Microsoft Research

Deep Learning for Speech/Language Processing

--- machine learning & signal processing perspectives

Li Deng

Deep Learning Technology Center

Microsoft Research, Redmond, USA

Tutorial given at Interspeech, Sept 6, 2015

Thanks go to many colleagues at DLTC/MSR, collaborating universities, and at Microsoft's engineering groups

Microsoft Research

Outline

- Part I: Basics of Machine Learning (Deep and Shallow) and of Signal Processing
- Part II: Speech
- Part III: Language

(In case you did not get link to slides, send email to: alexander.raake@tu-ilmenau.de)

Microsoft Research

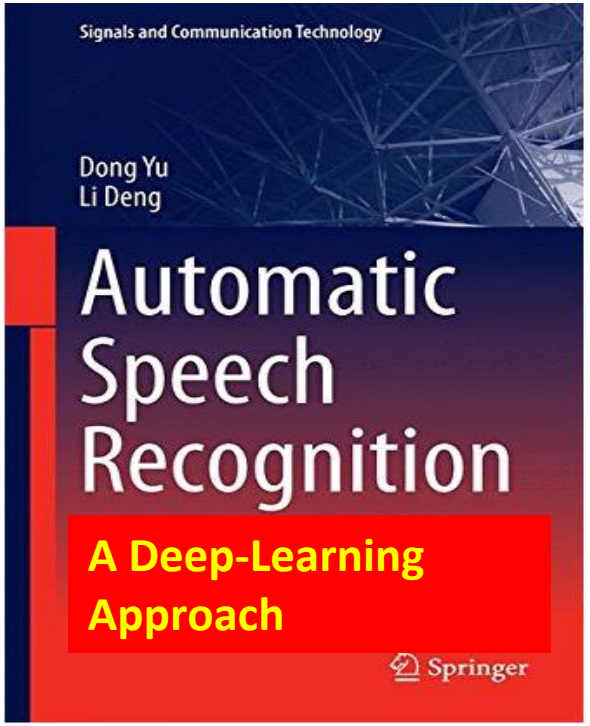
Reading Material

Books:

Bengio, Yoshua (2009). "Learning Deep Architectures for AI".

L. Deng and D. Yu (2014) "Deep Learning: Methods and Applications"
http://research.microsoft.com/pubs/209355/DeepLearning-NowPublishing-Vol7-SIG-039.pdf

