Deep Learning for NLP

An Introduction to Neural Word Embeddings*

*and some more fun stuff…

Roelof Pieters
PhD candidate KTH/CSC
CIO/CTO Feeda AB

Feeda

KTH, December 4, 2014

ROYAL INSTITUTE OF TECHNOLOGY

roelof@kth.se www.csc.kth.se/~roelof/ @graphific

1. DEEP LEARNING

2. NLP: WORD EMBEDDINGS

1. DEEP LEARNING

2. NLP: WORD EMBEDDINGS

A couple of headlines… [all November '14]

Baidu's Andrew Ng on Deep Learning and Innovation in Silicon Valley

Nervana Systems raises $3.3M to build hardware designed for deep learning

by Derrick Harris Aug. 21, 2014 - 5:48 AM PST

Artificially Intelligent Robot Scientists Could Be Next Project for Google's AI Firm

Deep learning might help you get an ultrasound at Walgreens

by

A Googler's Quest to Teach Machines How to Understand Emotions

Google, Spotify, & Pandora bet a computer could generate a better playlist than you can

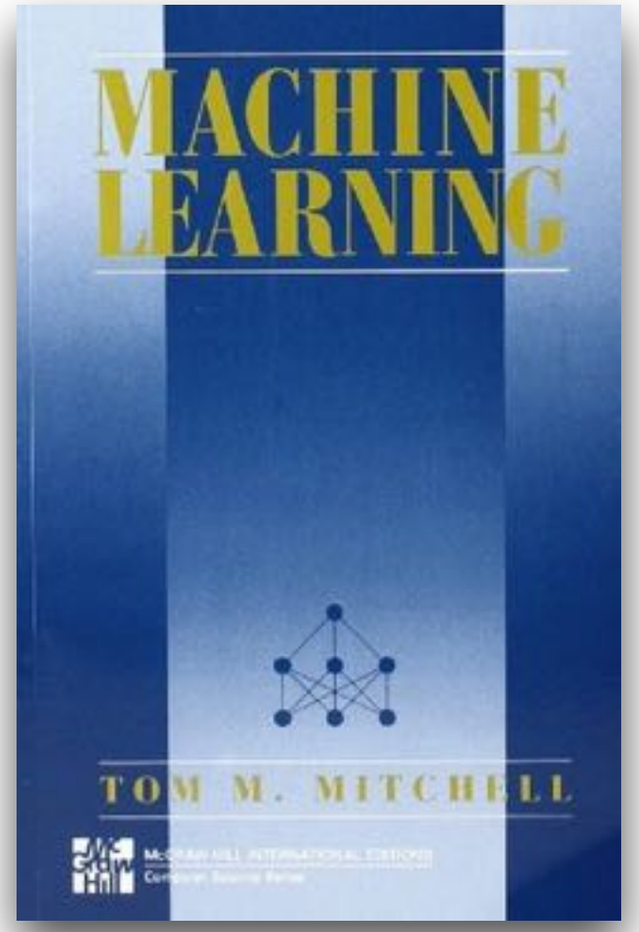
Butterfly Network Hopes to Bring Deep Learning AI to Medicine

Enlitic picks up $2M to help diagnose diseases with deep learning

Deep Learning = Machine Learning

Improving some task T based on experience E with respect to performance measure P.

— T. Mitchell 1997



Learning denotes changes in the system that are **adaptive** in the sense that they enable the system to do the same task (or tasks drawn from a population of similar tasks) more effectively the next time.

— H. Simon 1983

"Why Should Machines Learn?" in Mitchell 1997

Deep Learning: What?

Representation learning

Attempts to automatically learn good features or representations

Deep learning

Attempt to learn multiple levels of representation of increasing complexity/abstraction

ML: Traditional Approach

For each new problem/question::

1. Gather as much LABELED data as you can get

2. Throw some algorithms at it (mainly put in an SVM and keep it at that)

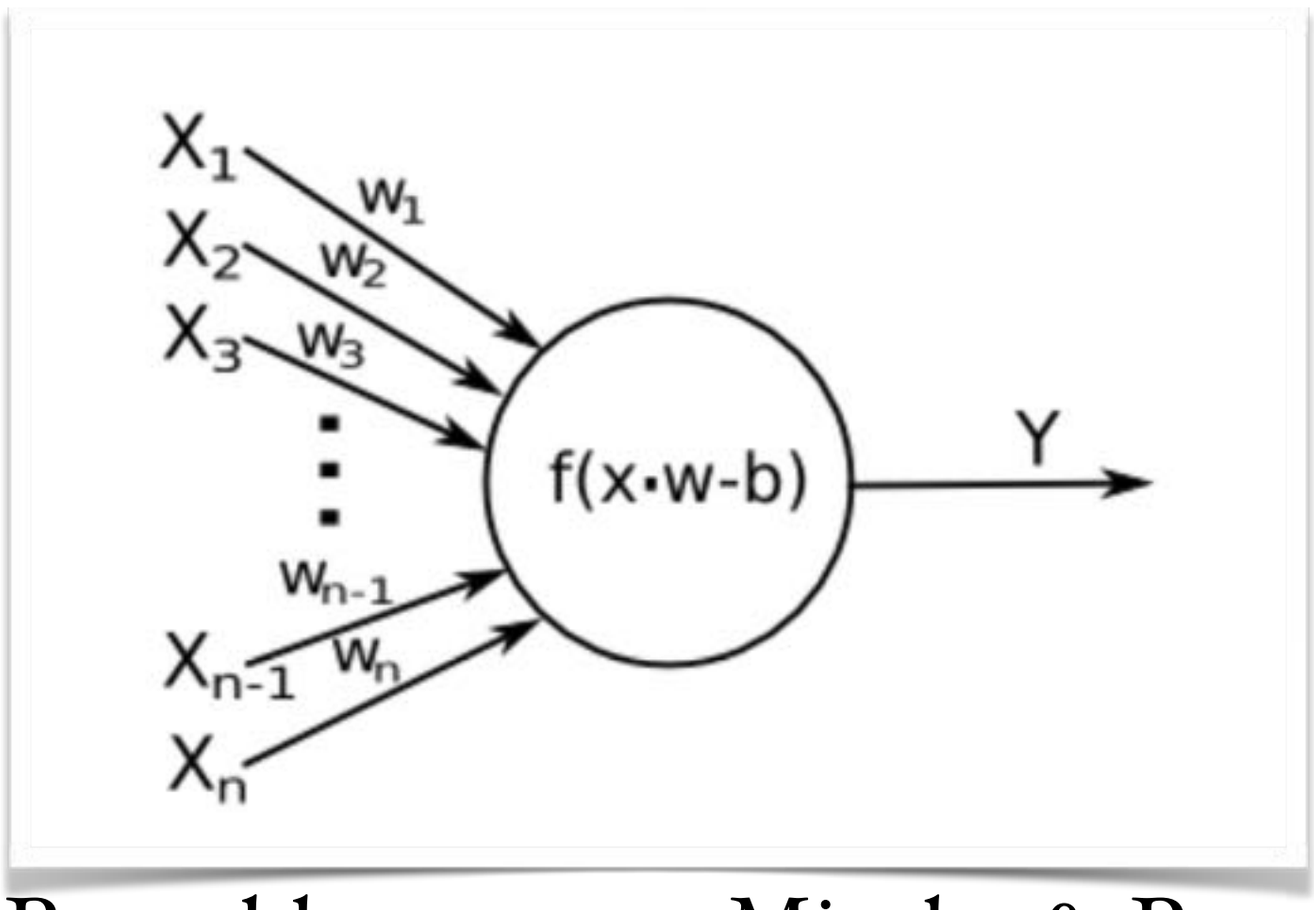
3. If you actually have tried more algos: Pick the best

4. Spend hours hand engineering some features / feature selection / dimensionality reduction (PCA, SVD, etc)

5. Repeat…

History

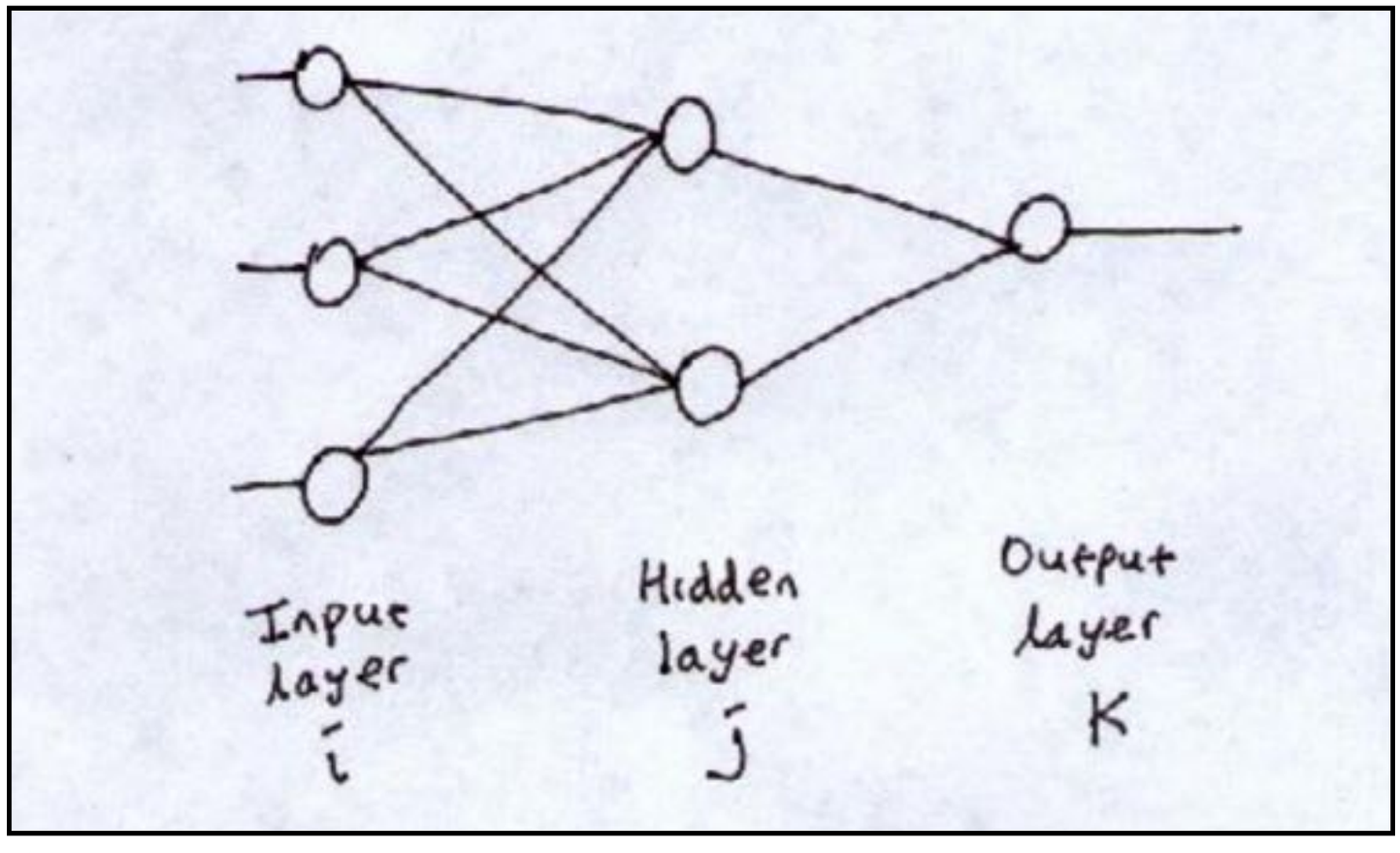
- **Perceptron ('57-69...)**
- Multi-Layered Perceptrons ('86)
- SVMs (popularized 00s)
- RBM ('92+)
- "2006"



Rosenblatt 1957 vs Minsky & Papert

History

- Perceptron (’57-69…)
- **Multi-Layered Perceptrons (’86)**
- SVMs (popularized 00s)
- RBM (‘92+)
- “2006”



(Rumelhart, Hinton & Williams, 1986)

Backprop Renaissance

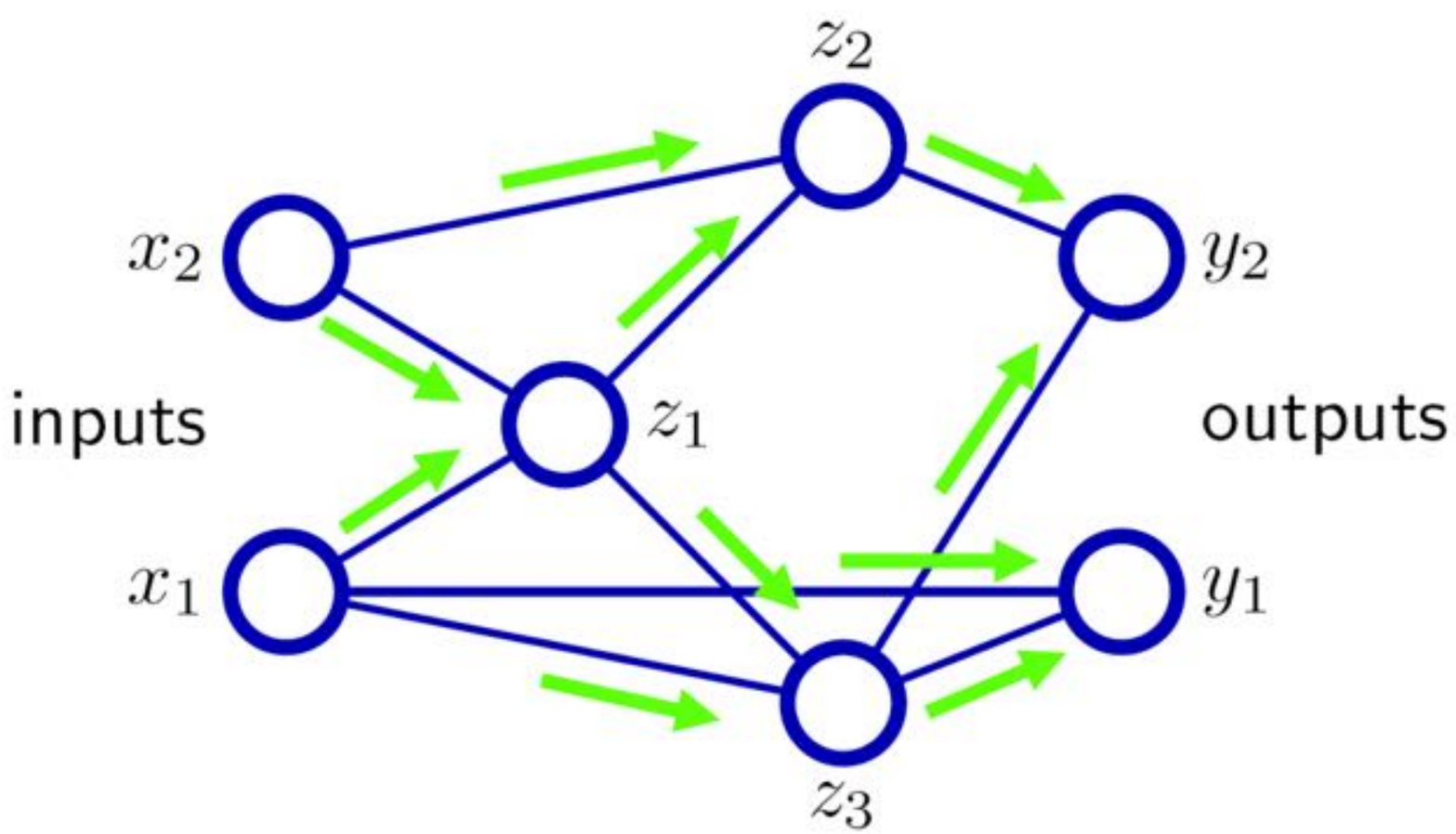
- Multi-Layered Perceptrons ('86)
  - Uses Backpropagation (Bryson & Ho 1969):

    back-propagates the error signal computed at the output layer to get derivatives for learning, in order to update the weight vectors until convergence is reached

Backprop Renaissance

Forward Propagation

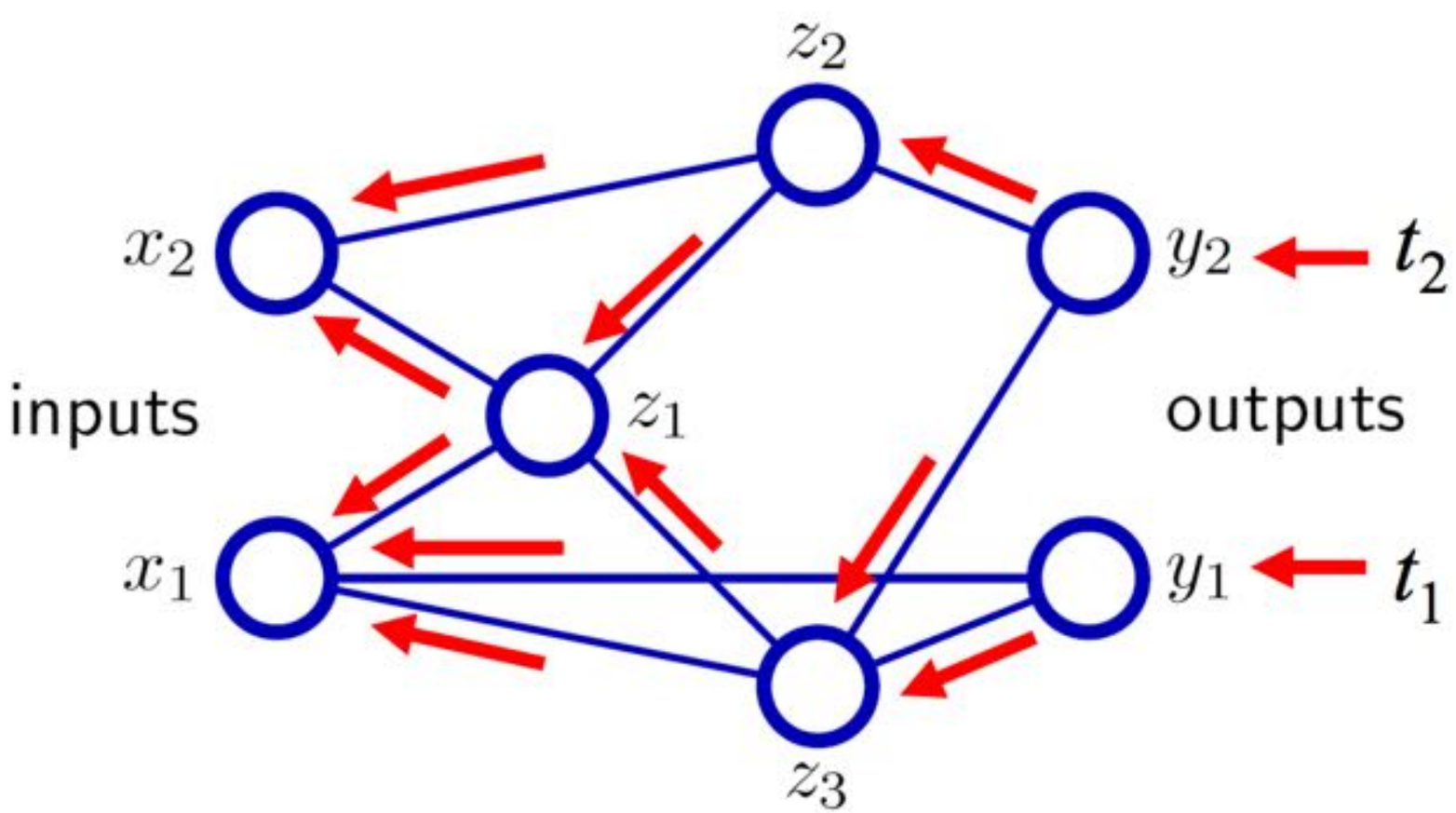
- Sum inputs, produce activation, feed-forward



Backprop Renaissance

Back Propagation (of error)

- Calculate total error at the top
- Calculate contributions to error at each step going backwards



