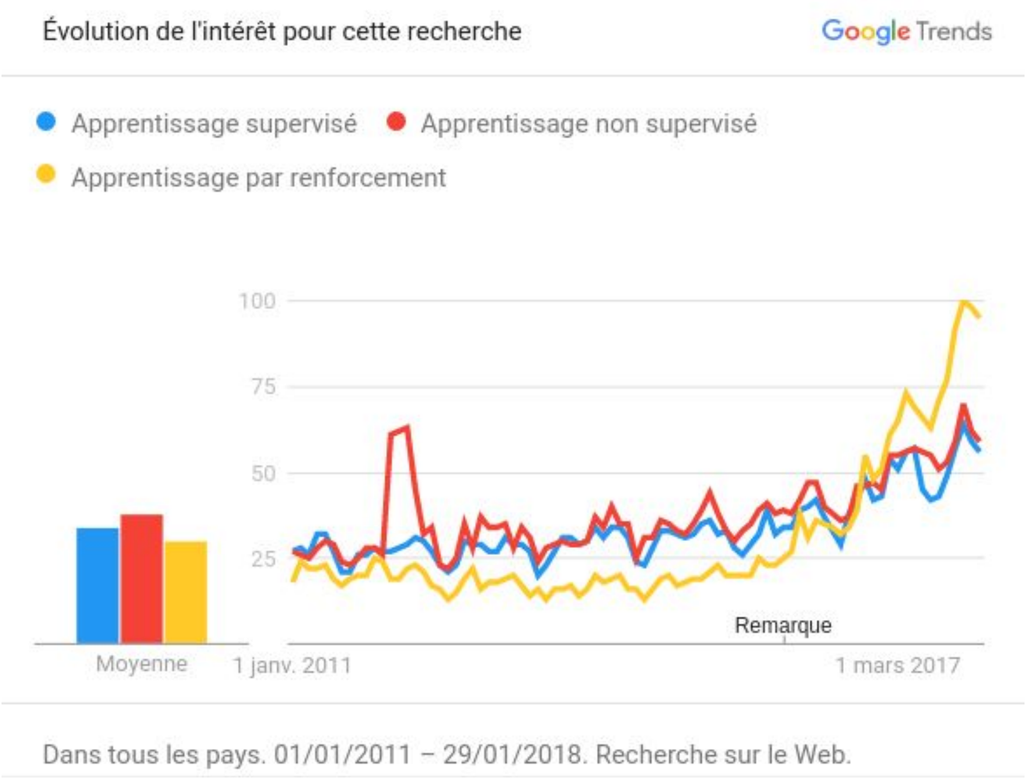
INTRODUCTION TO DEEP LEARNING

Mouhidine SEIV
FOUNDER - RIMINDER

Deep LEARNING A PRACTICAL COURSE
ECOLE POLYTECHNIQUE, 05/04/2018

History of Learning Systems



riminder https://github.com/riminder

Program & Course Logistics

- **Course 1 :** (05-04-18)
    - Introduction to Deep Learning - Mouhidine SEIV (Riminder)
- **Course 2 :** (12-04-18)
    - Deep Learning in Computer Vision - Slim FRIKHA (Riminder)
- **Course 3 :** (19-04-18)
    - Deep Learning in NLP - Paul COURSAUX (Riminder)
- **Course 4 :** (26-04-18)
    - Efficient Methods and Compression for Deep Learning - INVITED GUEST
- **Course 5:** (03-05-18)
    - Introduction to Deep Learning Frameworks - INVITED GUEST
- **Course 6:** (10-05-18)
    - Deployment in Production and Parallel Computing - INVITED GUEST