D. Yu and L. Deng (2014). "Automatic Speech Recognition: A Deep Learning Approach" (Publisher: Springer).



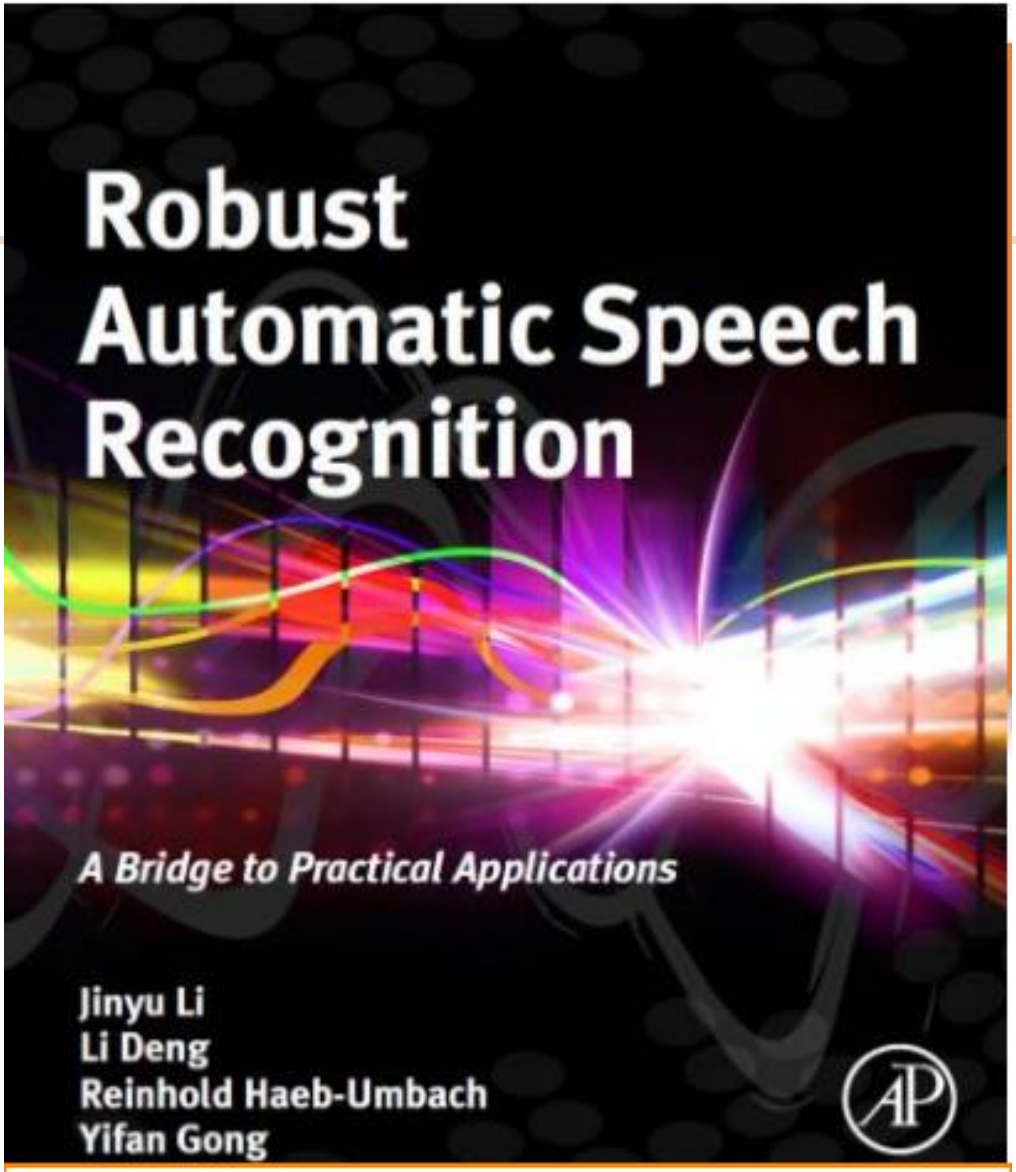