Backpropagation

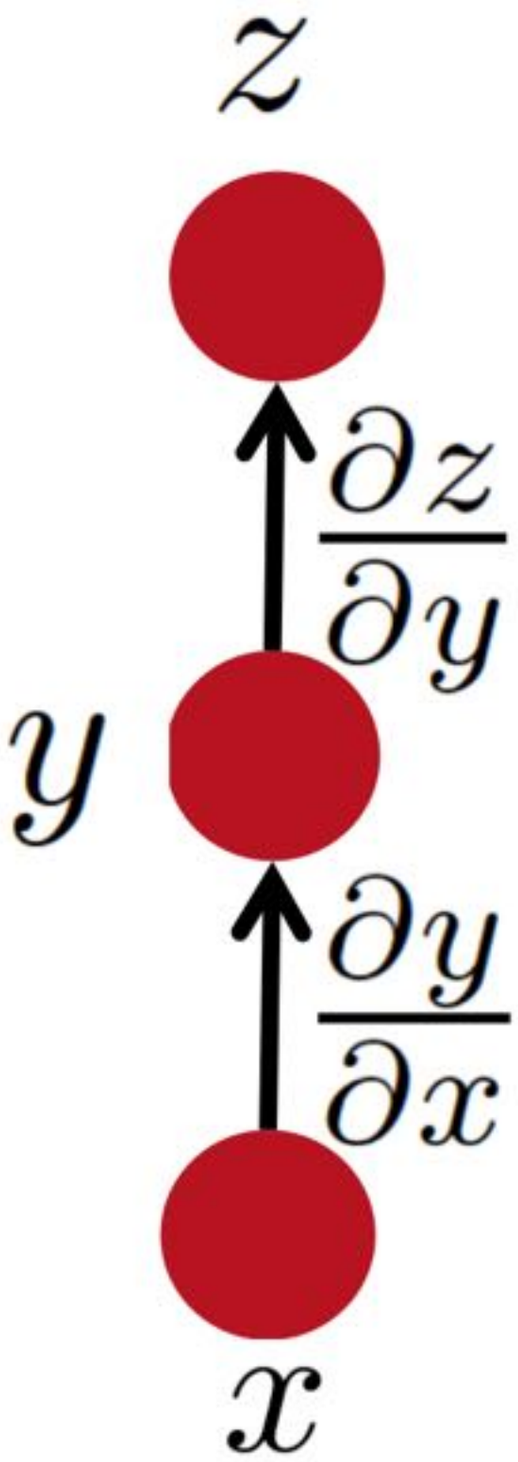
- Compute gradient of example-wise loss wrt parameters

- Simply applying the derivative chain rule wisely

$$z = f(y) \quad y = g(x) \quad \frac{\partial z}{\partial x} = \frac{\partial z}{\partial y}\frac{\partial y}{\partial x}$$

- If computing the loss (example, parameters) is O(n) computation, then so is computing the gradient

Simple Chain Rule



$$\Delta z = \frac{\partial z}{\partial y}\Delta y$$

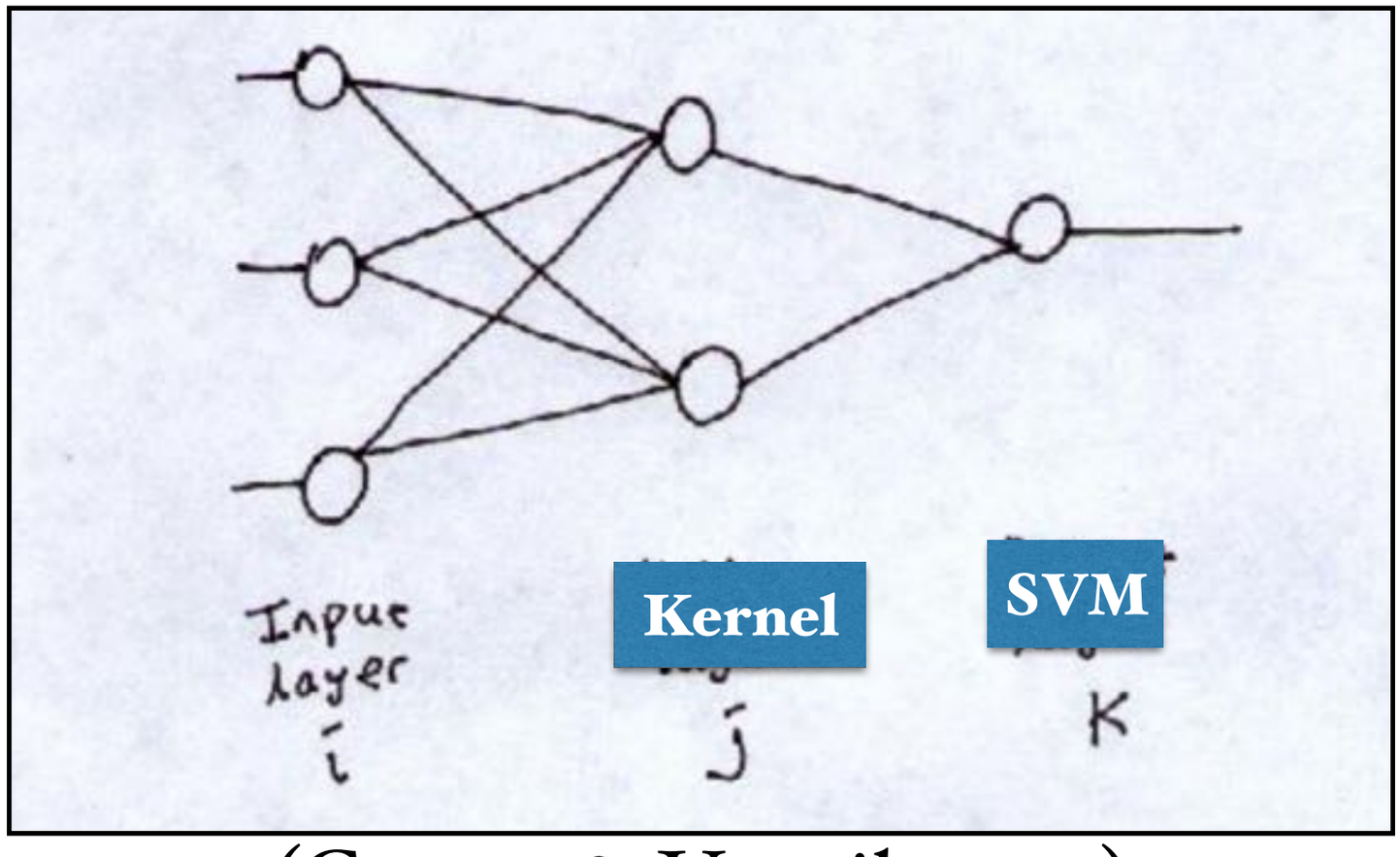
$$\Delta y = \frac{\partial y}{\partial x}\Delta x$$

$$\Delta z = \frac{\partial z}{\partial y}\frac{\partial y}{\partial x}\Delta x$$

$$\frac{\partial z}{\partial x} = \frac{\partial z}{\partial y}\frac{\partial y}{\partial x}$$

History

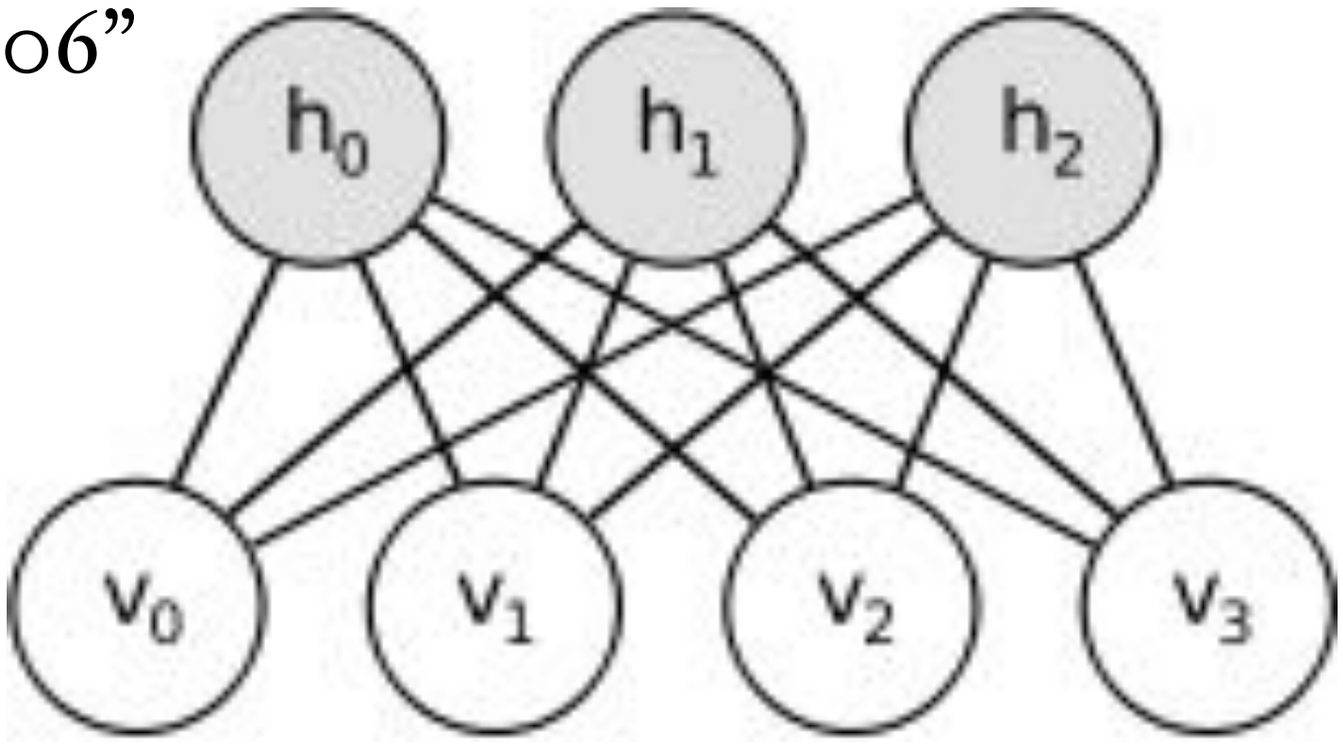
- Perceptron (’57-69…)
- Multi-Layered Perceptrons (’86)
- **SVMs (popularized oos)**
- RBM (‘92+)
- “2006”



(Cortes & Vapnik 1995)

History

- Perceptron ('57-69...)
- Multi-Layered Perceptrons ('86)
- SVMs (popularized 00s)
- **RBM ('92+)**
- "2006"



- **Form of log-linear Markov Random Field (MRF)**
- **Bipartite graph, with no intra-layer connections**

RBM: Structure

- **Energy Function**

$$E(v,h) = -\sum_i a_i v_i - \sum_j b_j h_j - \sum_i \sum_j v_i w_{i,j} h_j$$

RBM: Training

- **Training Function:**

$$\arg\max_W \prod_{v \in V} P(v) \qquad \arg\max_W \mathbb{E}\left[\sum_{v \in V} \log P(v)\right]$$

- often by contrastive divergence (CD) (Hinton 1999; Hinton 2000)
- Gibbs sampling
- Gradient Descent
- Goal: compute weight updates

more info: Geoffrey Hinton (2010). A Practical Guide to Training Restricted Boltzmann Machines

History

- Perceptron ('57-69...)
- Multi-Layered Perceptrons ('86)
- SVMs (popularized 00s)
- RBM ('92+)
- **"2006"**

1. **More labeled data ("Big Data")**
2. **GPU's**
3. **"layer-wise unsupervised feature learning"**

Stacking Single Layer Learners

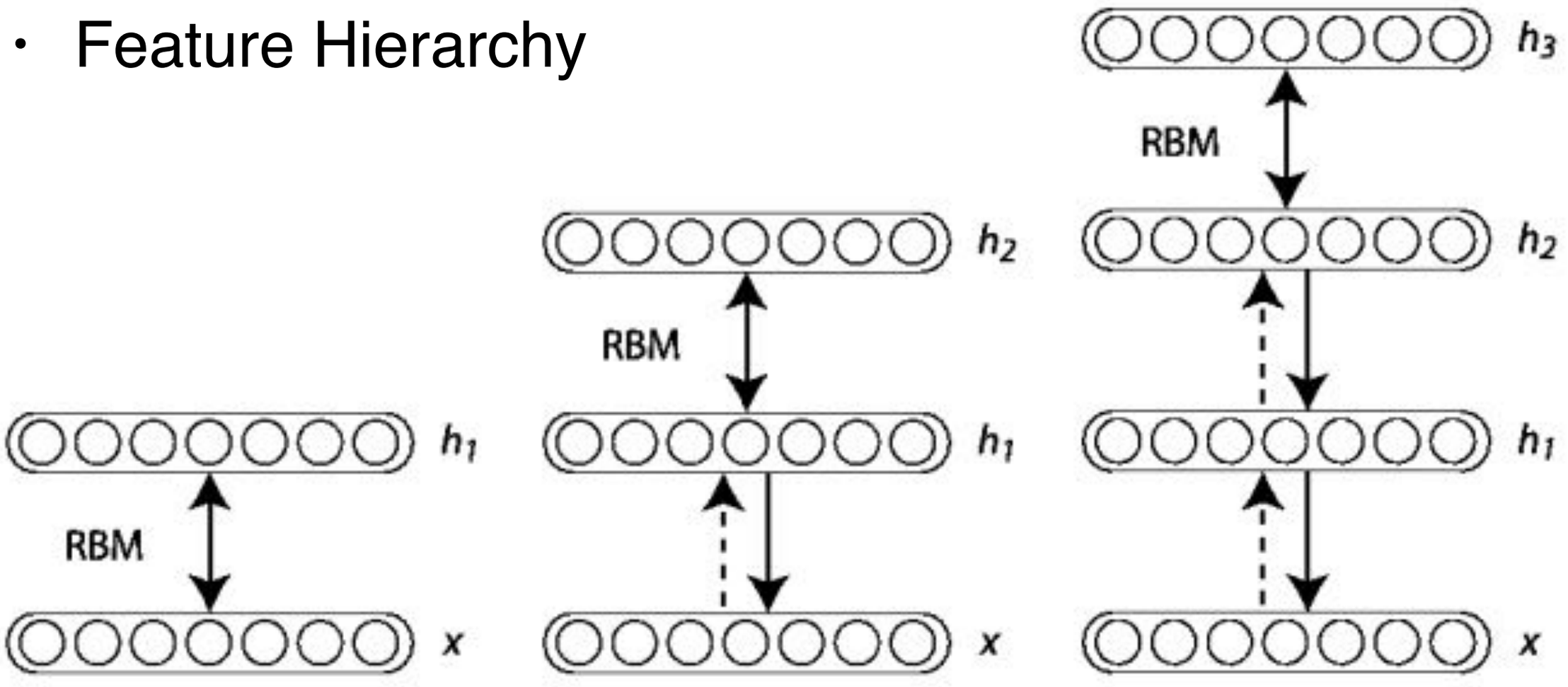
One of the big ideas from 2006: layer-wise unsupervised feature learning

- Stacking Restricted Boltzmann Machines (RBM) -> Deep Belief Network (DBN)

- Stacking regularized auto-encoders -> deep neural nets

Deep Belief Network (DBN)

- Stacked RBM
- Introduced by Hinton et al. (2006)
- 1st RBM hidden layer == 2th RBM input layer
- Feature Hierarchy



Stacked Autoencoders

Université de Montréal

Bengio

Hinton

Google

Restricted Boltzmann Machine

LeCun

facebook

Sparse Representations

SHALLOW

Recurrent Neural Net

Convolutional Neural Net

Neural Net

Boosting

Perceptron

SVM

Deep (sparse/denoising) Autoencoder

ΣΠ

Deep Belief Net

BayesNP

DEEP

Autoencoder Neural Net

Sparse Coding

GMM

Restricted BM

SHALLOW

Recurrent Neural Net

Convolutional Neural Net

Neural Net

Boosting

Perceptron

SVM

SUPERVISED

UNSUPERVISED

Deep (sparse/denoising) Autoencoder

ΣΠ

Deep Belief Net

BayesNP

DEEP

Neural Net

Sparse Coding

GMM

Restricted BM

SUPERVISED

SHALLOW

Recurrent Neural Net

Convolutional Neural Net

Neural Net

Boosting

Perceptron

SVM

UNSUPERVISED

DEEP

Deep (sparse/denoising) Autoencoder

ΣΠ

Deep Belief Net

BayesNP

Neural Net

Sparse Coding

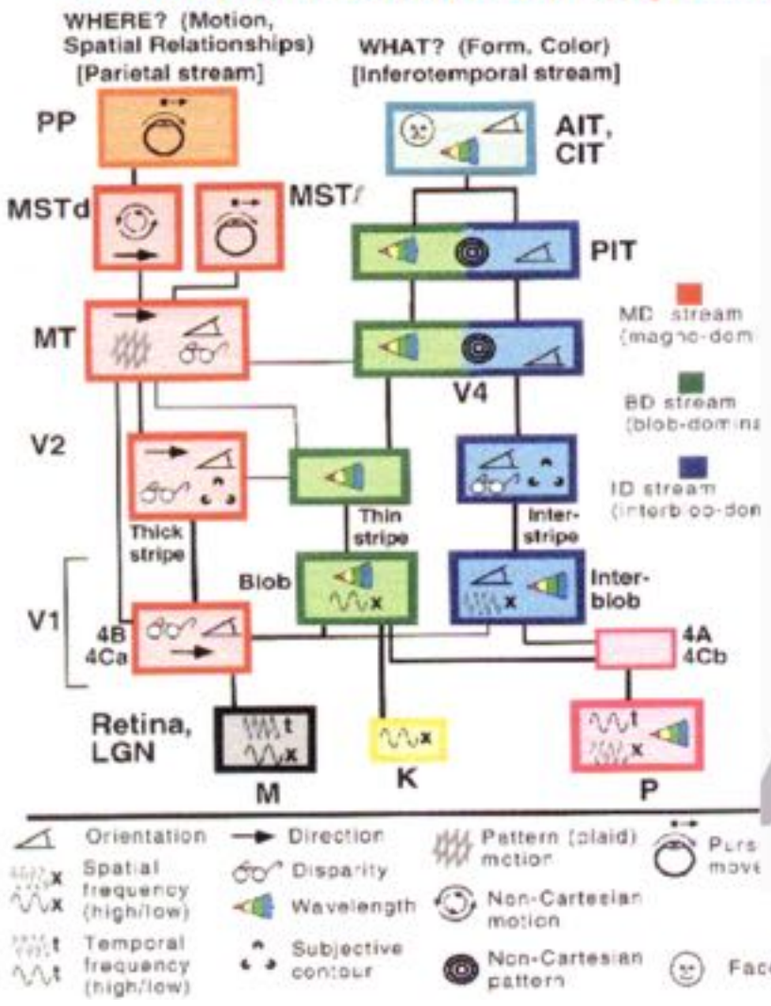
GMM

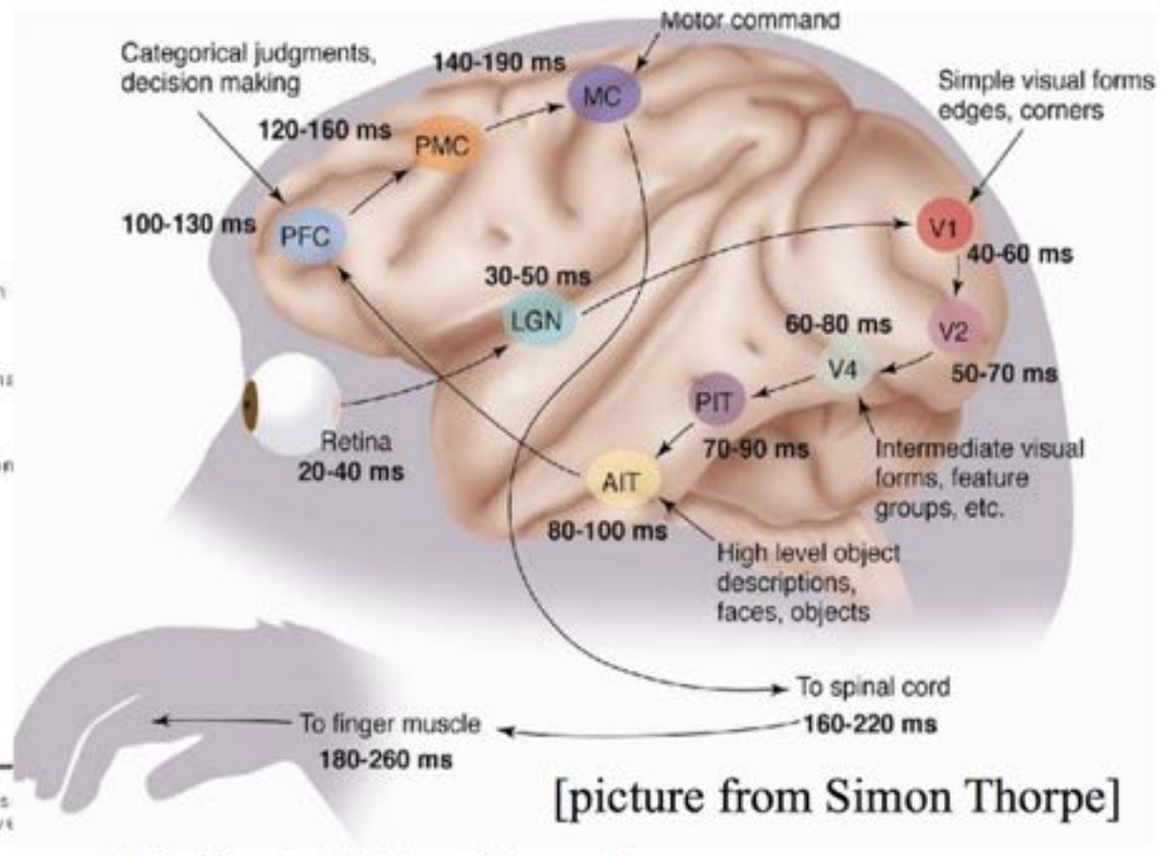
Restricted BM

Biological Justification

Deep Learning = Brain "inspired"