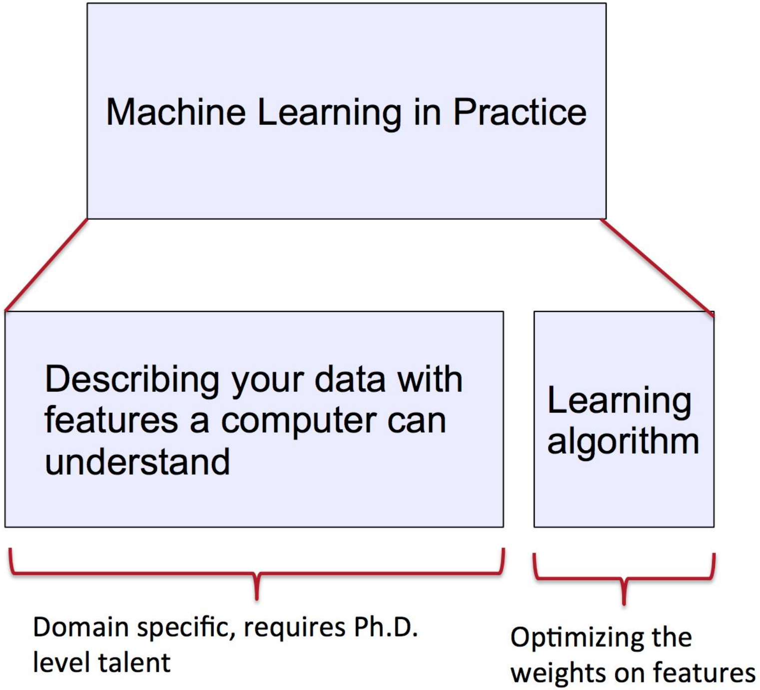
Location: Ecole Polytechnique from 6:30 pm to 7:30pm

riminder https://github.com/riminder

Prerequisites

- Proficiency in python
- Linear algebra
- Basic probability and statistics
- Basic machine learning (cost functions, derivatives , gradient methods)

Machine Learning vs Deep Learning



Reasons for Exploring Deep Learning

- In 2006 deep learning techniques started outperforming other Machine learning techniques. Why now?

- DL techniques benefit more from a lot of data
- Faster machines and multicore CPU/GPU help DL
- New models, algorithms, ideas

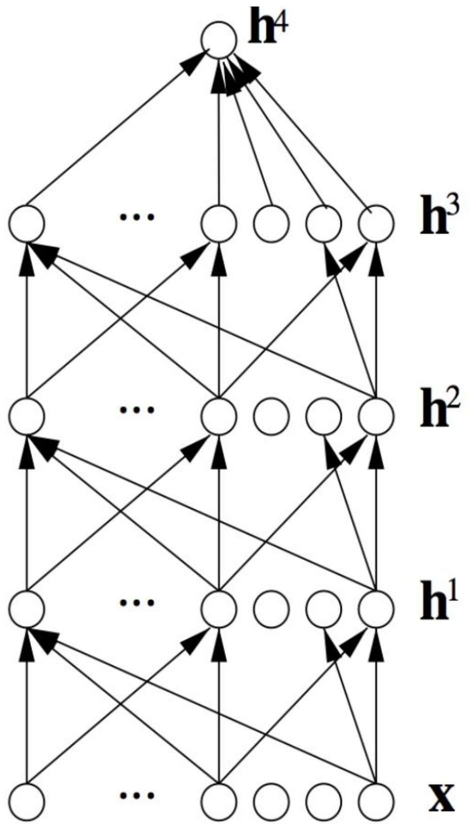
→ **Improved performance (first in speech and vision, then NLP)**

What is Deep Learning?

- Representation learning attempts to automatically learn good features or representations

- Deep learning algorithms attempt to learn (multiple levels of) representation and an outputs

- From “raw” inputs **x** (e.g. words)



Deep Representations

- A deep representation is a composition of many functions

$$
\begin{tikzcd}
x \arrow[r] & h_1 \arrow[r] & \dots \arrow[r] & h_n \arrow[r] & y \arrow[r] & l \\
& w_1 \arrow[u] & \dots & w_n \arrow[u] & &
\end{tikzcd}
$$

- Its gradient can be backpropagated by the chain rule

$$
\begin{tikzcd}
\frac{\partial l}{\partial x} & \frac{\partial l}{\partial h_1} \arrow[l, "\frac{\partial h_1}{\partial x}"'] \arrow[d, "\frac{\partial h_1}{\partial w_1}"'] & \dots \arrow[l, "\frac{\partial h_2}{\partial h_1}"'] & \frac{\partial l}{\partial h_n} \arrow[l, "\frac{\partial h_n}{\partial h_{n-1}}"'] \arrow[d, "\frac{\partial h_n}{\partial w_n}"'] & \frac{\partial l}{\partial y} \arrow[l, "\frac{\partial y}{\partial h_n}"'] \\
& \frac{\partial l}{\partial w_1} & \dots & \frac{\partial l}{\partial w_n} &
\end{tikzcd}
$$