Reading Material

DEEP LEARNING

An MIT Press book in preparation

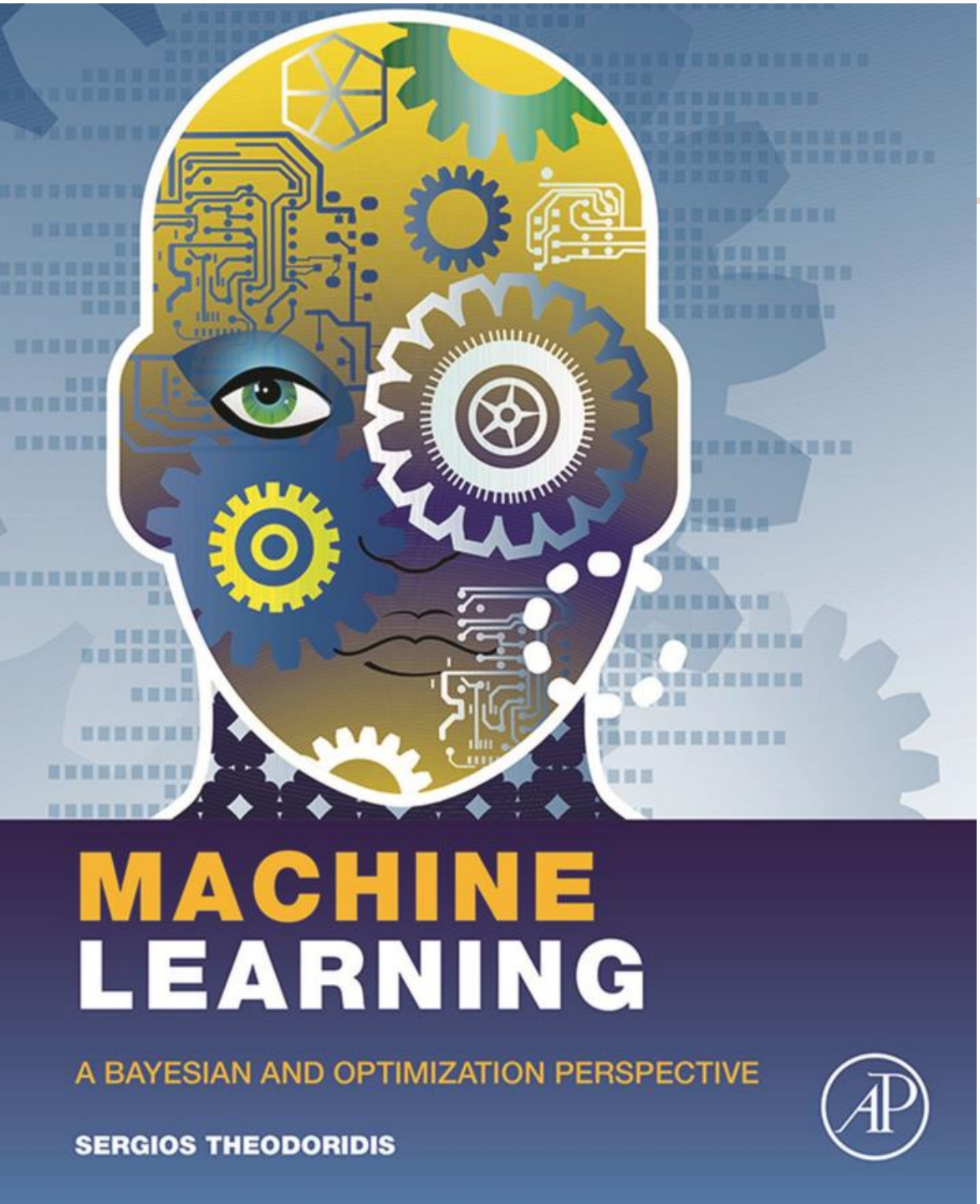
Yoshua Bengio, Ian Goodfellow and Aaron Courville