Audio/Visual Cortex has multiple stages == Hierarchical





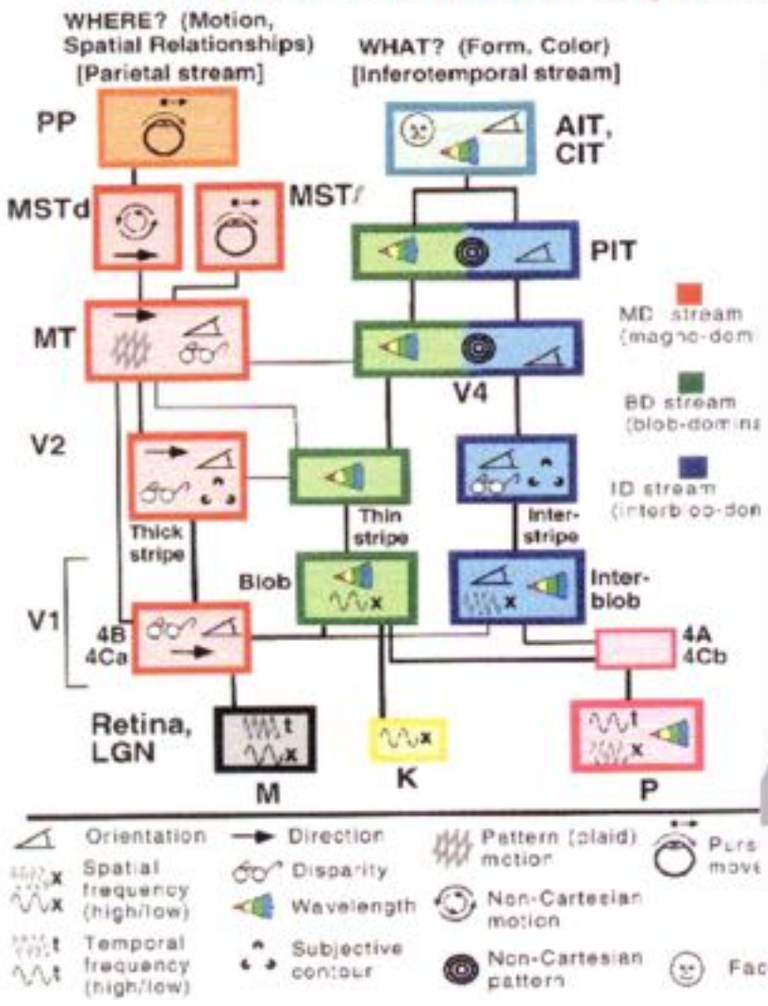
[picture from Simon Thorpe]

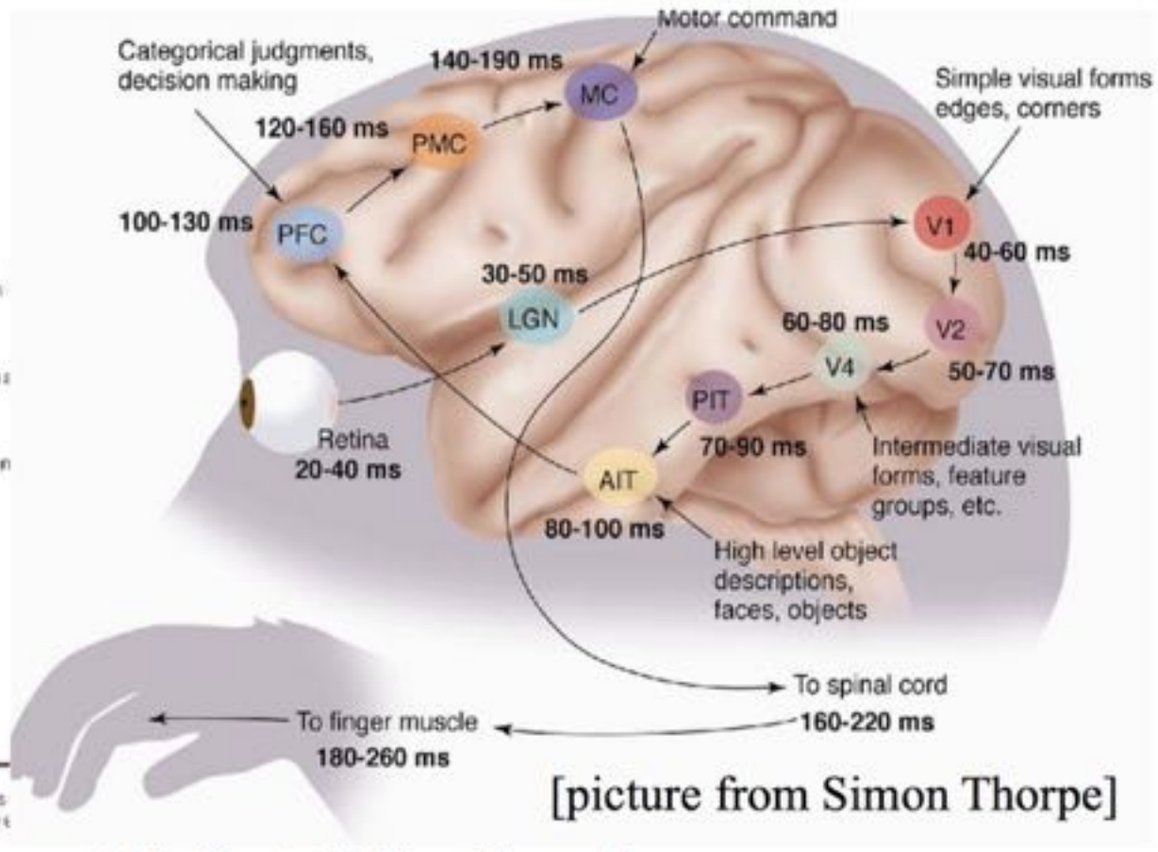
[Gallant & Van Essen]

Biological Justification

Deep Learning = Brain "inspired"

Audio/Visual Cortex has multiple stages == Hierarchical





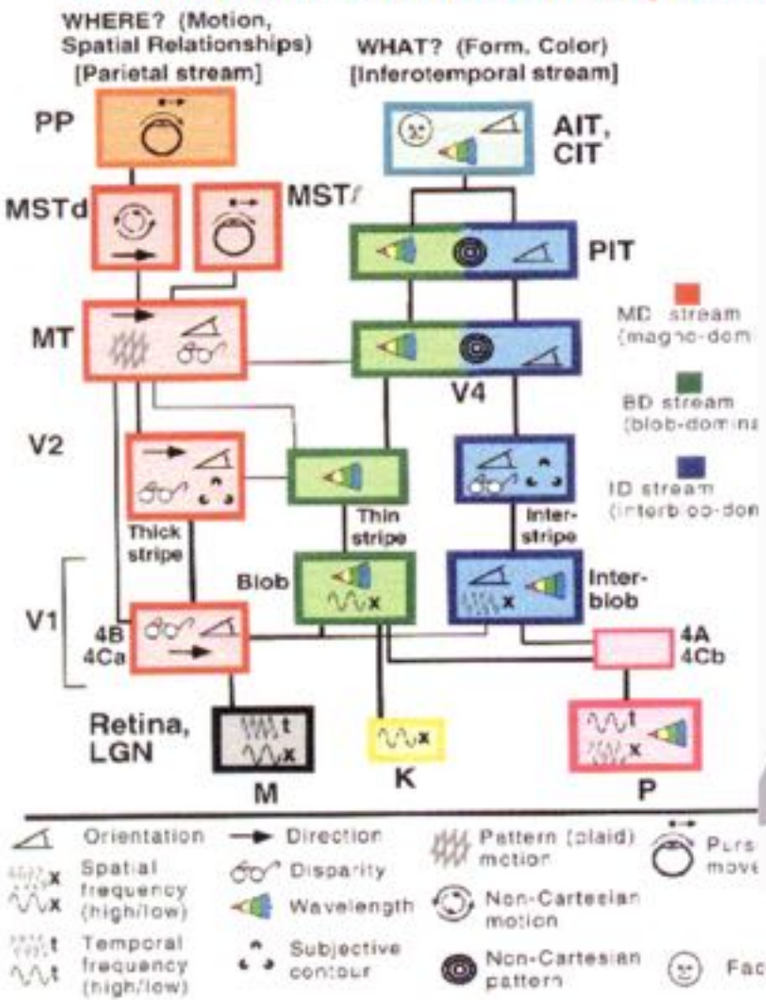
[Gallant & Van Essen]

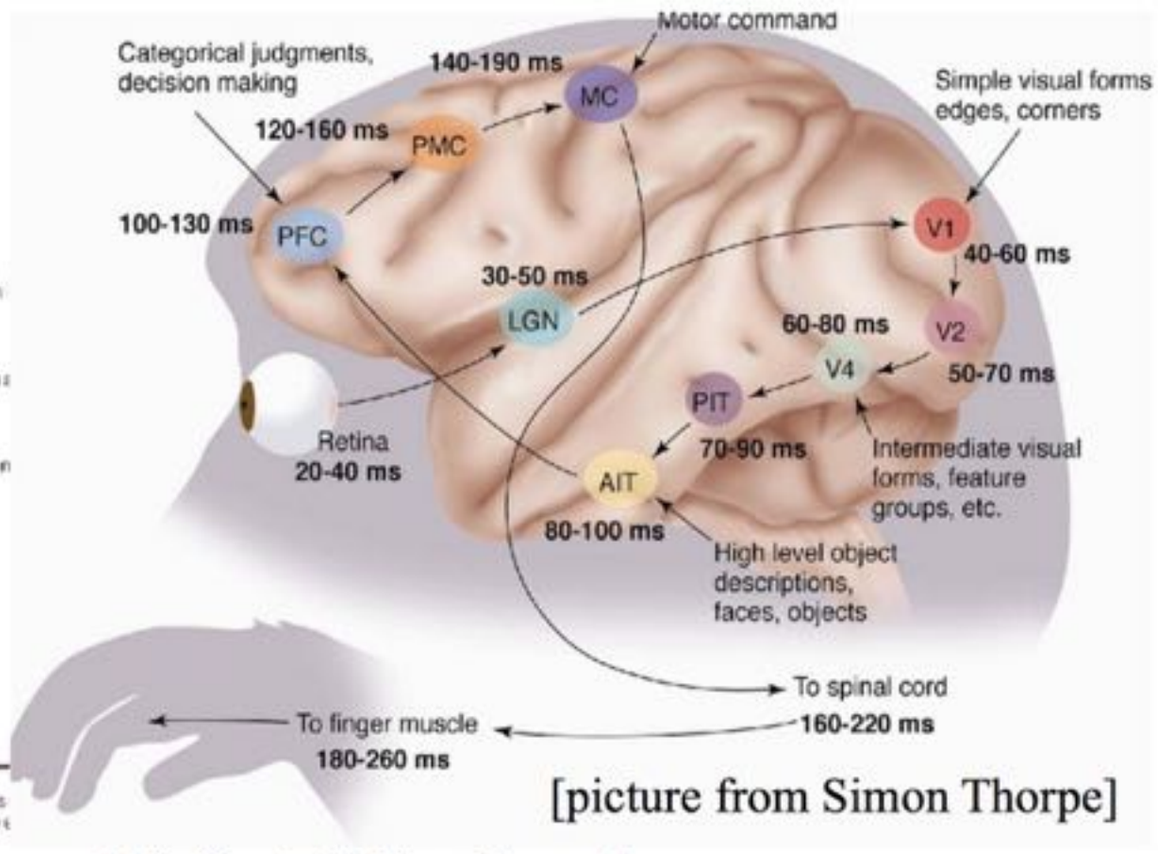
"Brainiacs" **vs** "Pragmatists"

Biological Justification

Deep Learning = Brain "inspired"

Audio/Visual Cortex has multiple stages == Hierarchical





[Gallant & Van Essen]

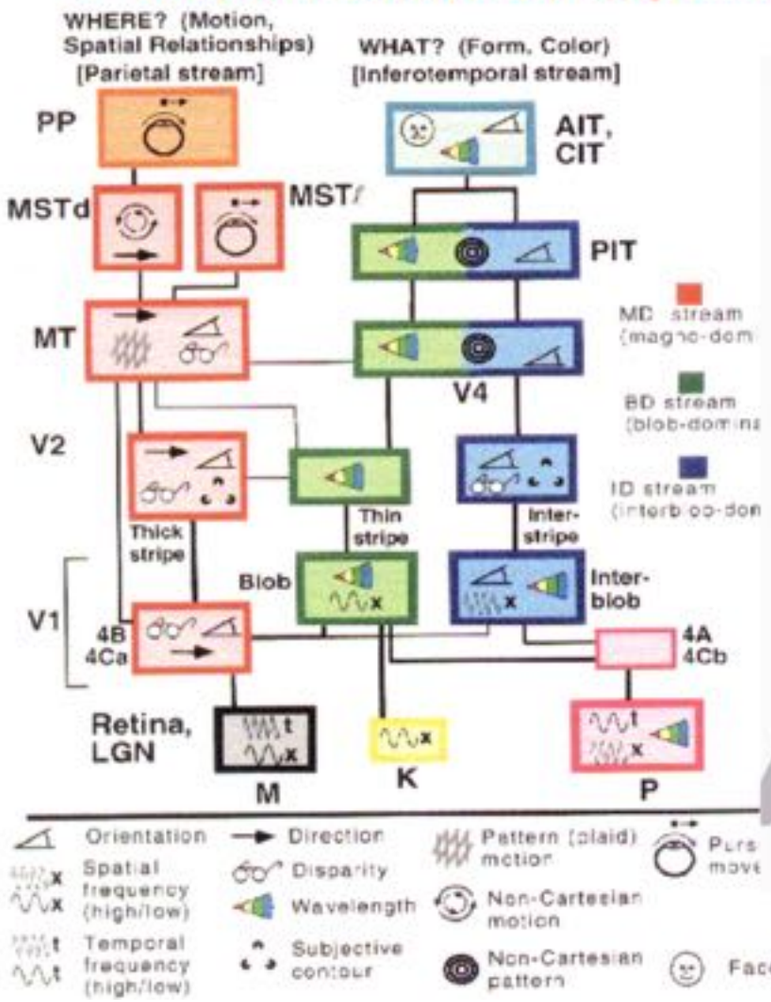
"Brainiacs" **vs** "Pragmatists"

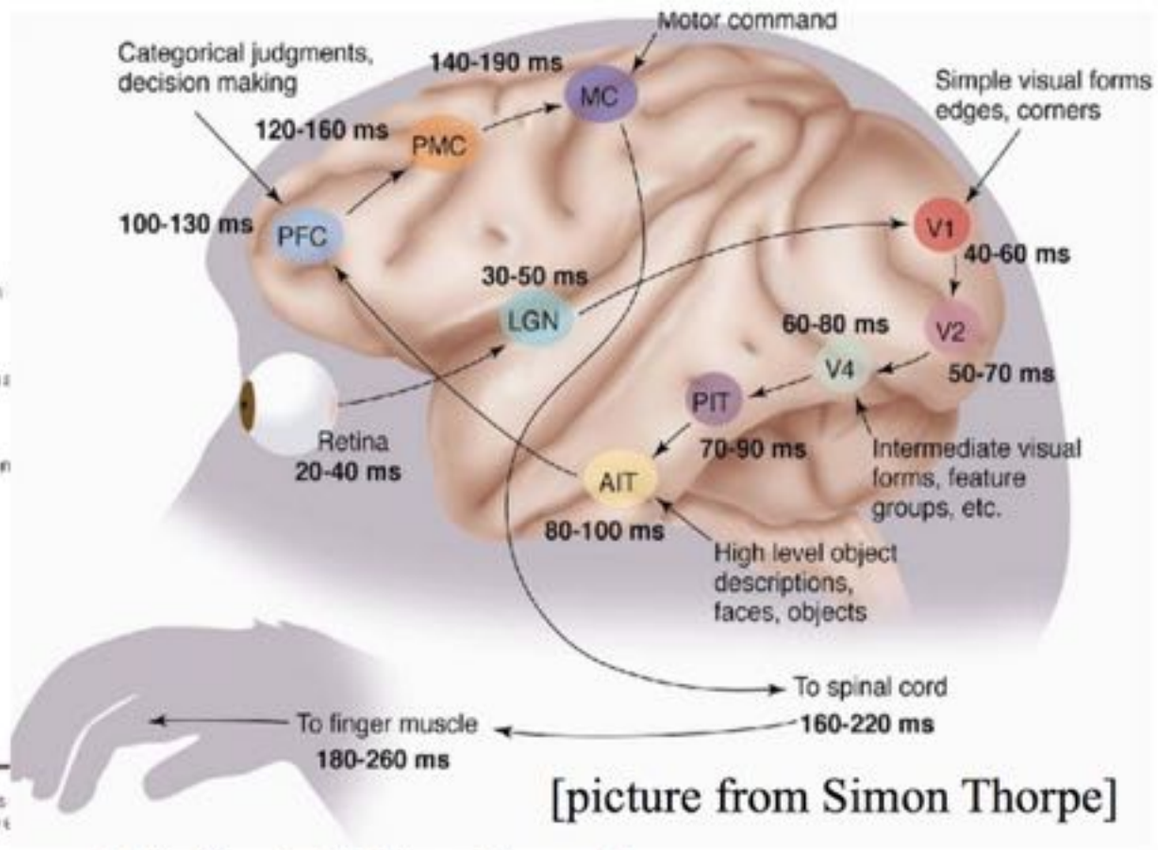
- Computational Biology
- CVAP

Biological Justification

Deep Learning = Brain "inspired"