Deep Neural Network

- A **deep neural network** is typically composed of:
  - Linear transformations

$$h_{k+1} = Wh_k$$

- Non-linear activation functions

$$h_{k+2} = f(h_{k+1})$$

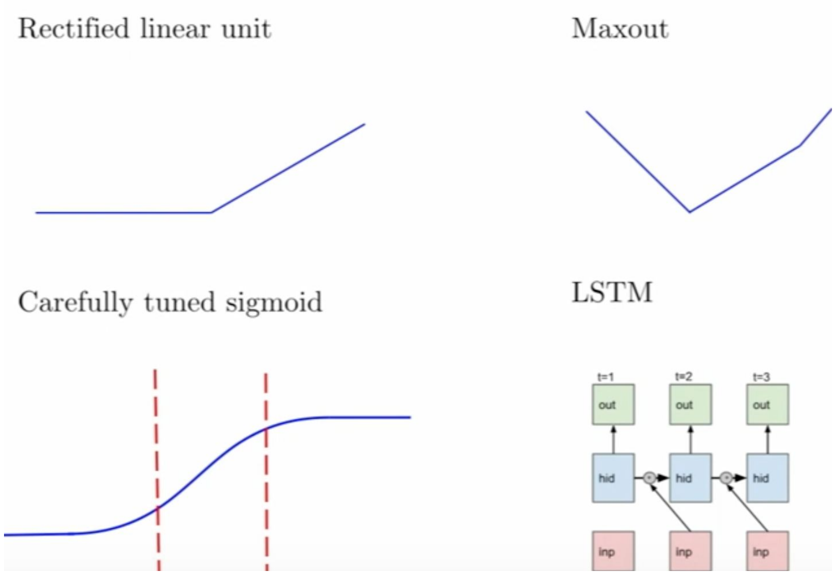
- A loss function on the output, e.g.
  - Mean-squared error

$$l = ||y^* - y||^2$$

  - Log likelihood

$$l = \log \mathbb{P}[y^*]$$

Non-linear activation functions

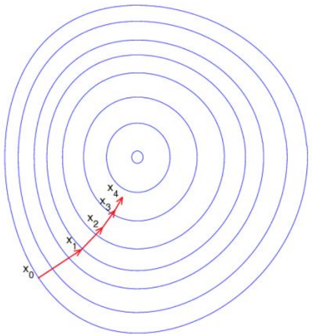


riminder https://github.com/riminder

Training Neural Networks

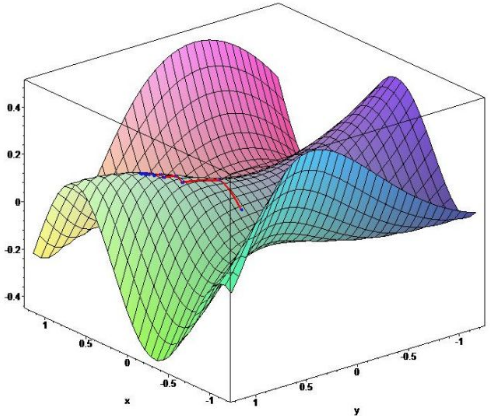
- Sample gradient of expected loss L(w) = E [l]

$$\frac{\partial l}{\partial \mathbf{w}} \sim \mathbb{E}\left[\frac{\partial l}{\partial \mathbf{w}}\right] = \frac{\partial L(\mathbf{w})}{\partial \mathbf{w}}$$