Microsoft Research



ISBN: 978-0-12-802398-3

PUB DATE: October 2015


MACHINE
LEARNING
A BAYESIAN AND OPTIMIZATION PERSPECTIVE
SERGIOS THEODORIDIS

Microsoft Research

Reading Material (cont'd)

Wikipedia:
https://en.wikipedia.org/wiki/Deep_learning

Papers:

G. E. Hinton, R. Salakutdinov. "Reducing the Dimensionality of Data with Neural Networks". *Science* **313**: 504–507, 2016.

G. E. Hinton, L. Deng, D. Yu, etc. "Deep Neural Networks for Acoustic Modeling in Speech Recognition: The shared views of four research groups," *IEEE Signal Processing Magazine*, pp. 82–97, November 2012.

G. Dahl, D. Yu, L. Deng, A. Acero. "Context-Dependent Pre-Trained Deep Neural Networks for Large-Vocabulary Speech Recognition". IEEE Trans. *Audio, Speech, and Language Processing, Vol* 20(1): 30–42, 2012. (plus other papers in the same special issue)

Y. Bengio, A. Courville, and P. Vincent. "Representation Learning: A Review and New Perspectives," *IEEE Trans. PAMI, special issue Learning Deep Architectures*, 2013.

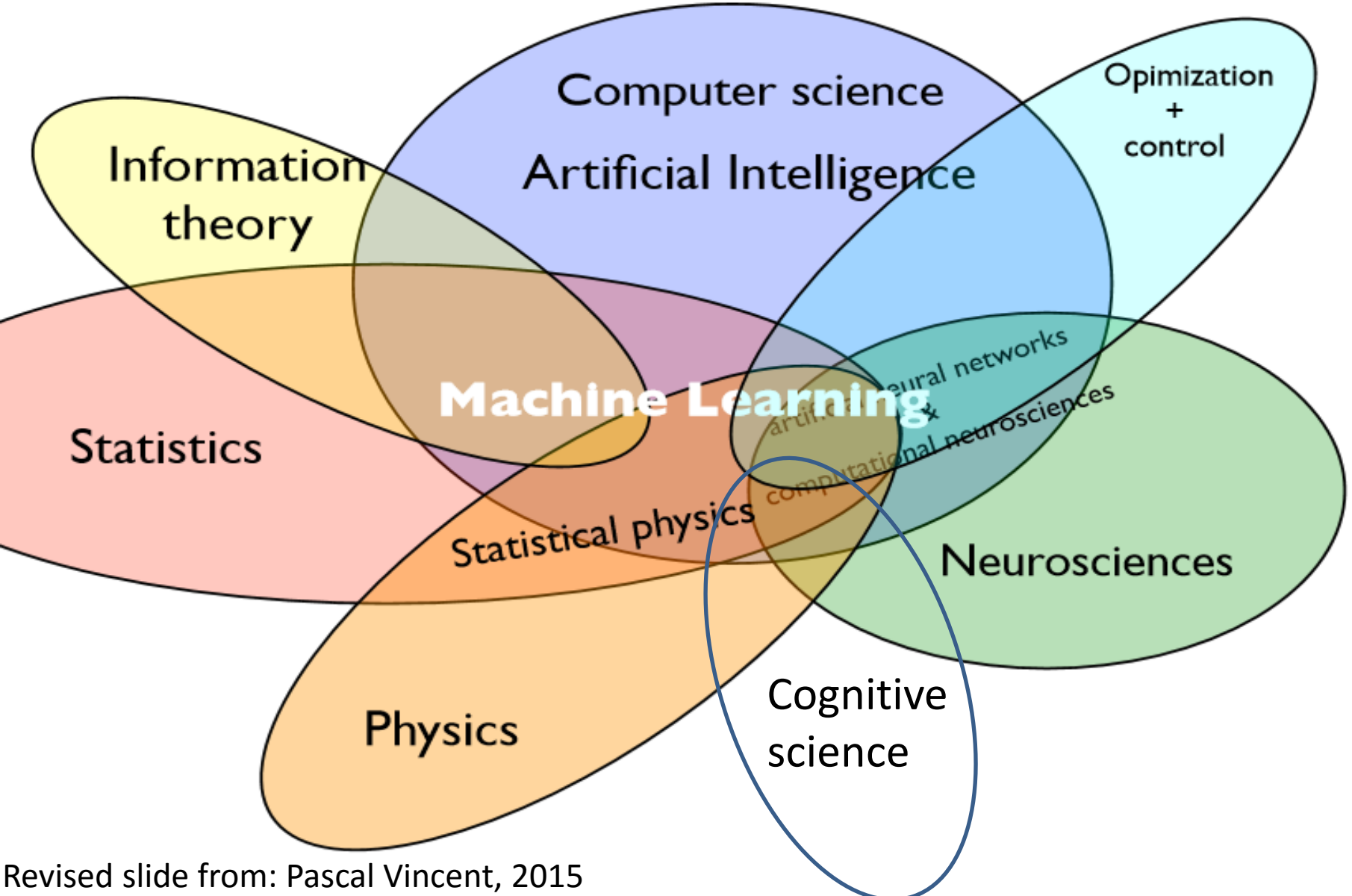
J. Schmidhuber. "Deep learning in neural networks: An overview," arXiv, October 2014.

Y. LeCun, Y. Bengio, and G. Hinton. "Deep Learning", Nature, Vol. 521, May 2015.

J. Bellegarda and C. Monz. "State of the art in statistical methods for language and speech processing," Computer Speech and Language, 2015

Part I: Machine Learning (Deep/Shallow) and Signal Processing

Current view of ML founding disciplines

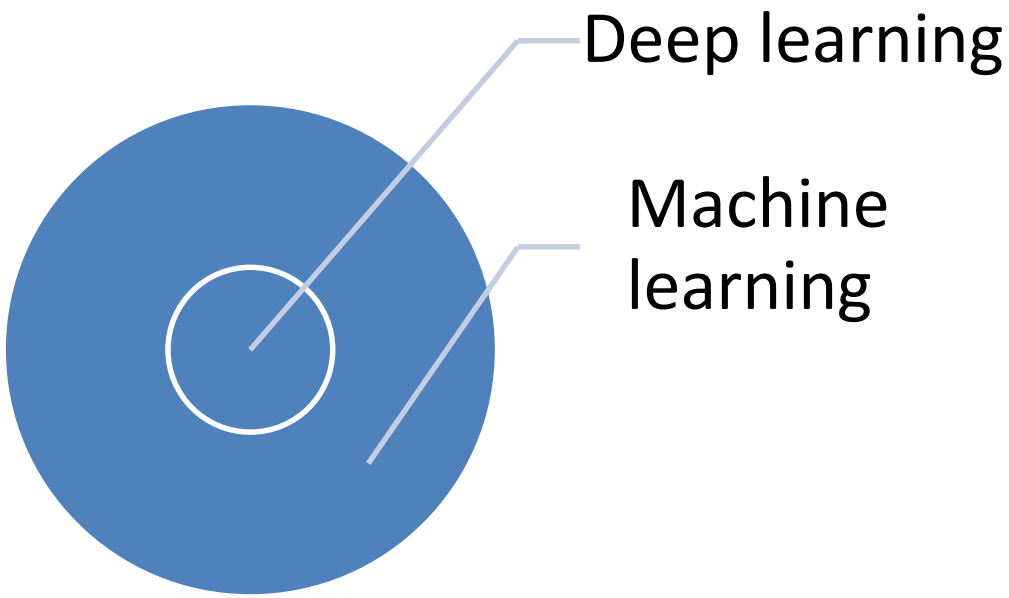


Revised slide from: Pascal Vincent, 2015

Microsoft Research

Machine Learning & Deep Learning

Machine learning



What Is Deep Learning?