Audio/Visual Cortex has multiple stages == Hierarchical





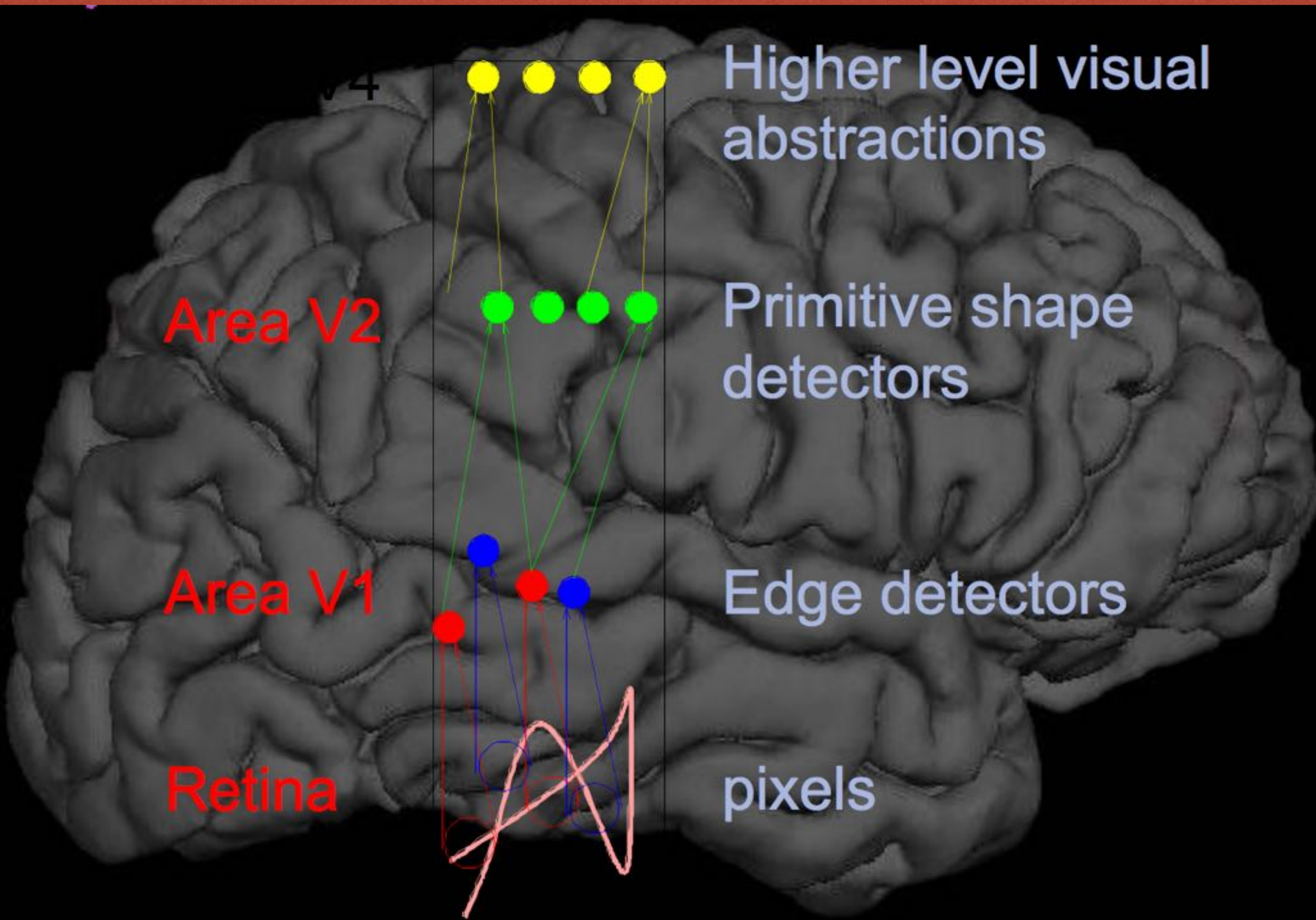
[Gallant & Van Essen]

"Brainiacs" **vs** "Pragmatists"

- Computational Biology
- Jorge Dávila-Chacón

- CVAP
- "that guy"

Different Levels of Abstraction



Different Levels of Abstraction

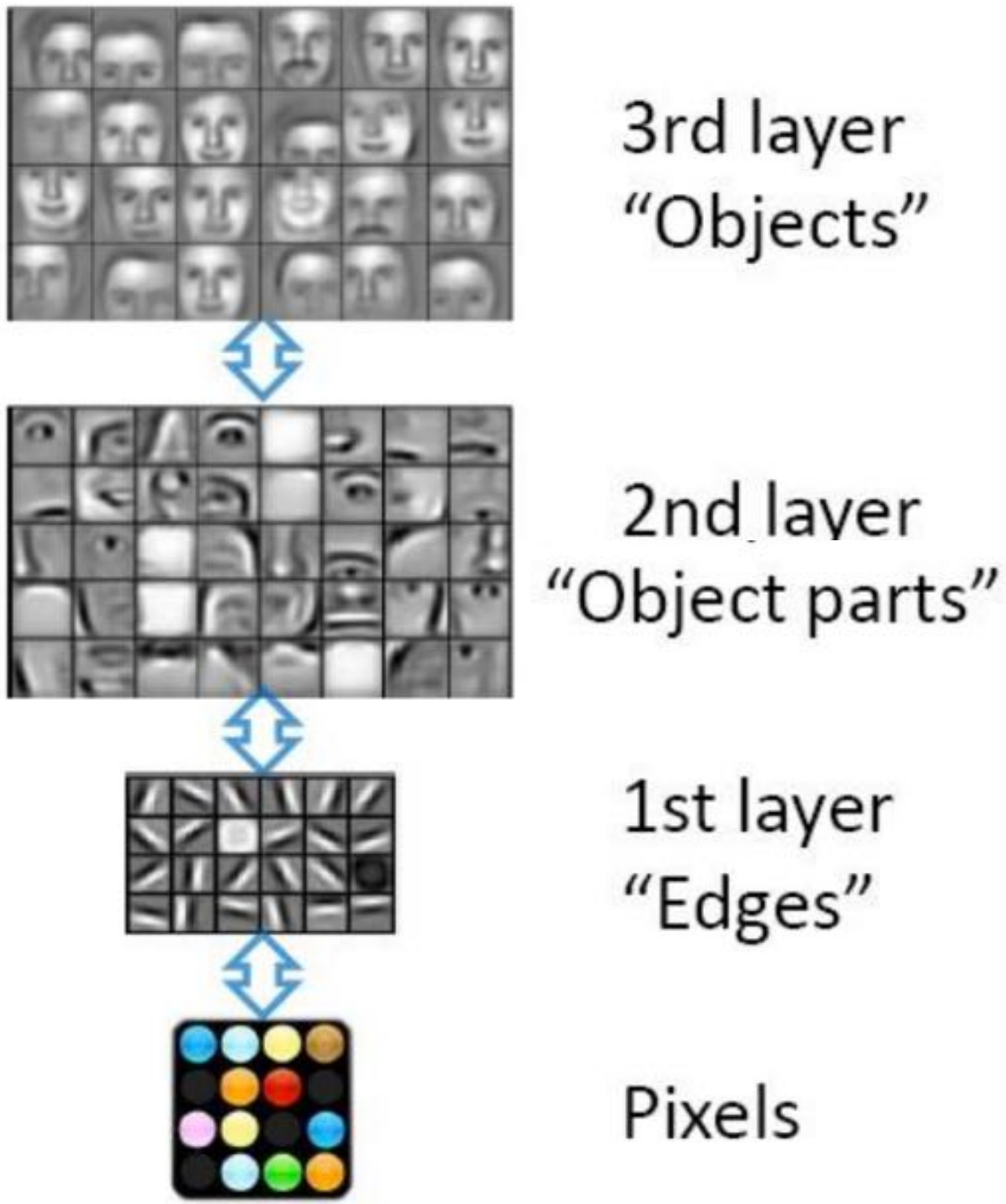
Hierarchical Learning

- Natural progression from low level to high level structure as seen in natural complexity

- Easier to monitor what is being learnt and to guide the machine to better subspaces

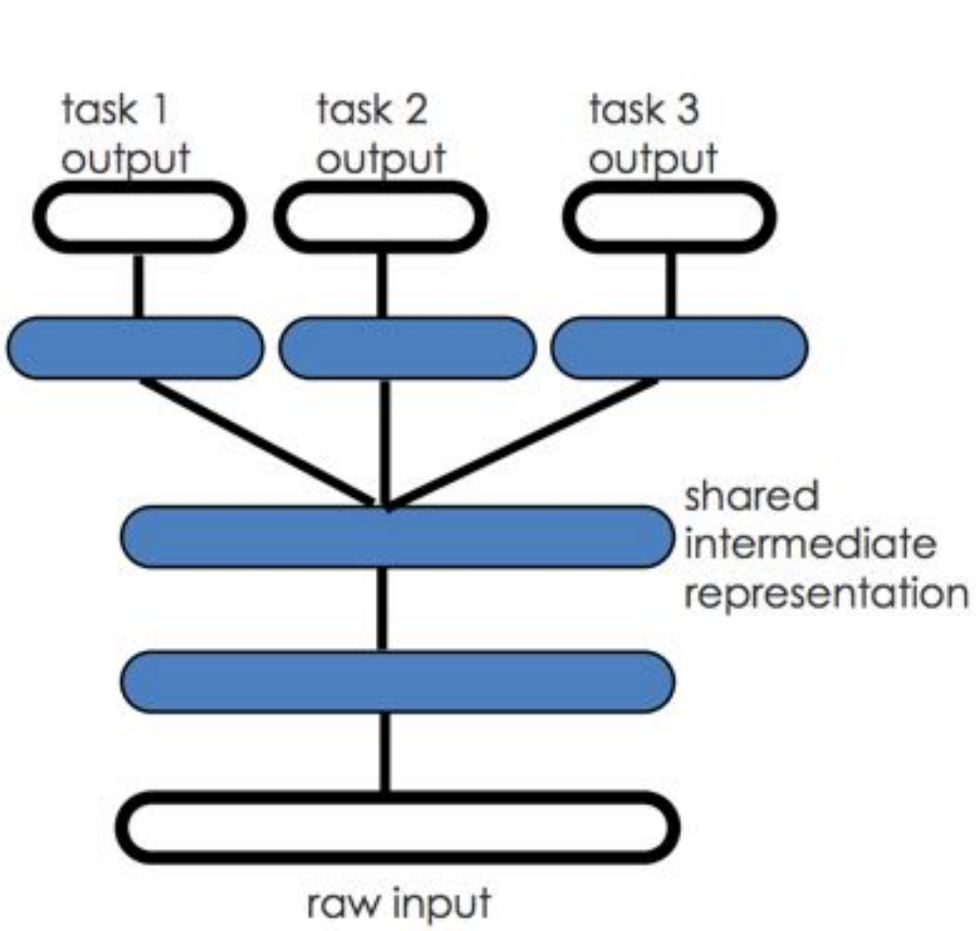
- A good lower level representation can be used for many distinct tasks

Feature Representation

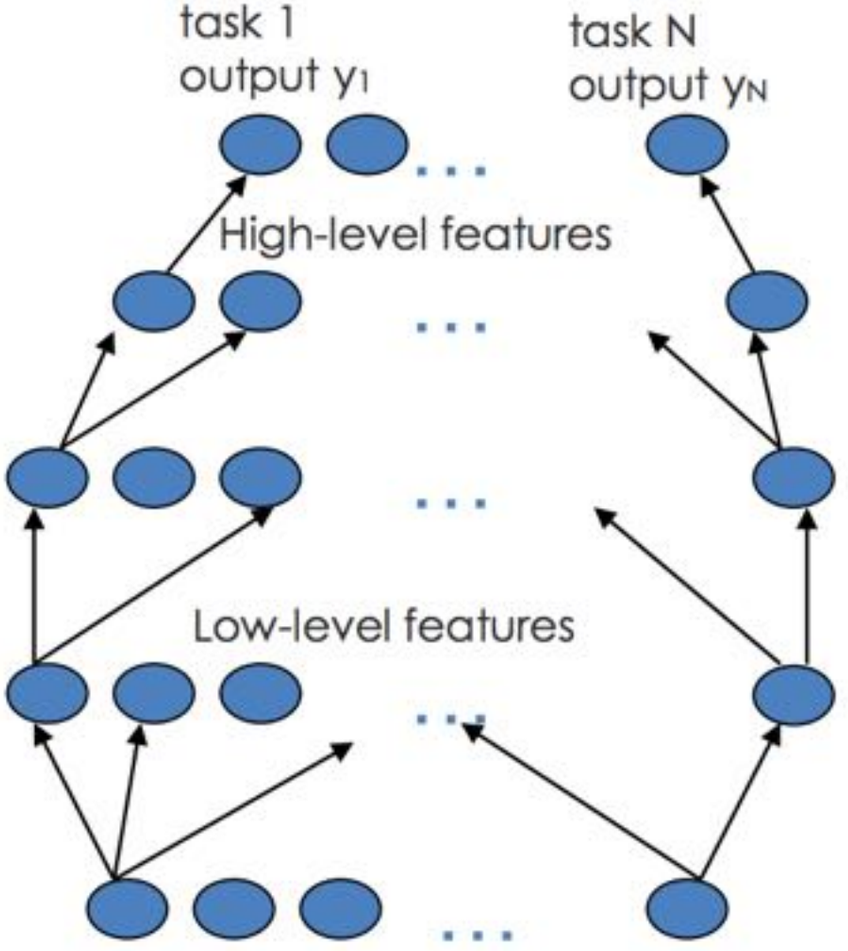


Generalizable Learning

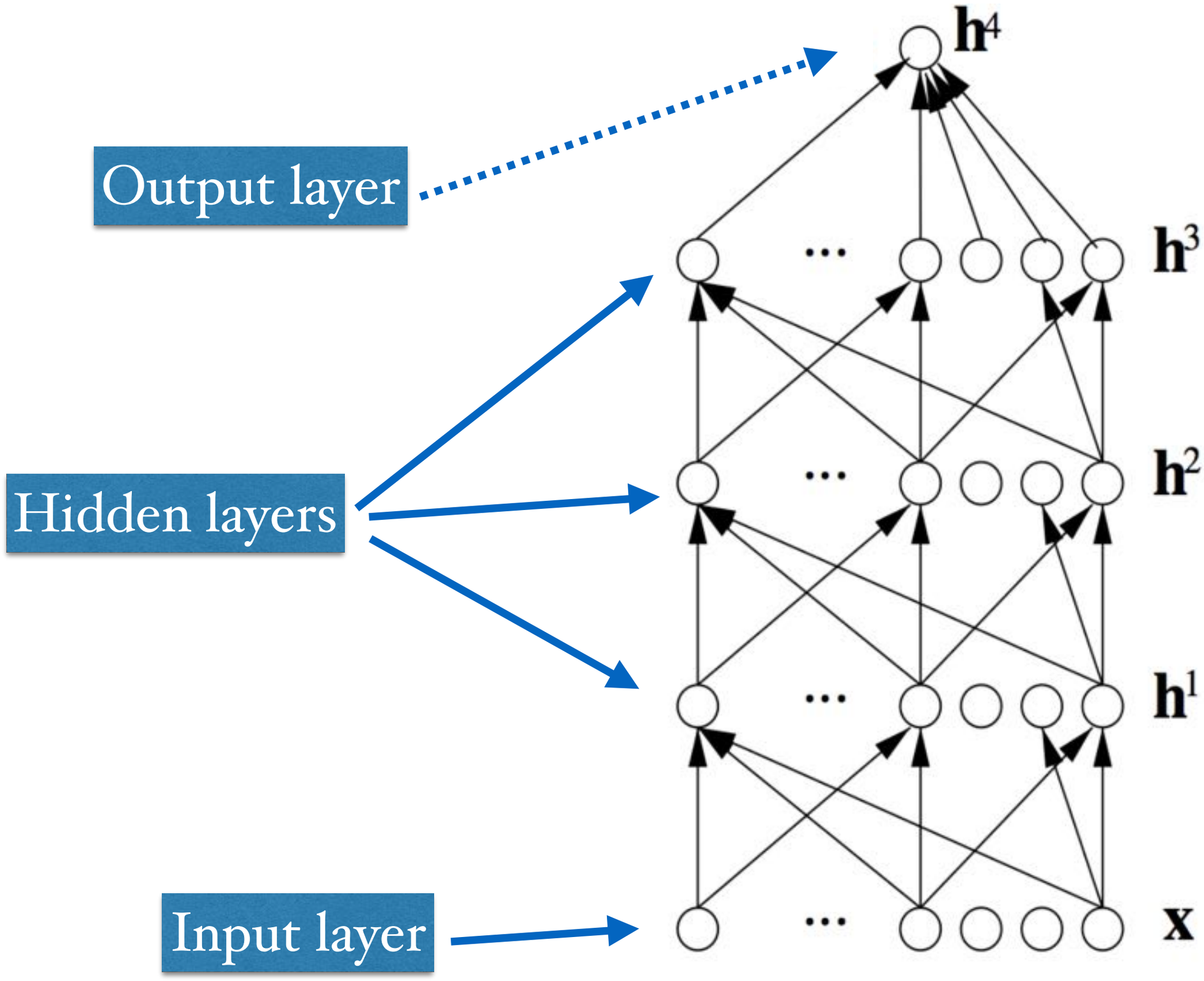
- Shared Low Level Representations
  - Multi-Task Learning
  - Unsupervised Training



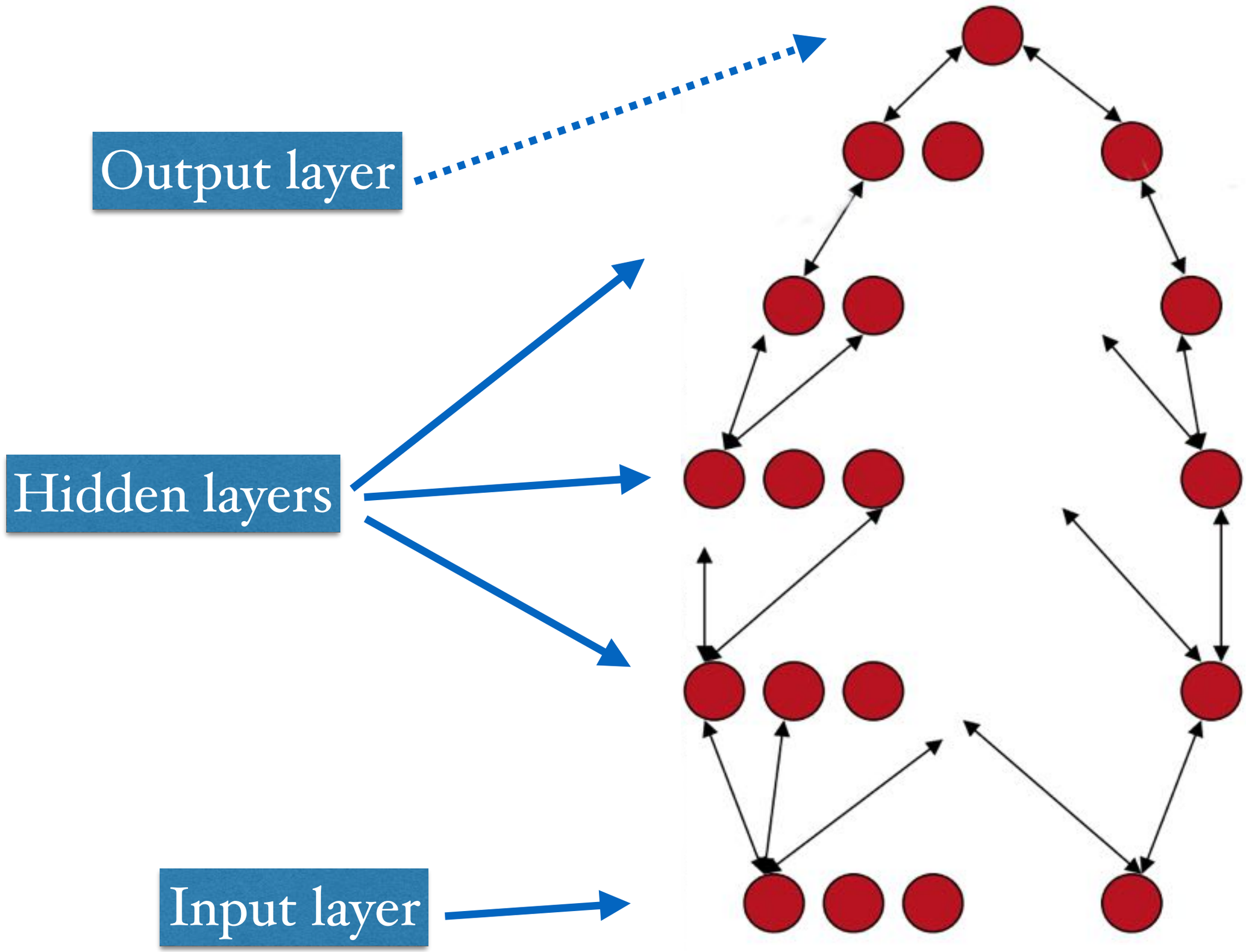
- Partial Feature Sharing
  - Mixed Mode Learning
  - Composition of Functions