- Adjust **w** down the sampled gradient

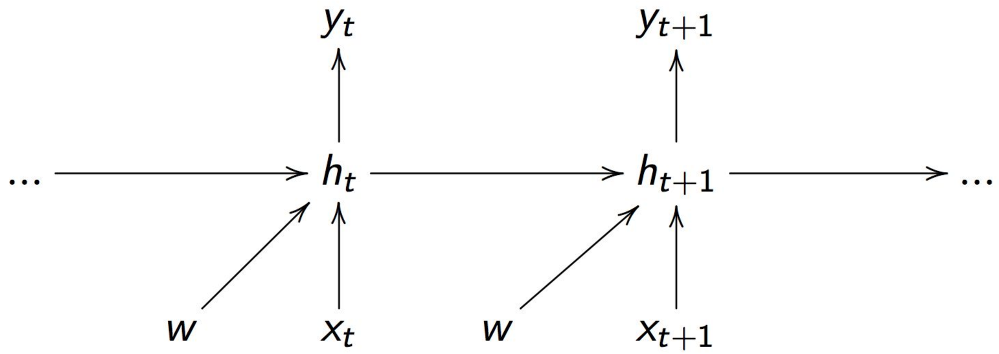
$$\Delta w \propto \frac{\partial l}{\partial \mathbf{w}}$$

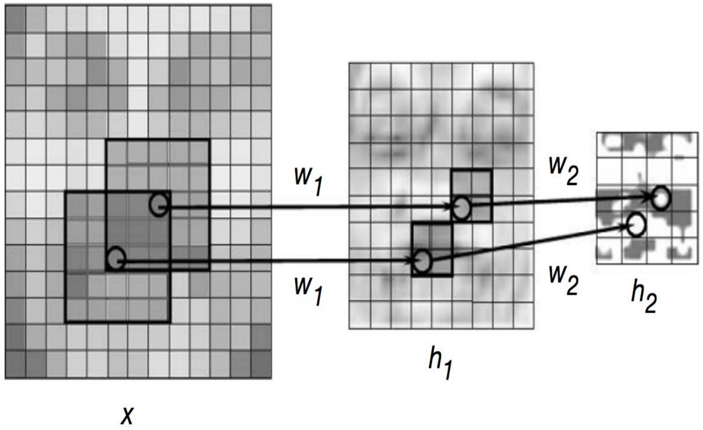


riminder https://github.com/riminder

Weight Sharing

- Recurrent neural network
  shares weights between time-steps





- Convolutional neural network
  shares weights between local regions

What is NLP?

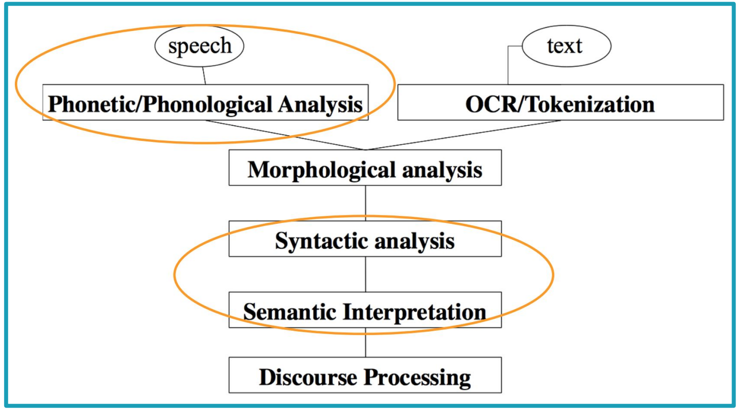
- Natural language processing is a field at the intersection of
    - Computer science
    - Artificial intelligence
    - and linguistics.

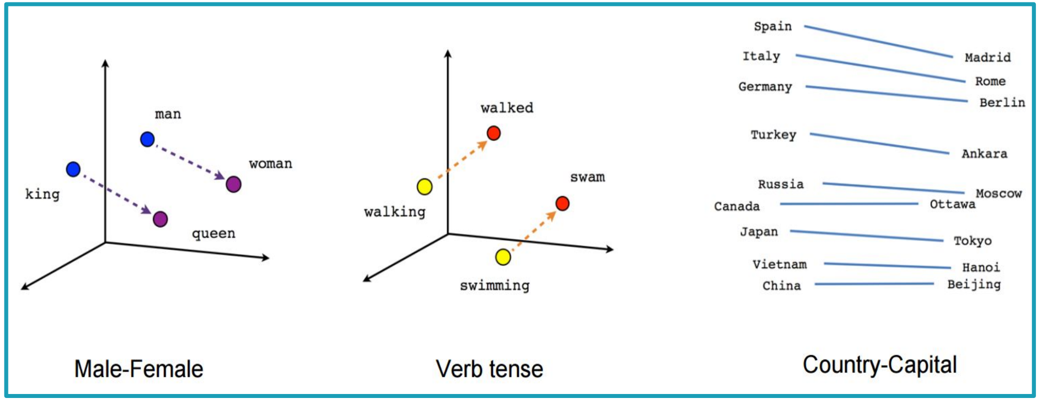
- Goal: for computers to process or “understand” natural language inorder to perform tasks that are useful, e.g.
    - Question Answering