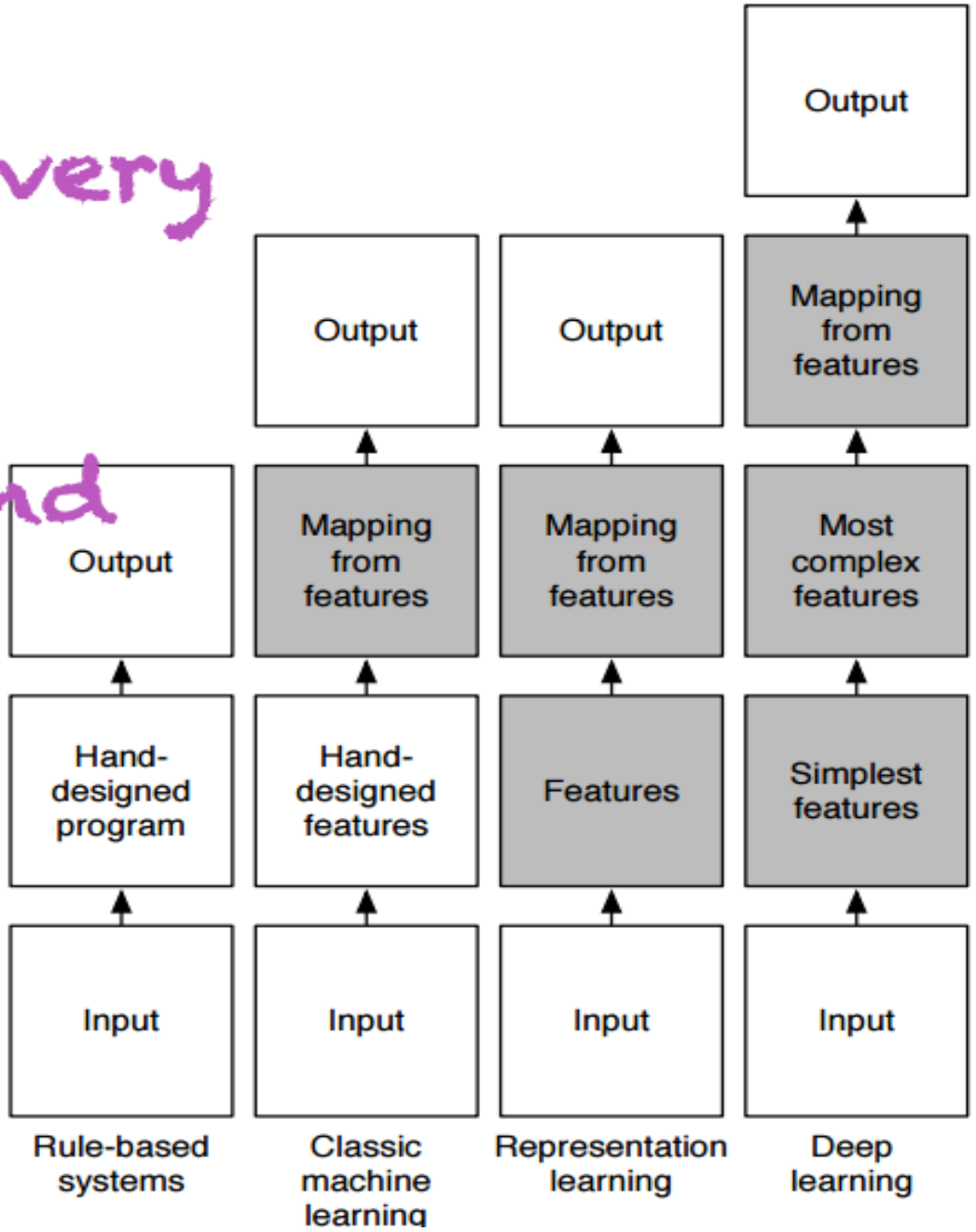
Deep learning

From Wikipedia, the free encyclopedia

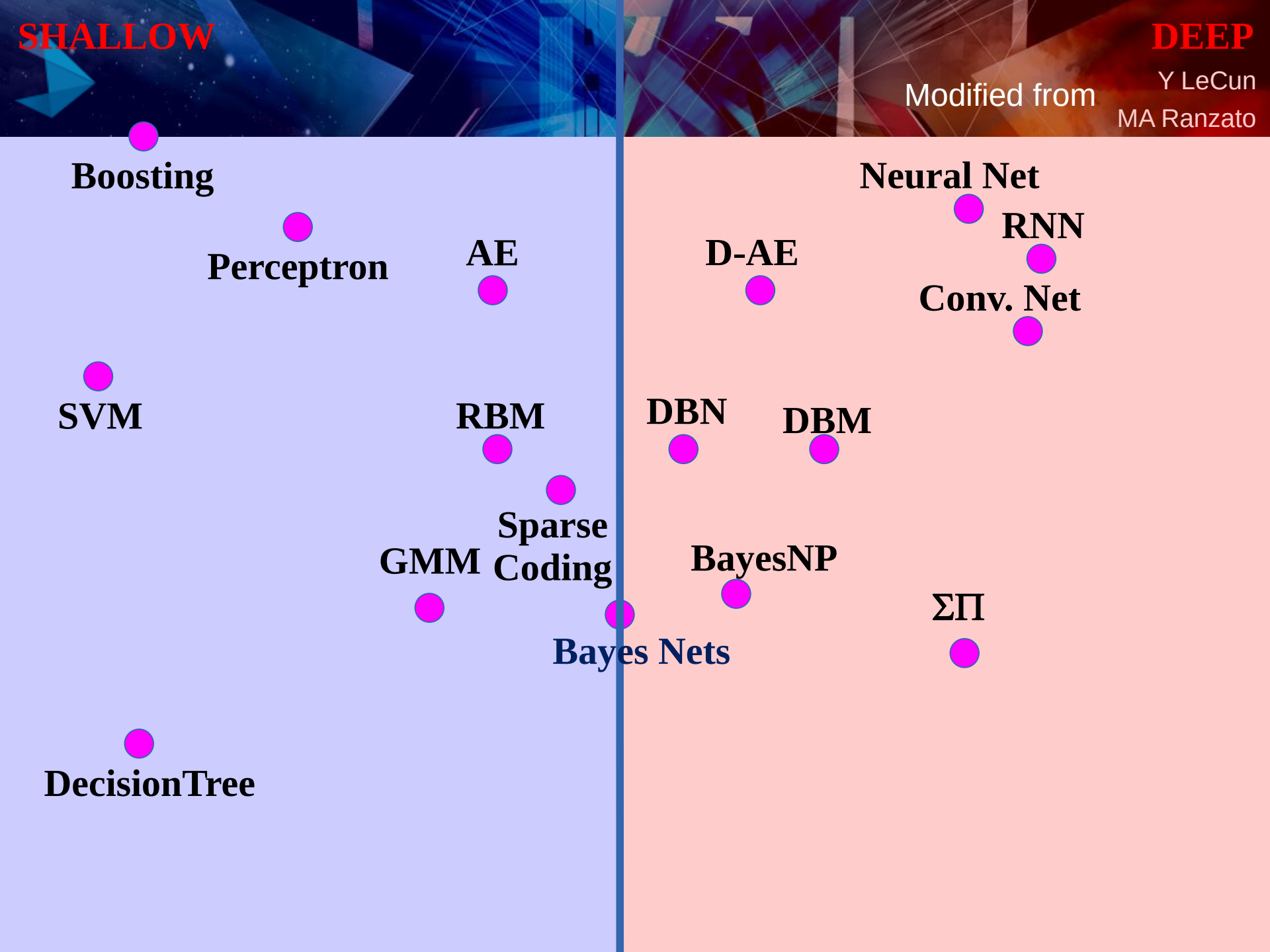
Deep learning (*deep machine learning*, or *deep structured learning*, or *hierarchical learning*, or sometimes *DL*) is a branch of machine learning based on a set of algorithms that attempt to model high-level abstractions in data by using model architectures, with complex structures or otherwise, composed of multiple non-linear transformations.[1](p198)[2][3][4]