Classic Deep Architecture

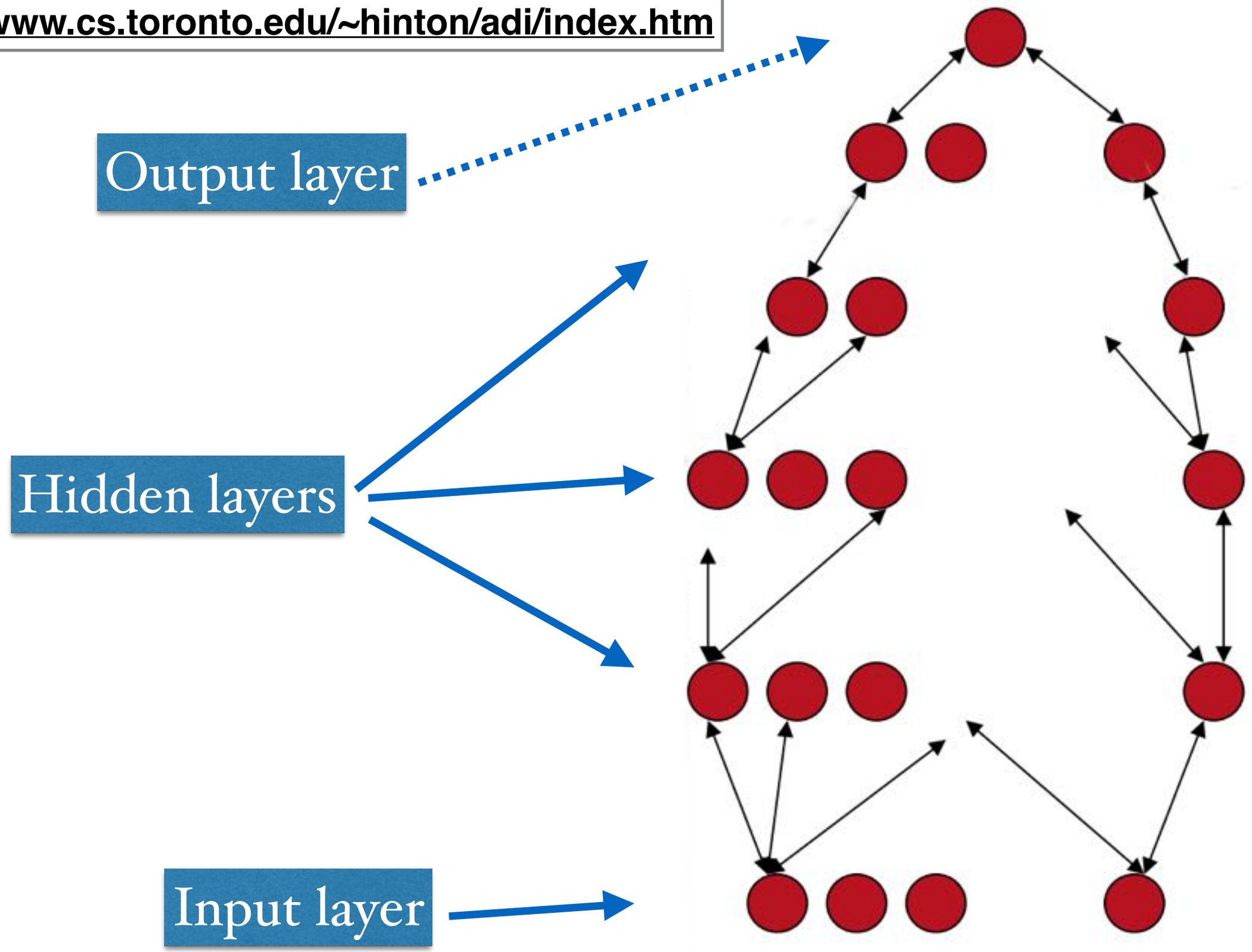


Modern Deep Architecture

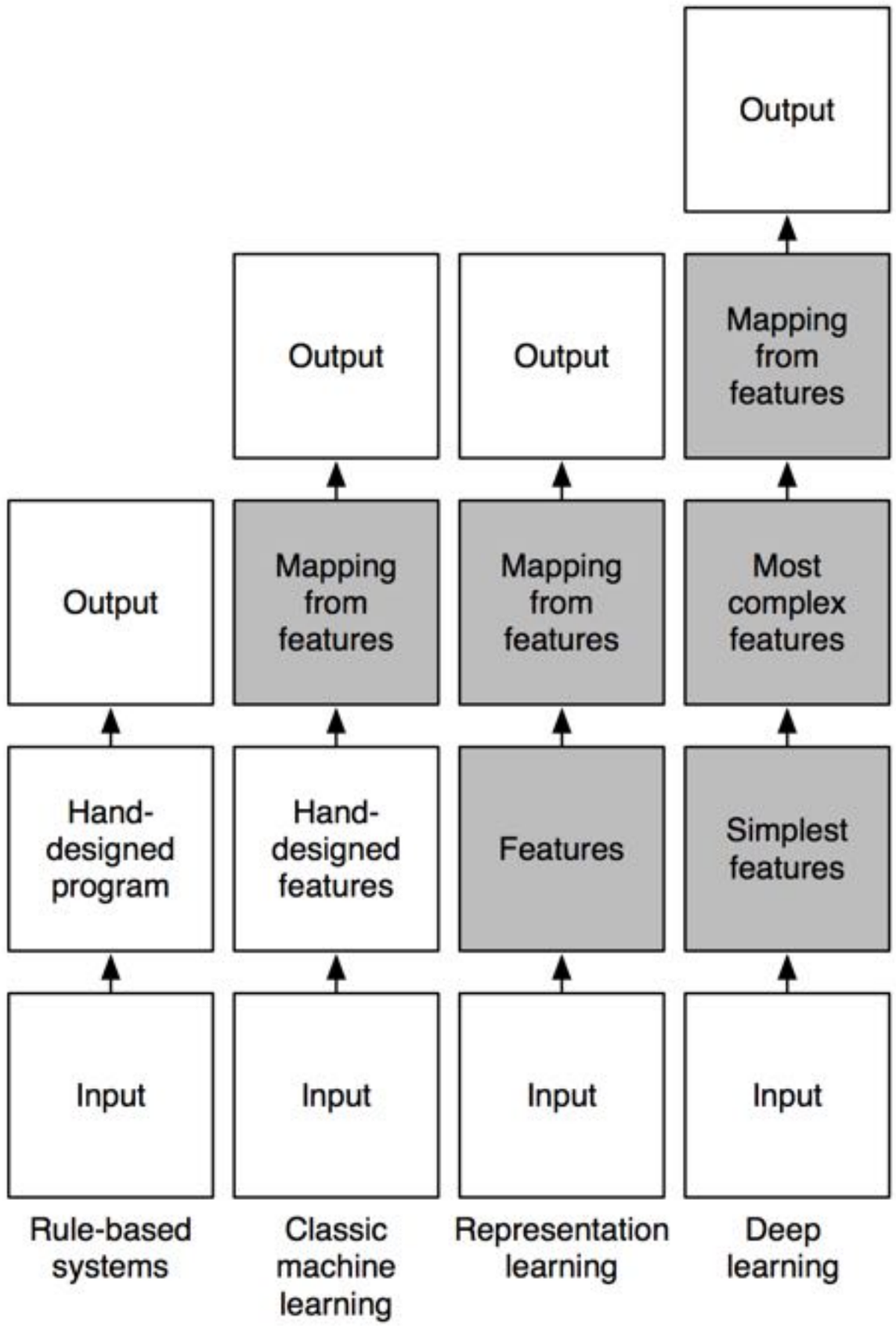


Modern Deep Architecture

movie time:
http://www.cs.toronto.edu/~hinton/adi/index.htm



Deep Learning



Why go Deep ?

Hierarchies

Distributed

Efficient

Sharing

Generalization

Unsupervised*

Major PWNAGE!

Black Box

Training Time

Much Data

No More Handcrafted Features !

Deep Learning: Why?

"I've worked all my life in Machine Learning, and I've never seen one algorithm knock over benchmarks like Deep Learning"

— Andrew Ng

Deep Learning: Why?

Beat state of the art in many areas:

- Language Modeling (2012, Mikolov et al)
- Image Recognition (Krizhevsky won 2012 ImageNet competition)
- Sentiment Classification (2011, Socher et al)
- Speech Recognition (2010, Dahl et al)
- MNIST hand-written digit recognition (Ciresan et al, 2010)

Deep Learning: Why for NLP ?

One Model rules them all ?

DL approaches have been successfully applied to:

- Automatic summarization
- Coreference resolution
- Discourse analysis
- Machine translation
- Morphological segmentation
- Named entity recognition (NER)
- Natural language generation
- Word sense disambiguation
- Relationship extraction
- Speech processing
- Part-of-speech tagging
- sentence boundary disambiguation
- Sentiment analysis
- Optical character recognition (OCR)
- Question answering
- Parsing
- Word segmentation
- Natural language understanding
- Information retrieval (IR)
- Speech recognition
- Topic segmentation and recognition
- Speech segmentation
- Information extraction (IE)

1. DEEP LEARNING

2. NLP: WORD EMBEDDINGS

Word Representation

- NLP treats words mainly (rule-based/statistical approaches at least) as atomic symbols:

  Love Candy Store

- or in vector space:

  [o o o o o **1** o o o o o o o o o o o o ...]

- also known as “one hot” representation.

- Its problem ?

Word Representation

- NLP treats words mainly (rule-based/statistical approaches at least) as atomic symbols:

Love Candy Store

- or in vector space:

[o o o o o **1** o o o o o o o o o o o o ...]

- also known as “one hot” representation.

- Its problem ?

Candy [o o o o o **1** o o o o o o o o o o o o ...] AND

Store [o o o o o o o o o o o o o o o o **1** o ...] = **0** !

Distributional representations

"You shall know a word by the company it keeps"
(J. R. Firth 1957)

One of the most successful ideas of modern statistical NLP!

government debt problems turning into **banking** crises as has happened in

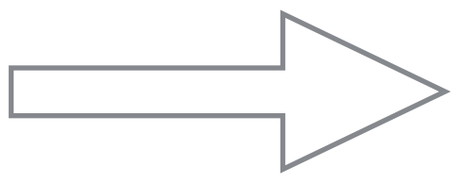
saying that Europe needs unified **banking** regulation to replace the hodgepodge

these words represent banking

- Hard (class based) clustering models:
- Soft clustering models

Language Modeling

- Word Embeddings (Bengio et al, 2001; Bengio et al, 2003) based on idea of distributed representations for symbols (Hinton 1986)



- Neural Word embeddings (Mnih and Hinton 2007, Collobert & Weston 2008, Turian et al 2010; Collobert et al. 2011, Mikolov et al. 2011)

Neural distributional representations

- Neural word embeddings
- Combine vector space semantics with the prediction of probabilistic models
- Words are represented as a **dense** vector:

Candy = [0.286, 0.792, −0.177, −0.107, 0.109, −0.542, 0.349, 0.271]

Vector Space Model

In a perfect world:

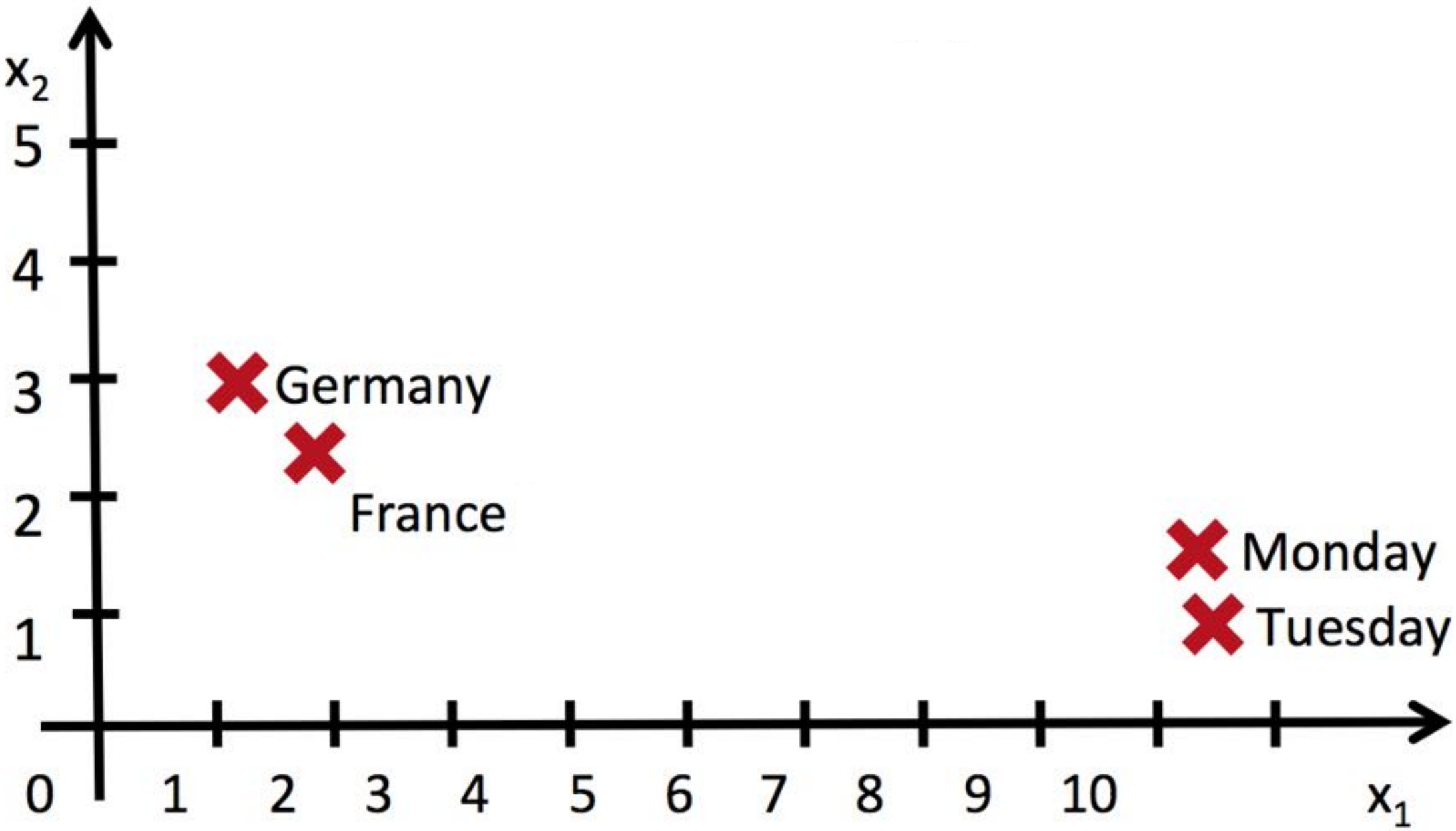


Figure (edited) from Bengio, “Representation Learning and Deep Learning”, July, 2012, UCLA

Vector Space Model

In a perfect world:

input:
- the country of my birth
- the place where I was born

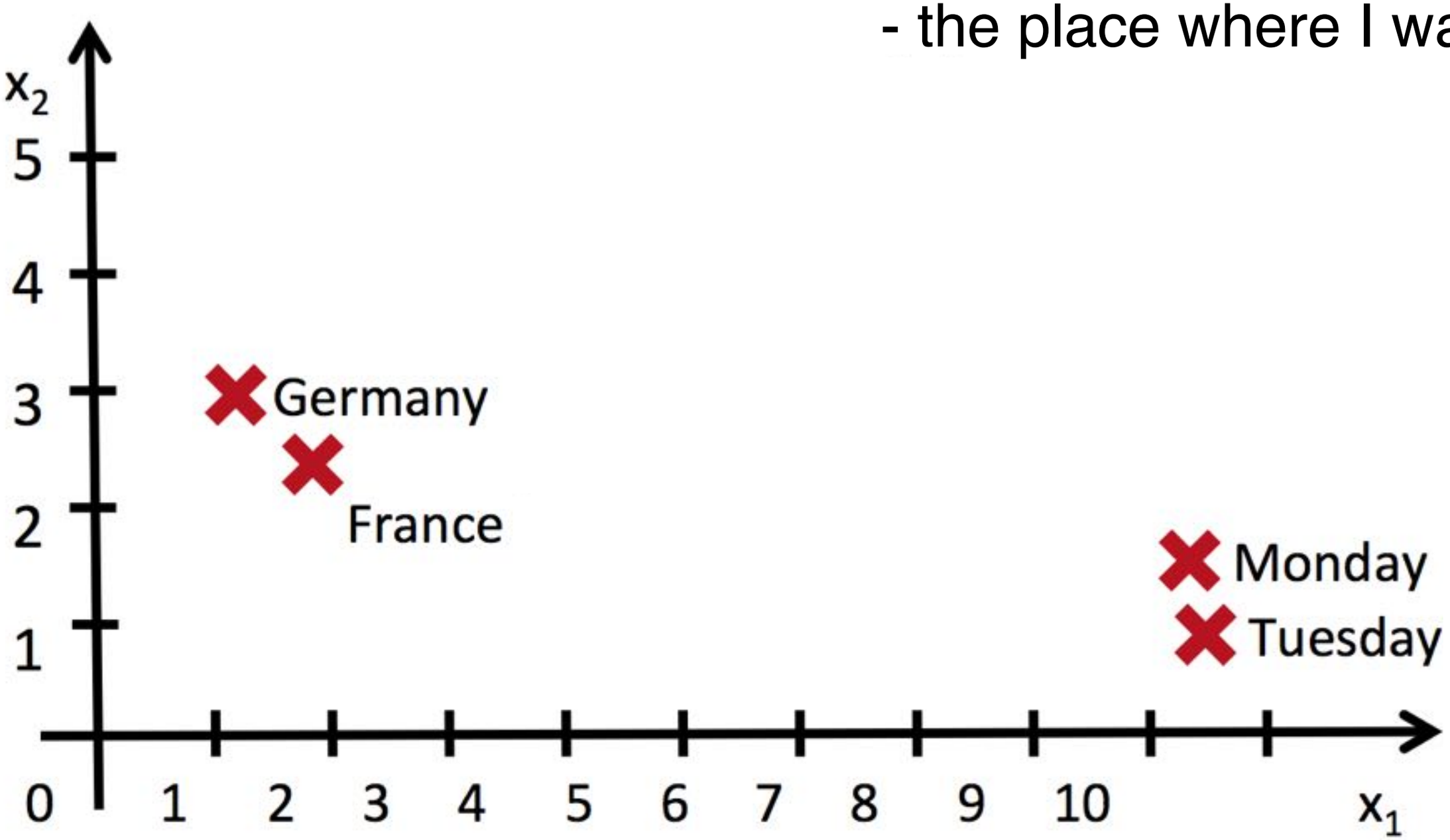


Figure (edited) from Bengio, “Representation Learning and Deep Learning”, July, 2012, UCLA