- Fully understanding and representing the meaning of language (or even defining it) is an illusive goal.
- Perfect language understanding is AI-complete

riminder https://github.com/riminder

Deep Learning for NLP





BEFORE Deep Learning

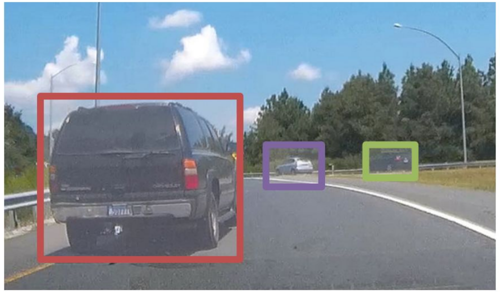
After Deep Learning

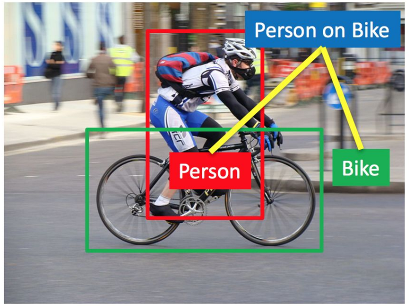
riminder https://github.com/riminder

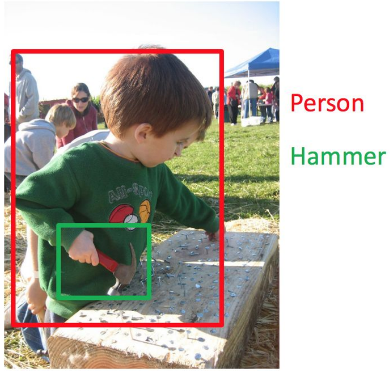
What is Computer Vision?

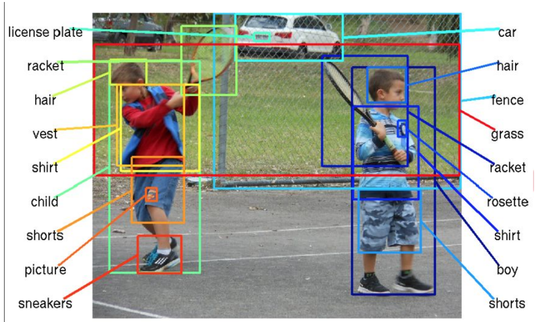
- Most deep learning groups have (until 2 years ago) focused on computer vision

- Break through paper: ImageNet Classification with Deep Convolutional Neural Networks by Krizhevsky et al. 2012









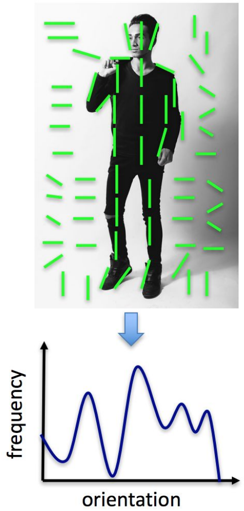
https://github.com/riminder

Deep Learning for Computer Vision

BEFORE Deep Learning

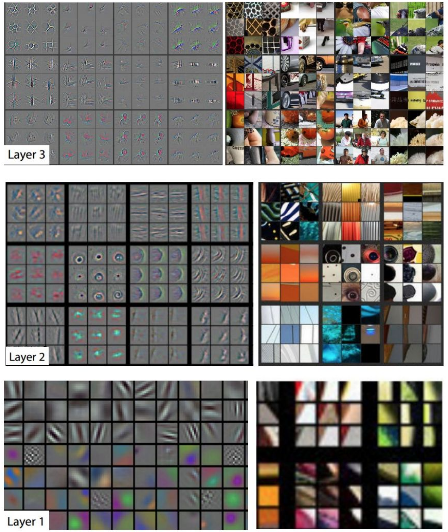
After Deep Learning

Image is CC0 1.0 public domain



Histogram of Gradients (HoG)
Dalal & Triggs, 2005

Deformable Part Model
Felzenswalb, McAllester, Ramanan, 2009



Zeiler and Fergus (2013)