Vector Space Model

In a perfect world:

input:
- the country of my birth
- the place where I was born

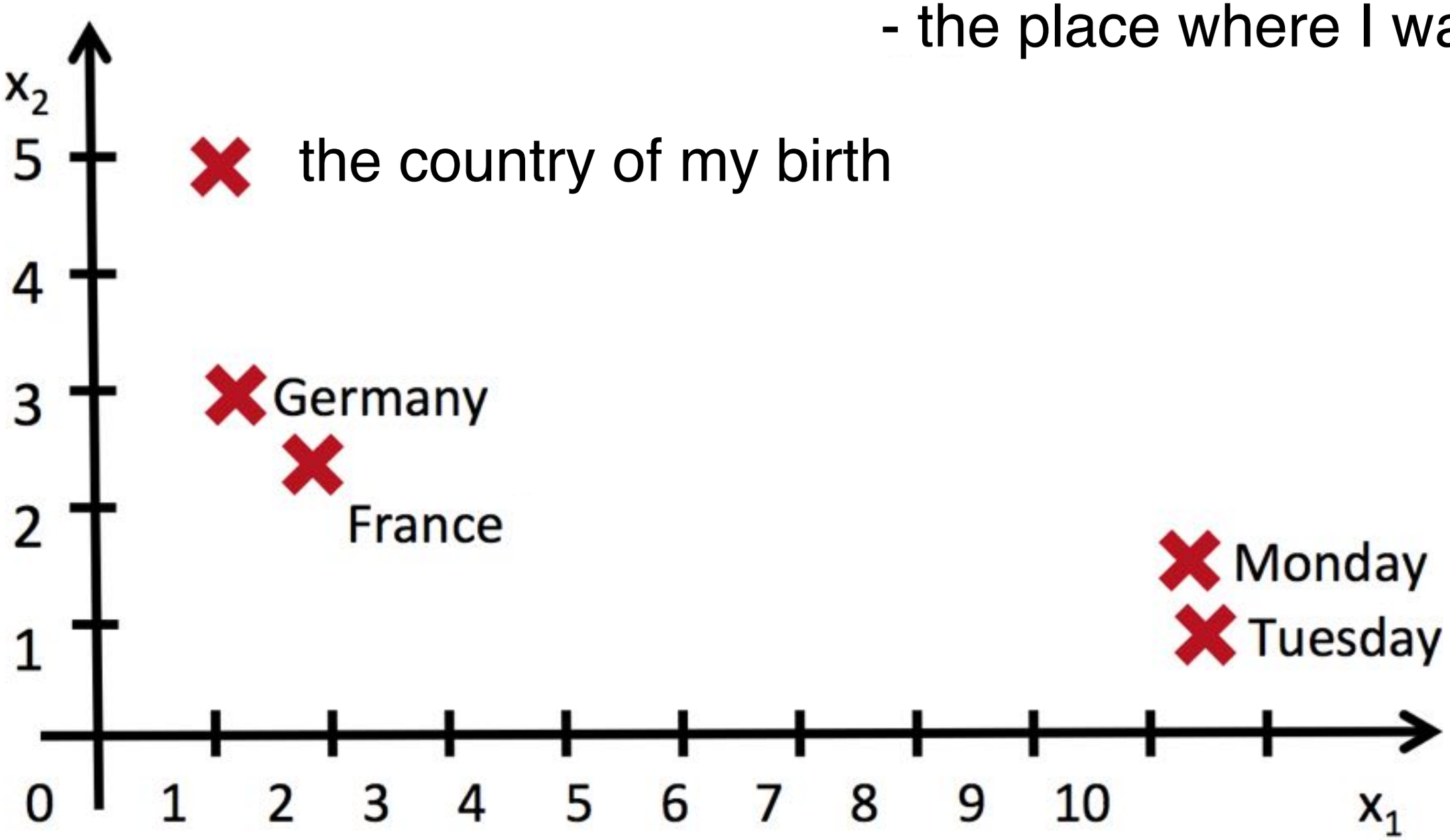


Figure (edited) from Bengio, “Representation Learning and Deep Learning”, July, 2012, UCLA

Vector Space Model

In a perfect world:

input:
- the country of my birth
- the place where I was born

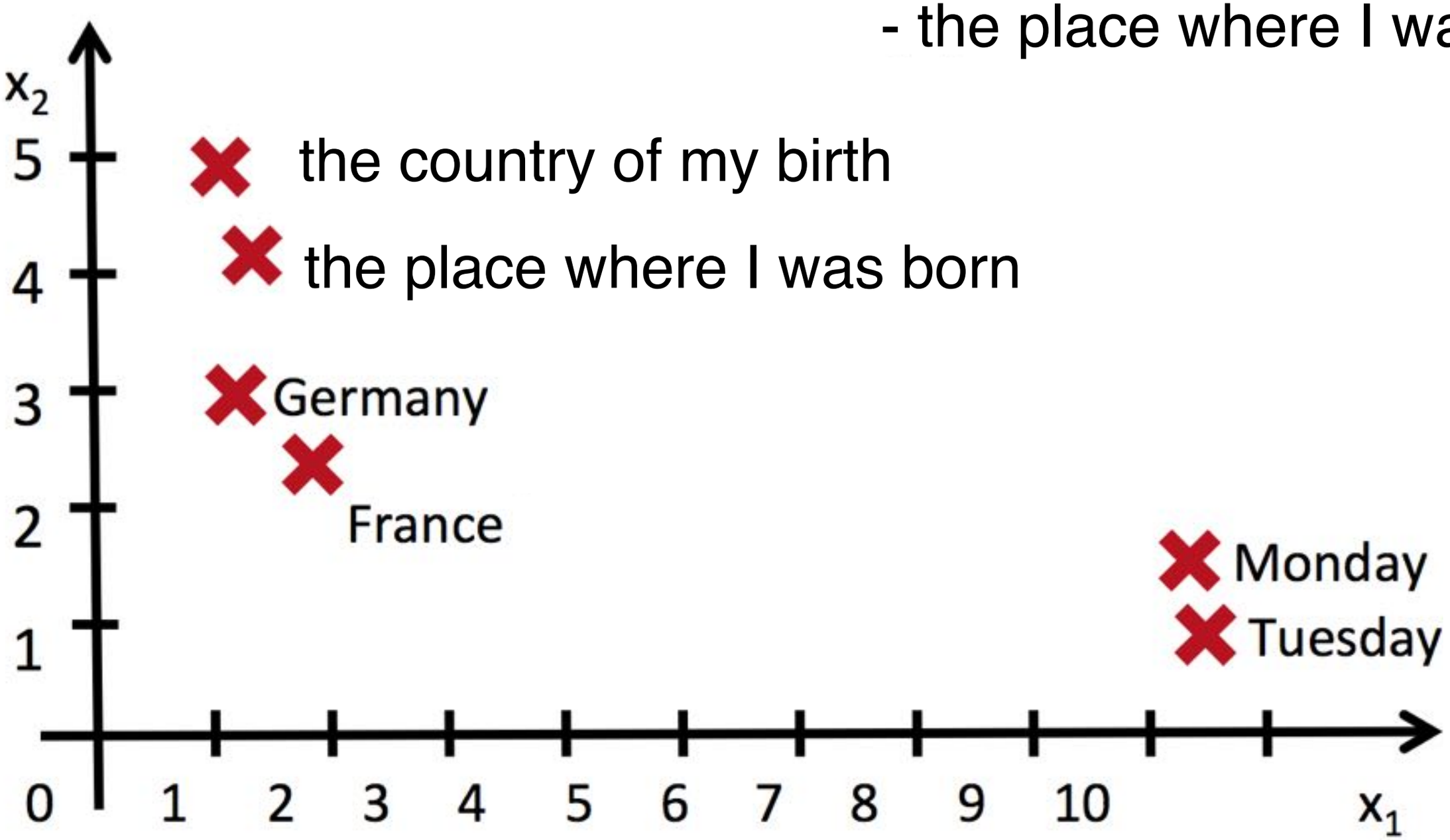


Figure (edited) from Bengio, “Representation Learning and Deep Learning”, July, 2012, UCLA

Vector Space Model

In a perfect world:

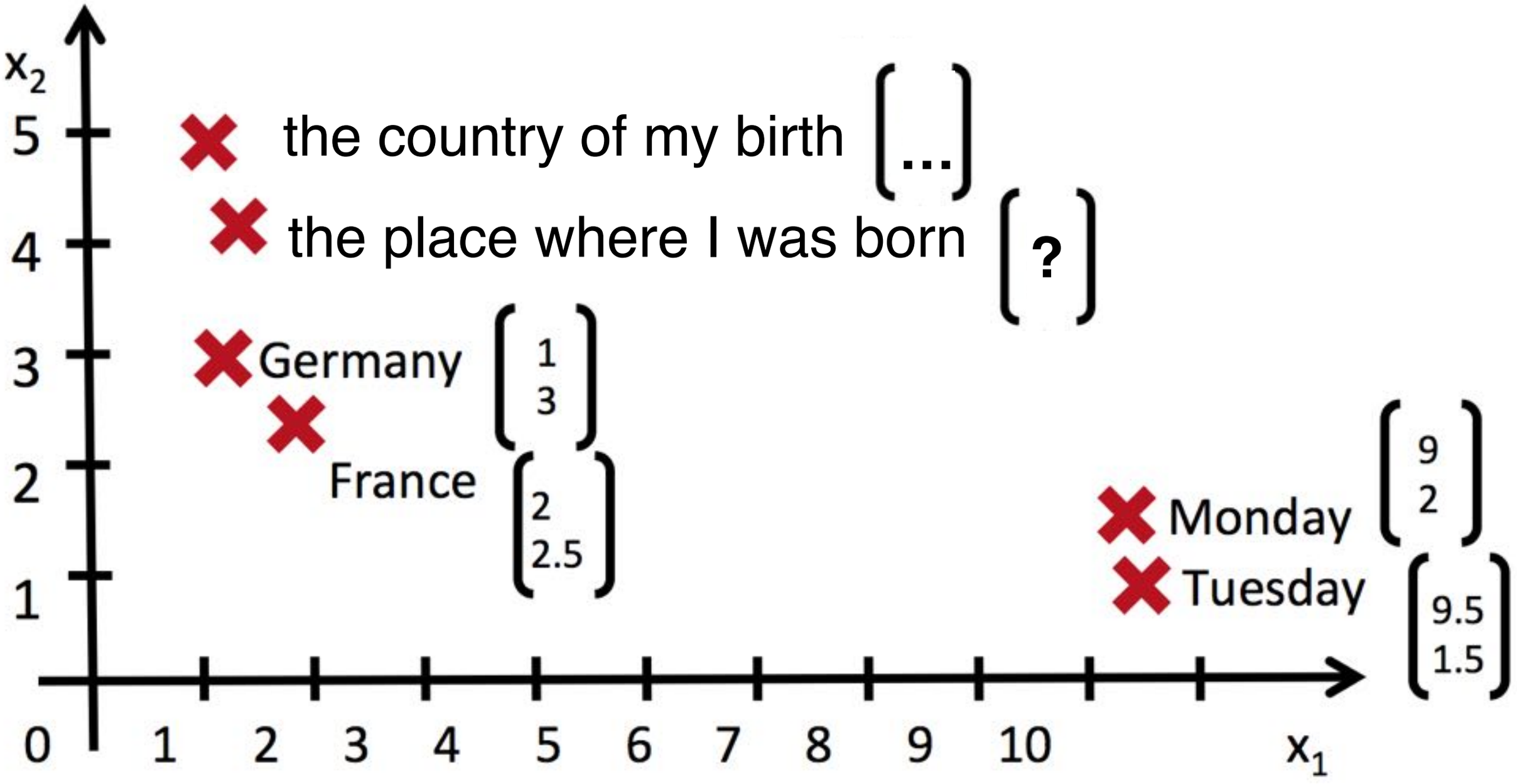


Figure (edited) from Bengio, “Representation Learning and Deep Learning”, July, 2012, UCLA

Recursive Neural (Tensor) Network

- Recursive Tensor (Neural) Network (RTNT)
  (Socher et al. 2011; Socher 2014)

- Top-down hierarchical net (vs feed forward)

- NLP!

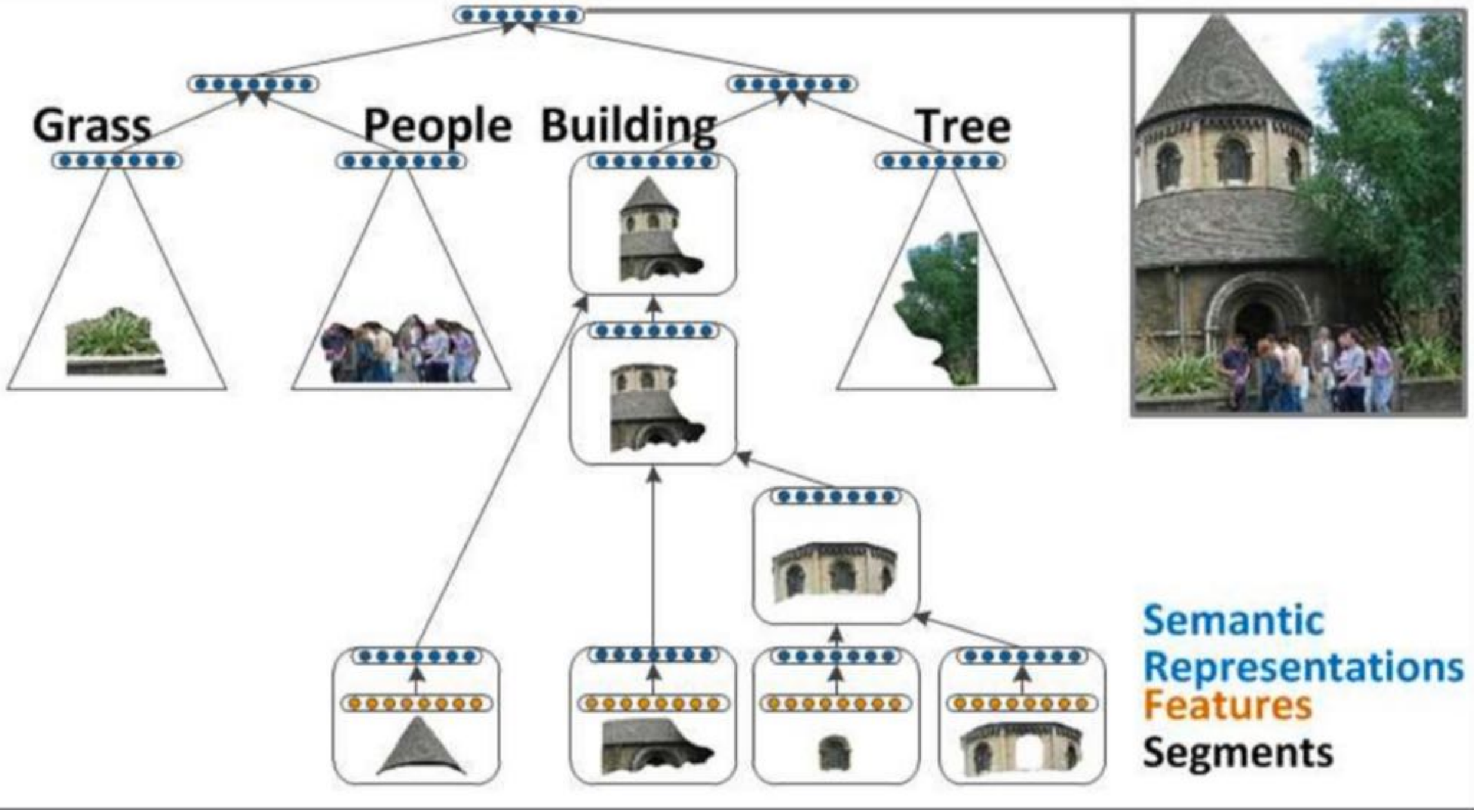
- Sequence based classification, windows of several events, entire scenes (rather than images), entire **sentences** (rather than words)

- Features = Vectors

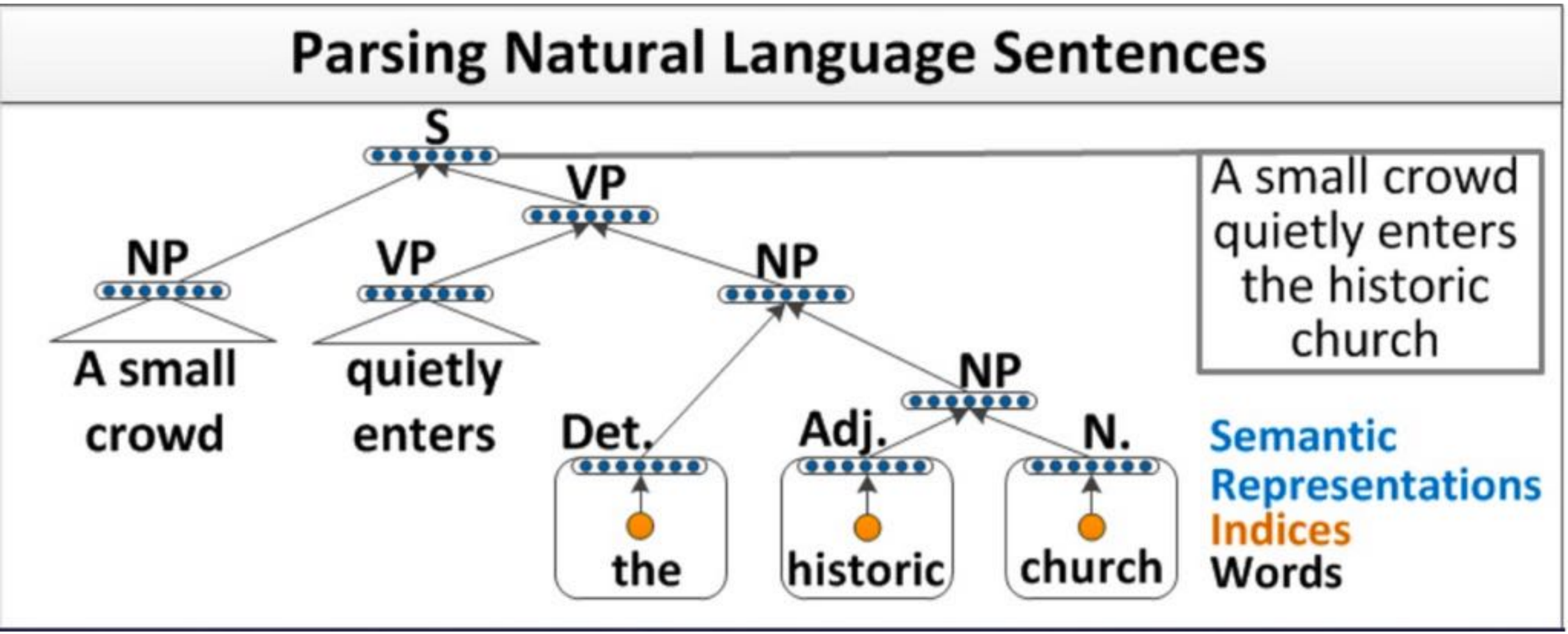
- A tensor = multi-dimensional matrix, or multiple matrices of the same size

Recursive Neural Tensor Network

Parsing Natural Scene Images



Recursive Neural Tensor Network



Compositionality

Principle of compositionality:

the "meaning (**vector**) of a complex expression (**sentence**) is determined by:

- the meanings of its constituent expressions (**words**) and
- the rules (**grammar**) used to combine them"

— Gottlob Frege
(1848 - 1925)

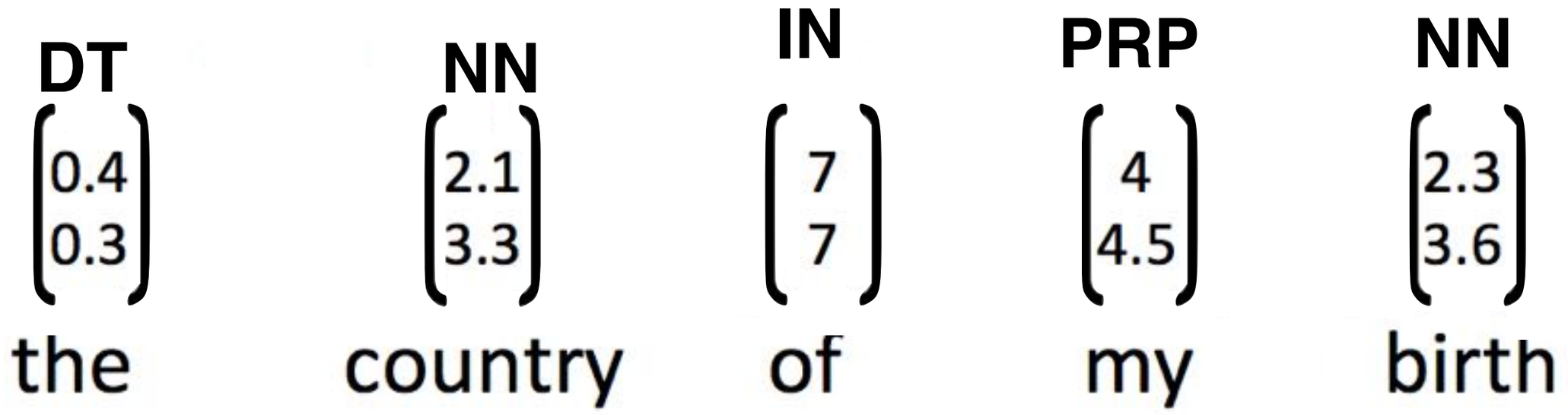
Compositionality

the country of my birth

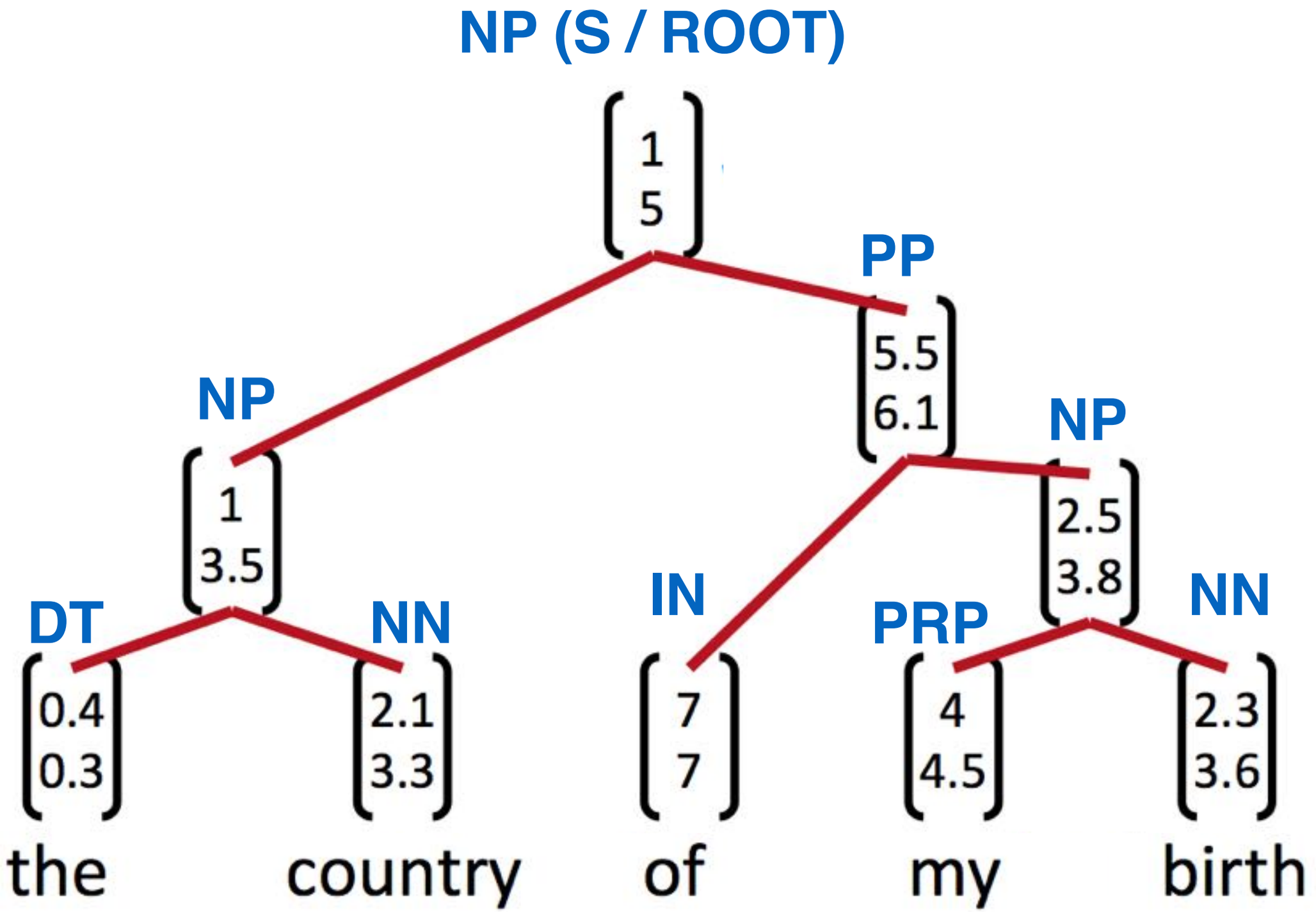
Compositionality

DT	NN	IN	PRP	NN
the	country	of	my	birth

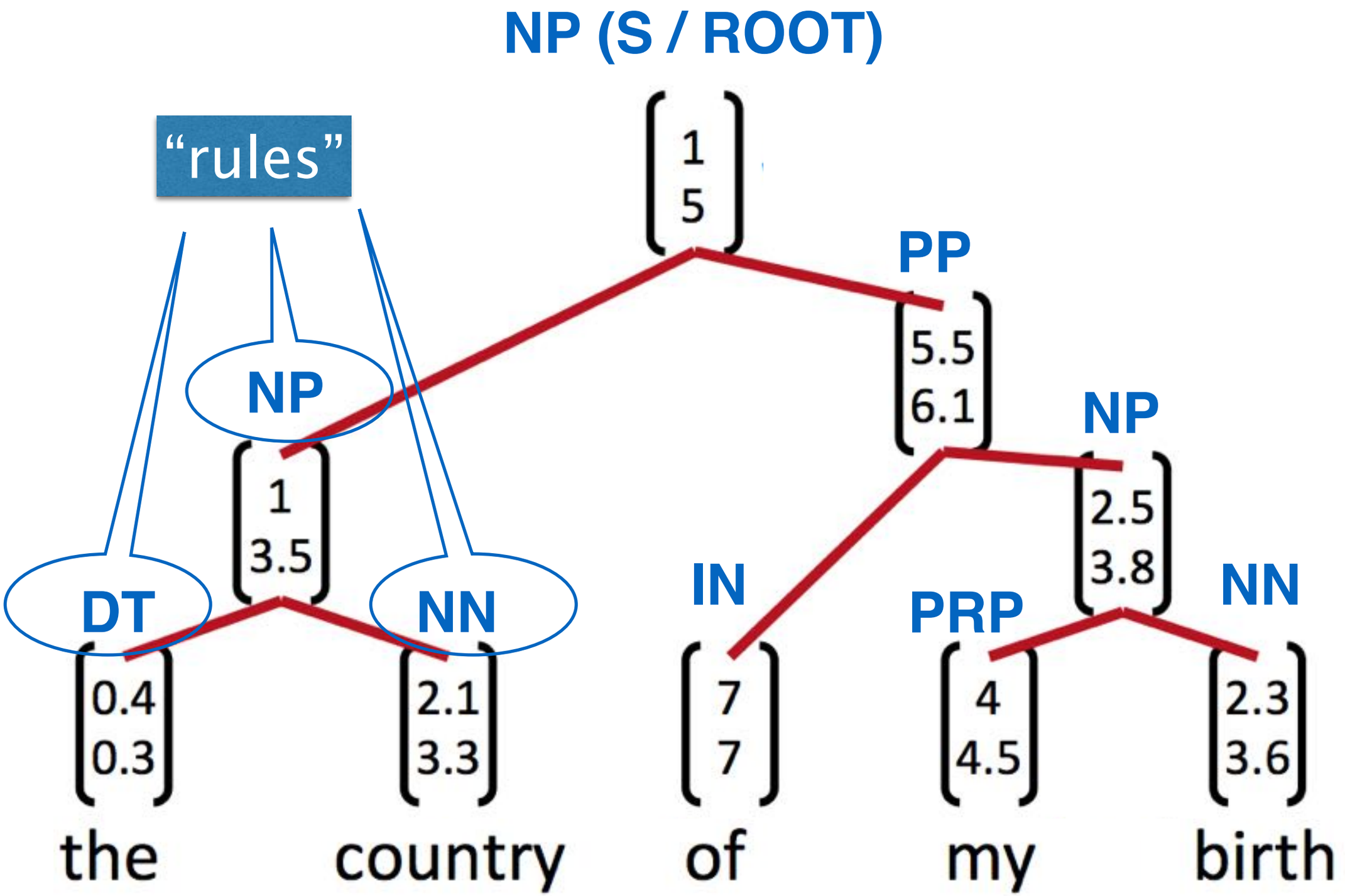
Compositionality



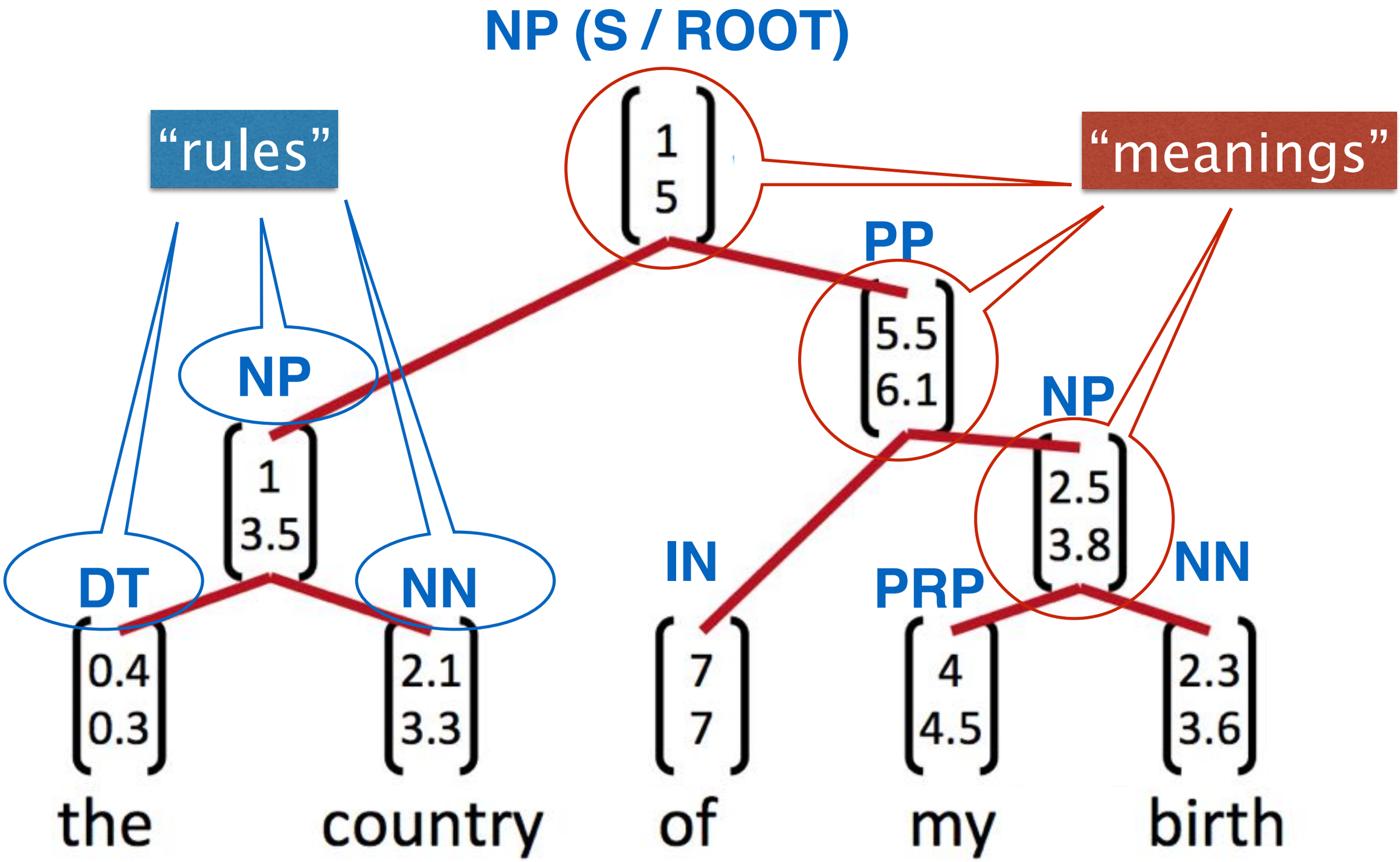
Compositionality



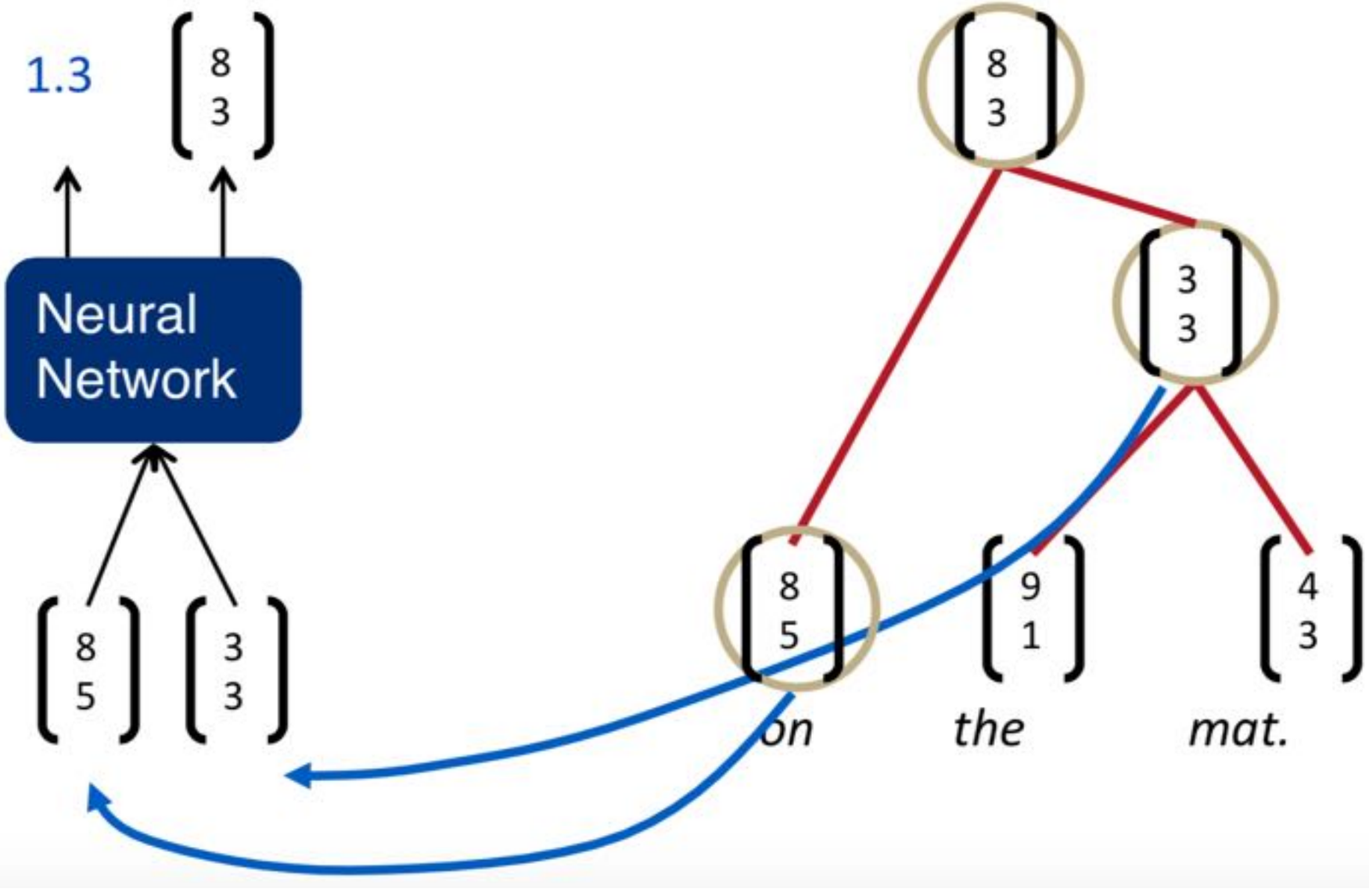
Compositionality



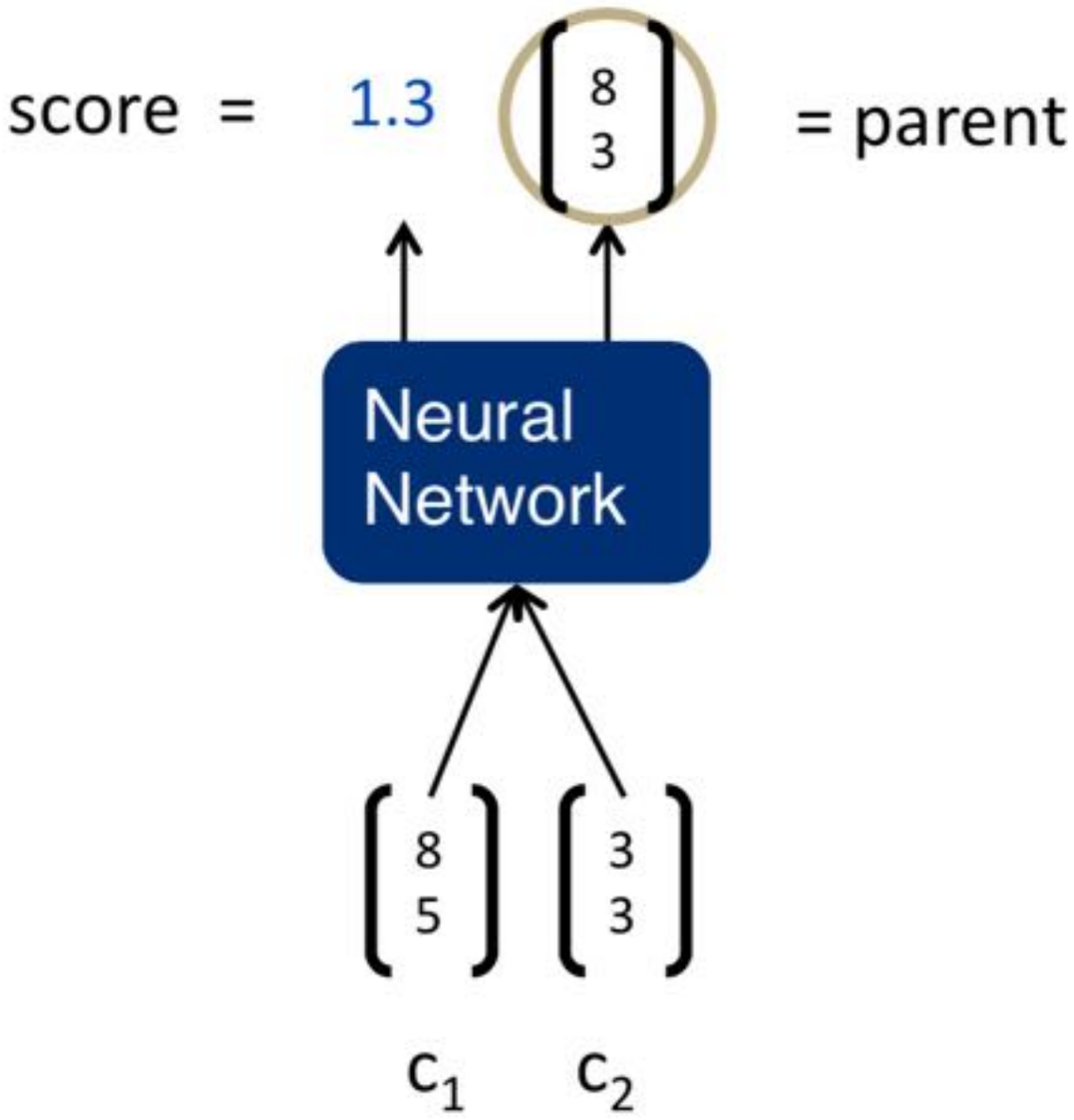
Compositionality



Vector Space + Word Embeddings: Socher



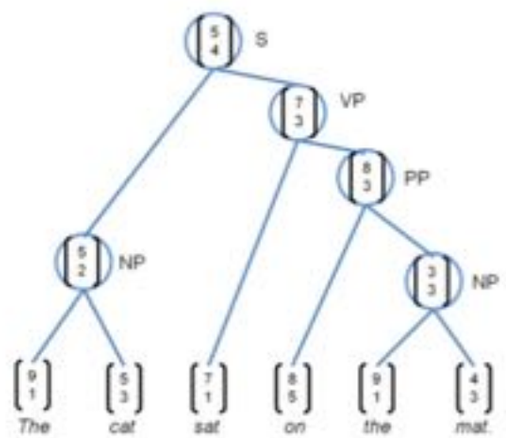
Vector Space + Word Embeddings: Socher



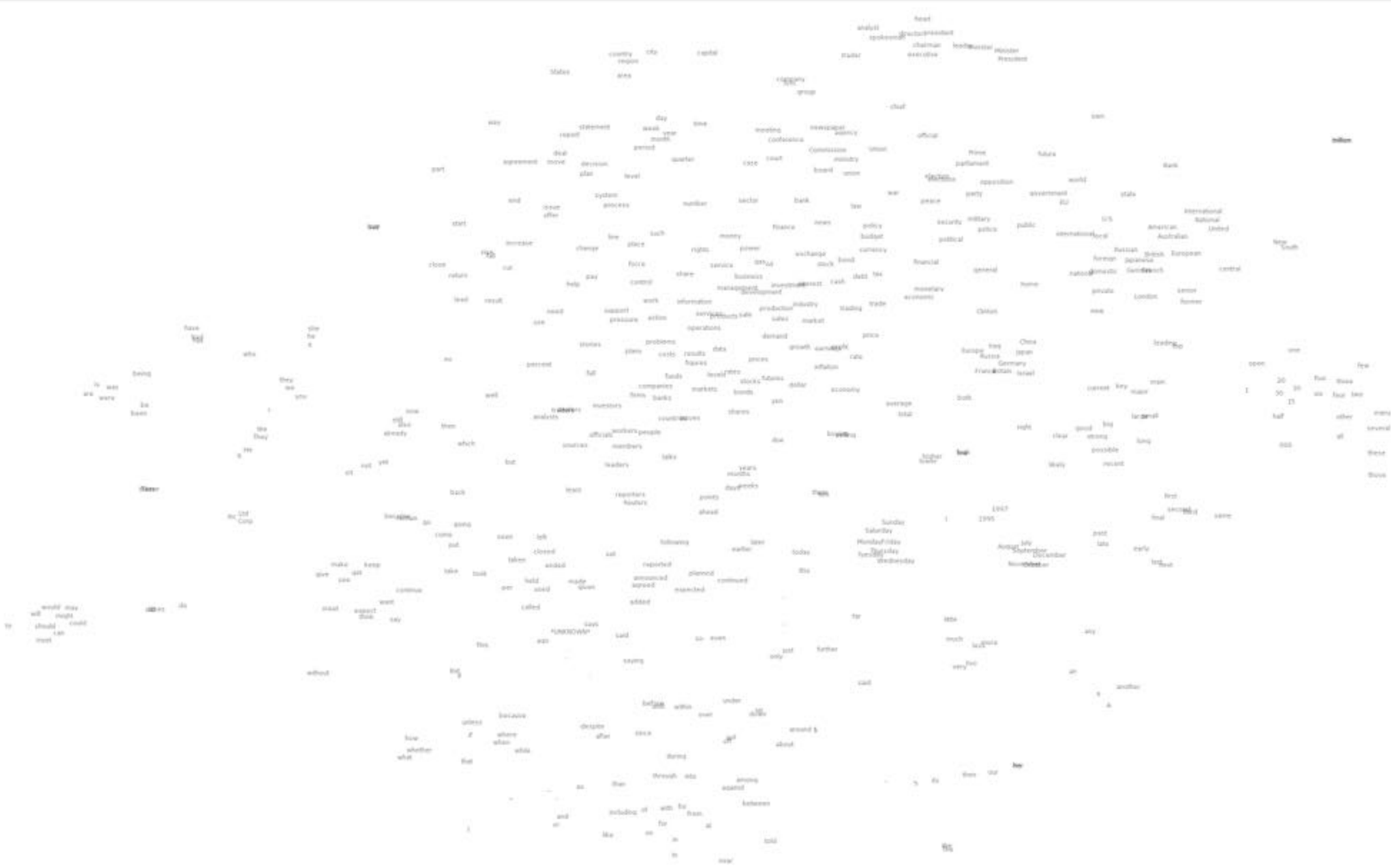
$$\text{score} = U^T p$$

$$p = \tanh\left(W \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} + b\right),$$

Same W parameters at all nodes of the tree



Word Embeddings: Turian



code & info: http://metaoptimize.com/projects/wordreprs/

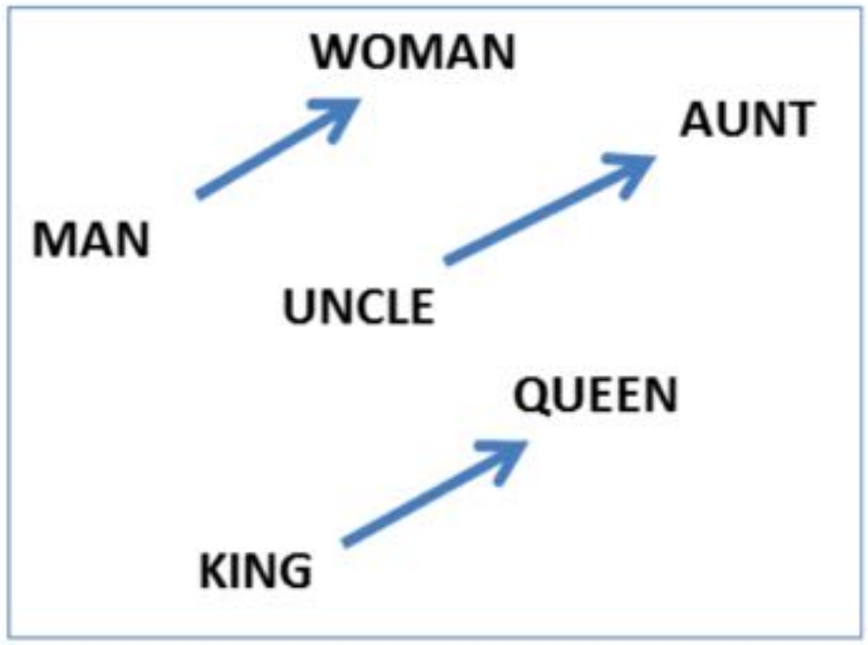
Word Embeddings: Turian

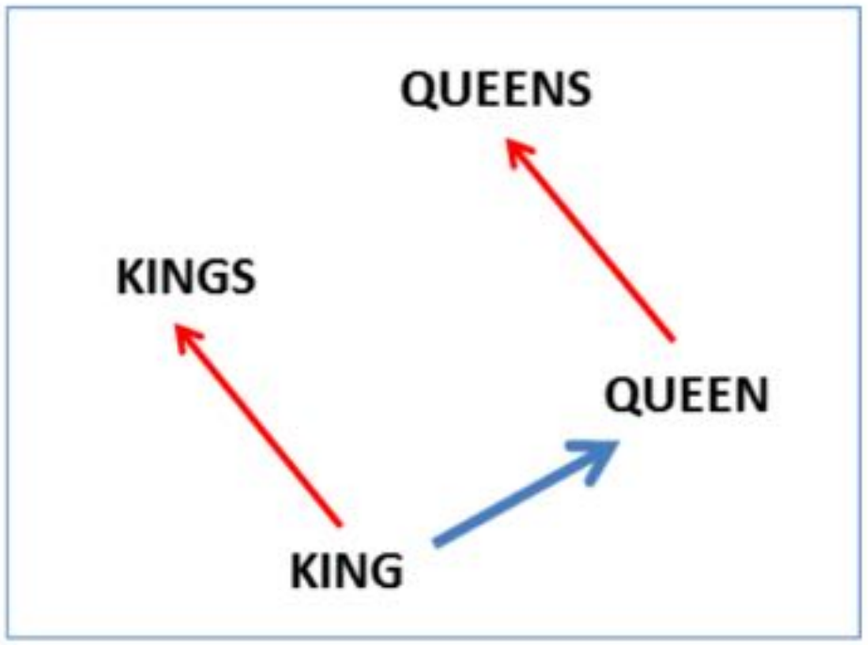
t-SNE visualizations of word embeddings. Left: Number Region; Right: Jobs Region. From Turian et al. 2011

Word Embeddings: Mikolov

- Recurrent Neural Network (Mikolov et al. 2010; Mikolov et al. 2013a)

W(“woman")–W(“man") ≃ W(“aunt")–W(“uncle")
W(“woman")–W(“man") ≃ W(“queen")–W(“king")





Figures from Mikolov, T., Yih, W., & Zweig, G. (2013). Linguistic Regularities in Continuous Space Word Representations

Word Embeddings: Mikolov

- Mikolov et al. 2013b

Type of relationship	Word Pair 1		Word Pair 2	
Common capital city	Athens	Greece	Oslo	Norway
All capital cities	Astana	Kazakhstan	Harare	Zimbabwe
Currency	Angola	kwanza	Iran	rial
City-in-state	Chicago	Illinois	Stockton	California
Man-Woman	brother	sister	grandson	granddaughter
Adjective to adverb	apparent	apparently	rapid	rapidly
Opposite	possibly	impossibly	ethical	unethical
Comparative	great	greater	tough	tougher
Superlative	easy	easiest	lucky	luckiest
Present Participle	think	thinking	read	reading
Nationality adjective	Switzerland	Swiss	Cambodia	Cambodian
Past tense	walking	walked	swimming	swam