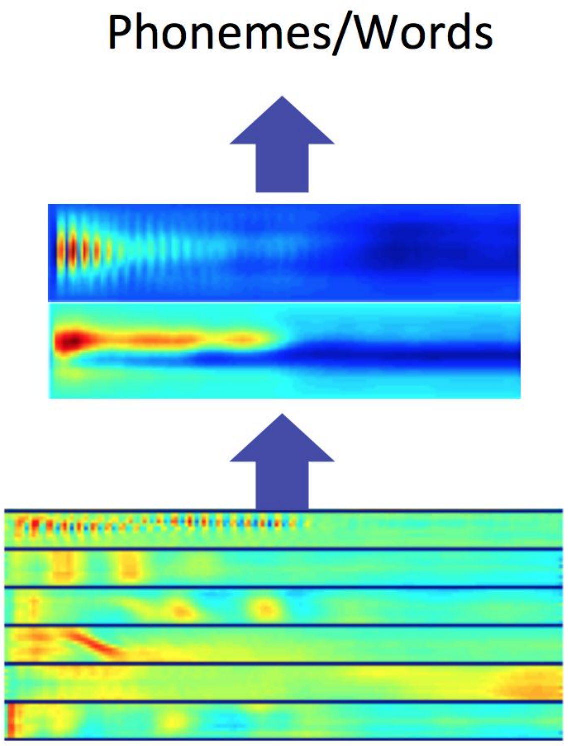
riminder https://github.com/riminder

Deep Learning for Speech

- The first breakthrough results of “deep learning” on large datasets happened in speech recognition

- Context-Dependent Pre-trained Deep Neural Networks for Large Vocabulary Speech Recognition Dahl al. (2010)

Acoustic model	Recog \ WER	RT03S FSH	Hub5 SWB
Traditional features	1-pass −adapt	**27.4**	**23.6**
Deep Learning	1-pass −adapt	**18.5** (−33%)	**16.1** (−32%)

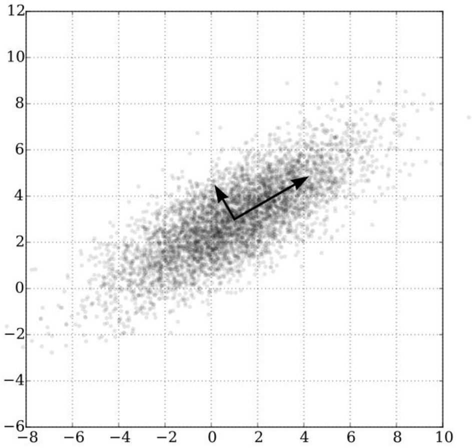


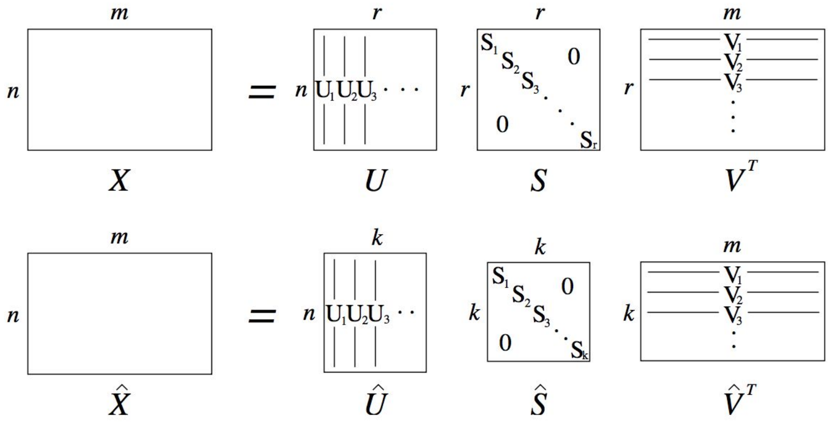
riminder https://github.com/riminder

Visualization

- Singular Value Decomposition (SVD)

- Principal component analysis (PCA)





$\hat{X}$ is the best rank k approximation to X, in terms of least squares.