Plural nouns	mouse	mice	dollar	dollars
Plural verbs	work	works	speak	speaks

Figures from Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013b). Efficient Estimation of Word Representations in Vector Space

That's all Folks!

Wanna Play ?

- cuda-convnet2 (Alex Krizhevsky, Toronto) (c++/CUDA, optimized for GTX 580)
  https://code.google.com/p/cuda-convnet2/

- Caffe (Berkeley) (Cuda/OpenCL, Theano, Python)
  http://caffe.berkeleyvision.org/

- OverFeat (NYU)
  http://cilvr.nyu.edu/doku.php?id=code:start

Wanna Play ?

- Theano - CPU/GPU symbolic expression compiler in python (from LISA lab at University of Montreal). http://deeplearning.net/software/theano/

- Pylearn2 - library designed to make machine learning research easy. http://deeplearning.net/software/pylearn2/

- Torch - Matlab-like environment for state-of-the-art machine learning algorithms in lua (from Ronan Collobert, Clement Farabet and Koray Kavukcuoglu) http://torch.ch/

- more info: http://deeplearning.net/software links/

Wanna Play with Me ?

Academic/Research

as PhD candidate KTH/CSC:
"Always interested in discussion Machine Learning, Deep Architectures, Graphs, and NLP"

ROYAL INSTITUTE OF TECHNOLOGY

roelof@kth.se
www.csc.kth.se/~roelof/

Internship / Entrepeneurship

as CIO/CTO Feeda:
"Always looking for additions to our brand new R&D team"

[Internships upcoming on KTH exjobb website…]

Feeda

roelof@feeda.com
www.feeda.com

Were Hiring!

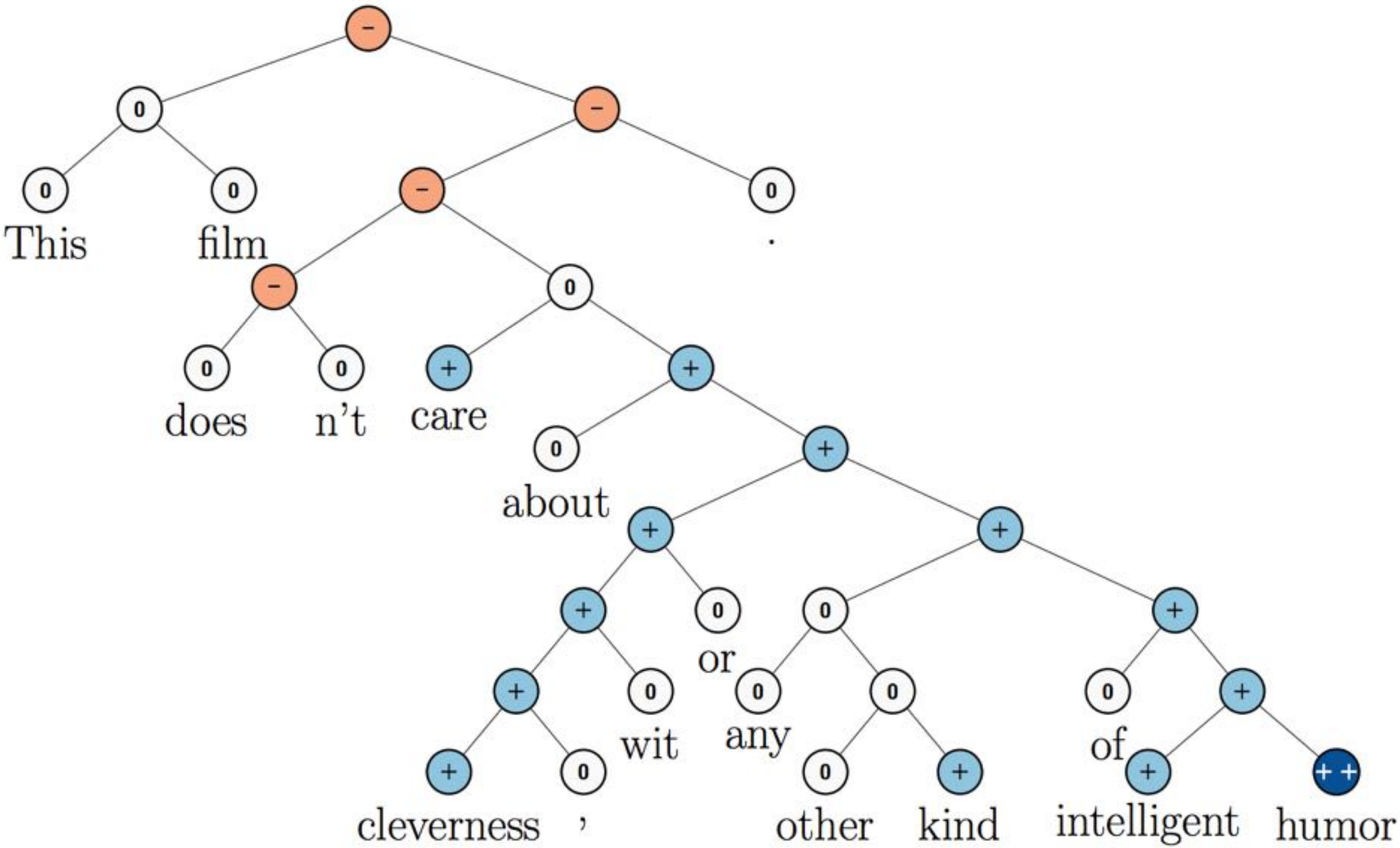
- **Software Developers**
- **Data Scientists**

Feeda

roelof@feeda.com

www.feeda.com

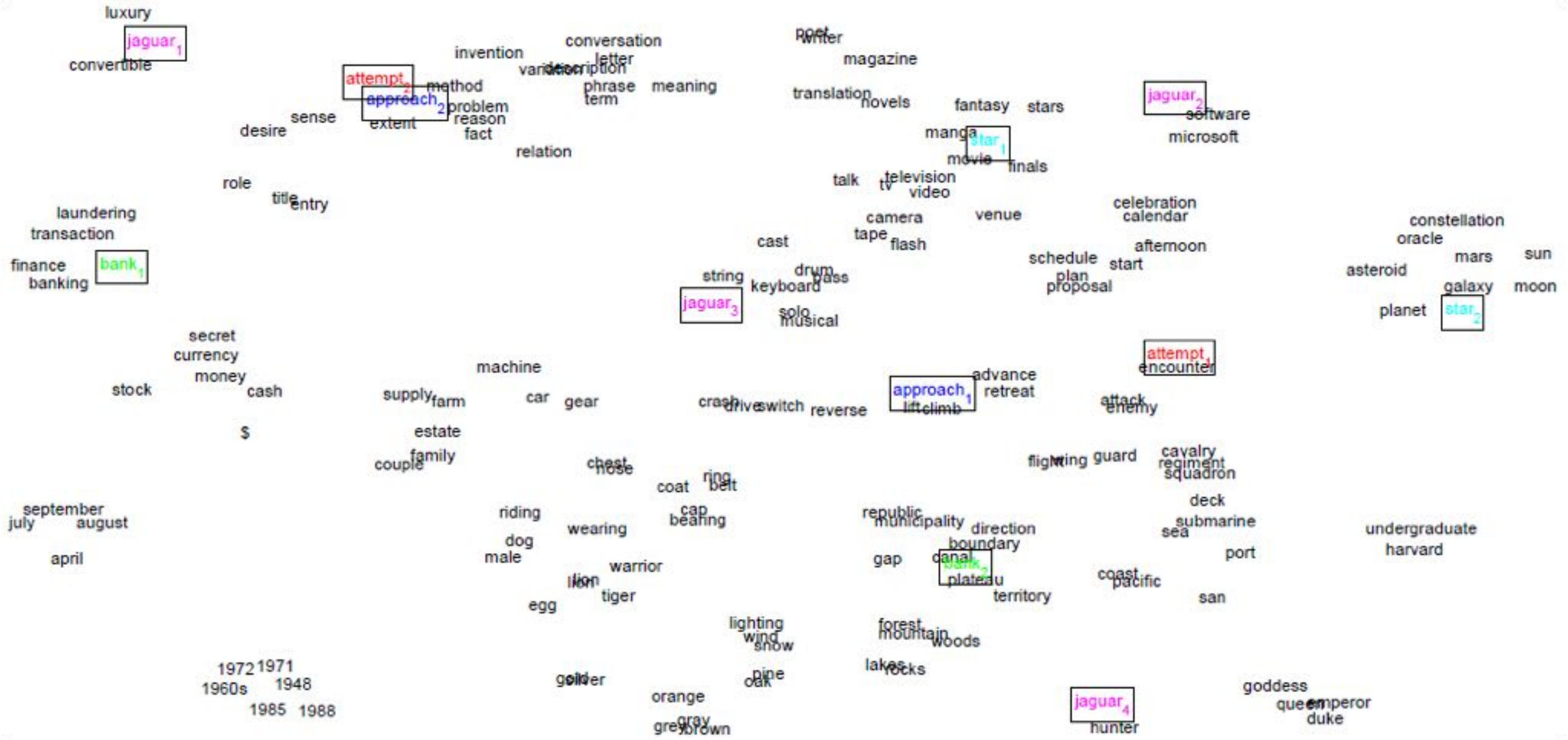
Appendum



Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Chris Manning, Andrew Ng and Chris Potts. 2013. Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank. EMNLP 2013

code & demo: http://nlp.stanford.edu/sentiment/index.html

Appendum



Eric H. Huang, Richard Socher, Christopher D. Manning, Andrew Y. Ng
Improving Word Representations via Global Context and Multiple Word Prototypes