- t-distributed stochastic neighbor embedding (T-SNE)
  https://distill.pub/2016/misread-tsne/

Singular Value Decomposition (SVD)

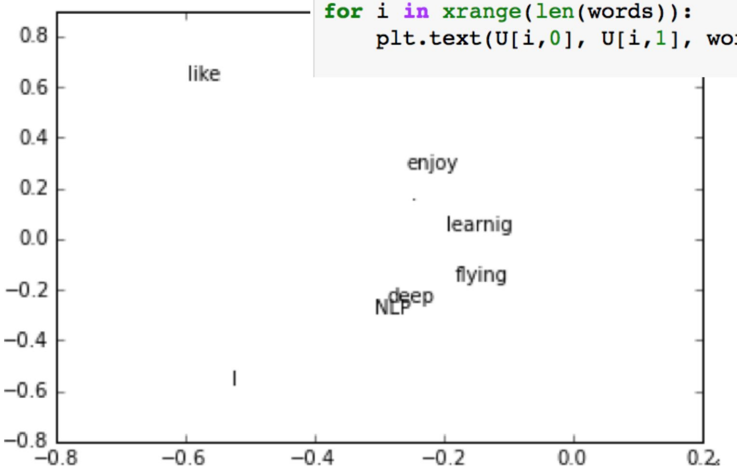
Corpus:

- I like deep learning. I like NLP. I enjoy flying.

```python
import numpy as np
la = np.linalg
words = ["I", "like", "enjoy",
         "deep","learnig","NLP","flying","."]
X = np.array([[0,2,1,0,0,0,0,0],
              [2,0,0,1,0,1,0,0],
              [1,0,0,0,0,0,1,0],
              [0,1,0,0,1,0,0,0],
              [0,0,0,1,0,0,0,1],
              [0,1,0,0,0,0,0,1],
              [0,0,1,0,0,0,0,1],
              [0,0,0,0,1,1,1,0]])

U, s, Vh = la.svd(X, full_matrices=False)
```

```python
for i in xrange(len(words)):
    plt.text(U[i,0], U[i,1], words[i])
```



- Printing first two columns of U corresponding to the 2 biggest singular values.

riminder https://github.com/riminder

Trouble with Learning systems

Adversarial example

98% Toaster

Bias example

95% Unqualified

riminder https://github.com/riminder

GENDER STEREOTYPING : WORD EMBEDDINGS

HE

homemaker

nurse

receptionist

librarian

SHE

maestro

skipper

protege

philosopher

SOFTBALL

pitcher

bookkeeper

registered nurse

waitress

FOOTBALL

footballer

businessman

maestro

cleric

GENDER STEREOTYPING : REPRESENTATION

Formulation:

- Let B be the gender subspace
- "Hard bias correction": For a gender neutral word w , set
  $w \leftarrow w - w_B / \ ||w - w_B||$
- "Soft Bias correction": For word matrix W, gender neutral words N find a linear transformation T

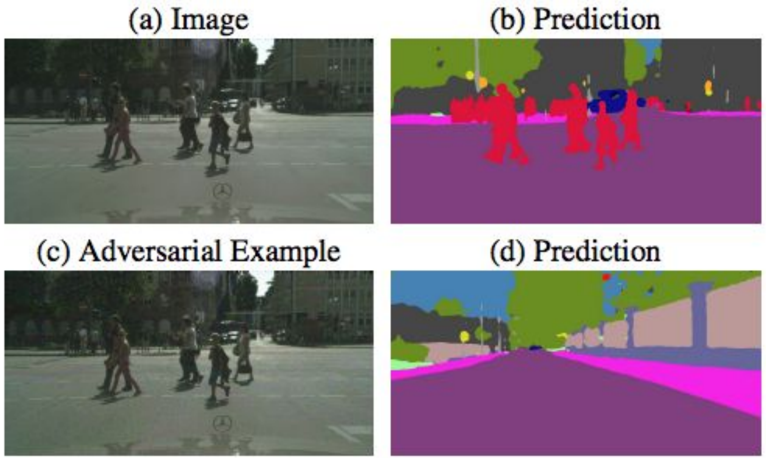
$$\min_T ||(TW)^T TW - W^T W||_F^2 + \lambda ||(TN)^T TB - ||_F^2$$

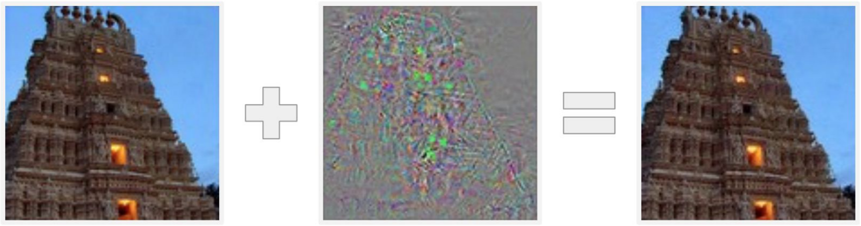
Results:

- Reduced gender stereotyping in the word embeddings
- Performance on downstream tasks still almost the same

Man Is To Computer Programmer as Woman is to Homemaker? Debiasing Word Embeddings. T Bolukbasi, K-W Chang, J Zou, V Saligrama, A Kalai. Arxiv 2016

Adversarial examples





Original image
Temple (97%)

Perturbations

Adversarial example
Ostrich (98%)

Generating Adversarial Examples

The Fast Gradient Sign Method

$$J(\tilde{\boldsymbol{x}}, \boldsymbol{\theta}) \approx J(\boldsymbol{x}, \boldsymbol{\theta}) + (\tilde{\boldsymbol{x}} - \boldsymbol{x})^\top \nabla_{\boldsymbol{x}} J(\boldsymbol{x}).$$

Maximize

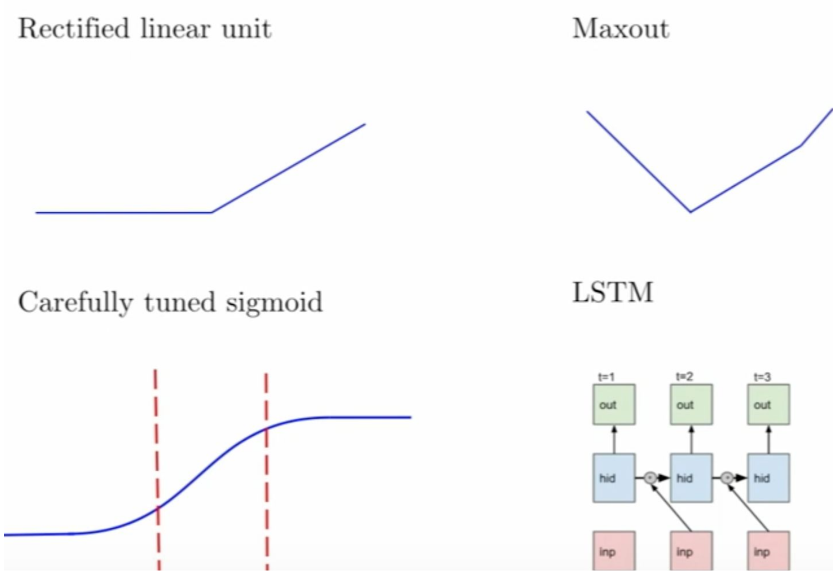
$$J(\boldsymbol{x}, \boldsymbol{\theta}) + (\tilde{\boldsymbol{x}} - \boldsymbol{x})^\top \nabla_{\boldsymbol{x}} J(\boldsymbol{x})$$

subject to

$$||\tilde{\boldsymbol{x}} - \boldsymbol{x}||_\infty \leq \epsilon$$

$$\Rightarrow \tilde{\boldsymbol{x}} = \boldsymbol{x} + \epsilon \text{sign}\left(\nabla_{\boldsymbol{x}} J(\boldsymbol{x})\right).$$

Modern deep nets are very piecewise linear



riminder https://github.com/riminder

POINT OF CONTACT

Mouhidine SEIV
CEO & Founder @Riminder
mouhidine.seiv@riminder.net