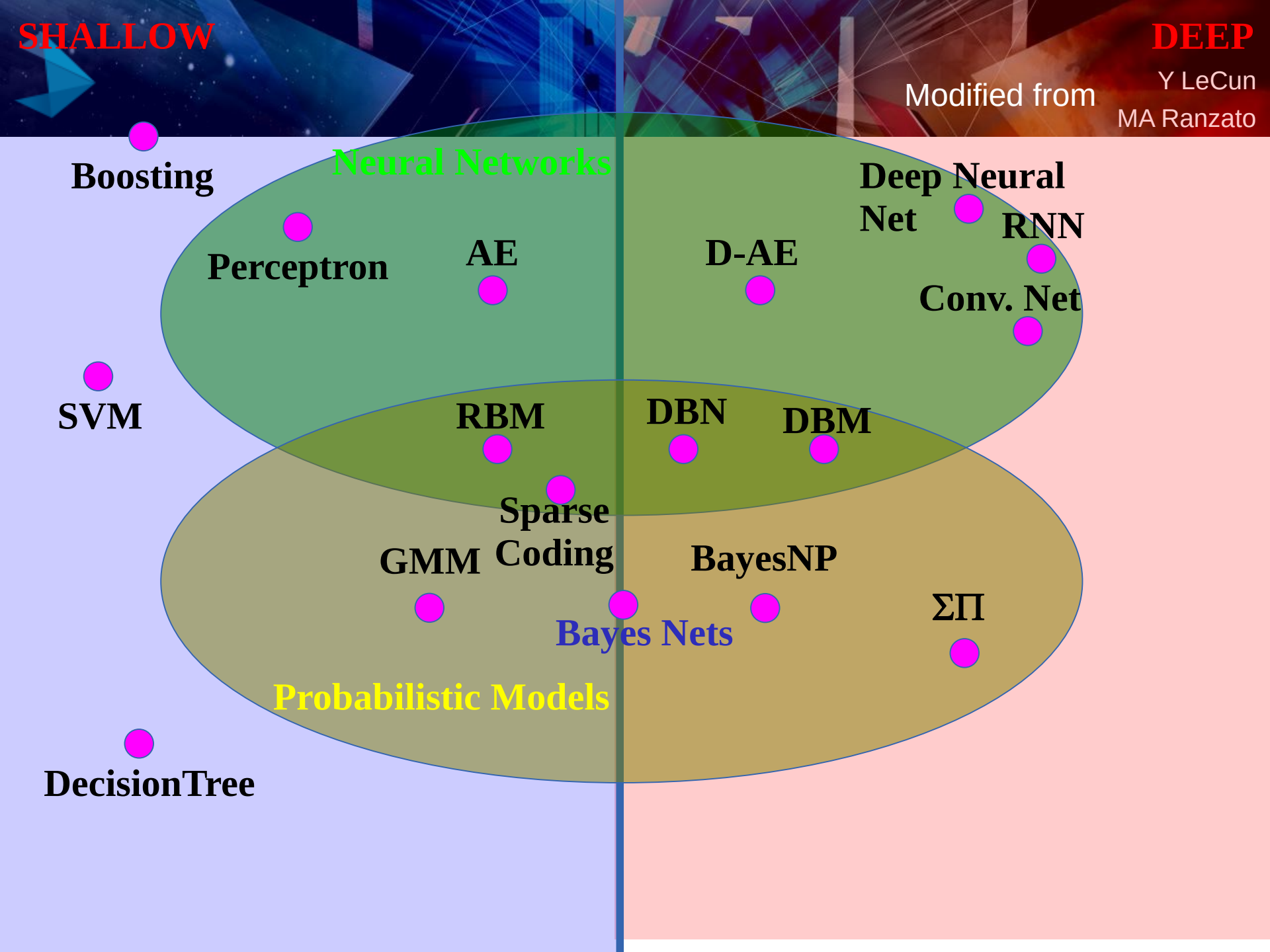
Automating Feature Discovery

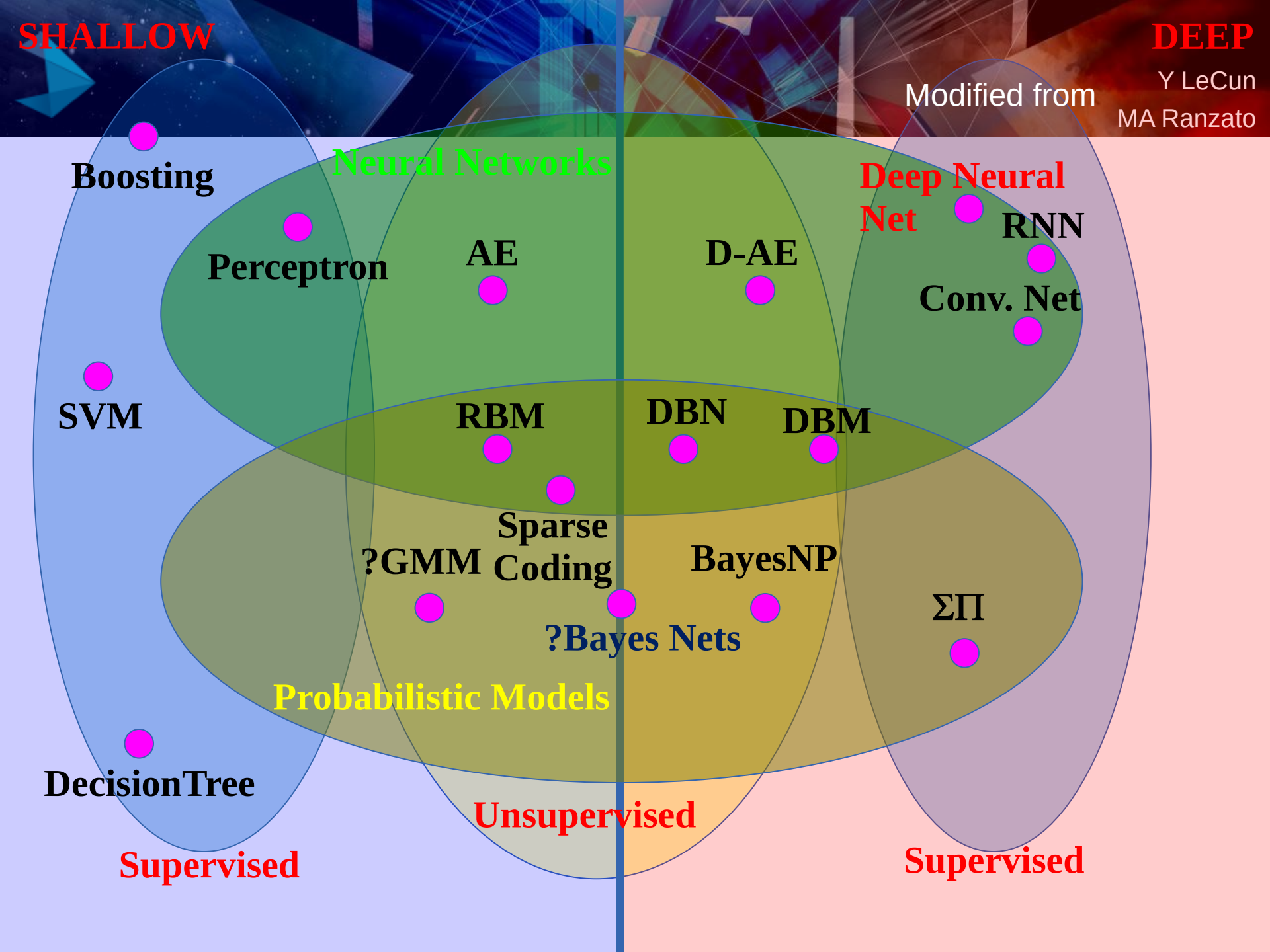
Discovering and representing higher-level abstractions



(Slide from: Yoshua Bengio)


SHALLOW
DEEP
Modified from
Y LeCun
MA Ranzato
Boosting
Perceptron
AE
D-AE
Neural Net
RNN
Conv. Net
SVM
RBM
DBN
DBM
Sparse
Coding
GMM
BayesNP
ΣΠ
Bayes Nets
DecisionTree


SHALLOW
DEEP
Modified from
Y LeCun
MA Ranzato
Boosting
Neural Networks
Deep Neural Net
RNN
Perceptron
AE
D-AE
Conv. Net
SVM
RBM
DBN
DBM
Sparse Coding
GMM
BayesNP
ΣΠ
Bayes Nets
Probabilistic Models
DecisionTree


SHALLOW
DEEP
Modified from
Y LeCun
MA Ranzato
Boosting
Neural Networks
Deep Neural Net
RNN
Perceptron
AE
D-AE
Conv. Net
SVM
RBM
DBN
DBM
Sparse Coding
?GMM
BayesNP
ΣΠ
?Bayes Nets
Probabilistic Models
DecisionTree
Unsupervised
Supervised
Supervised

Signal Processing → Information Processing

Signals / Processing	Audio/Music	Speech	Image/ Animation/ Graphics	Video	Text/ Language
Coding/ Compression	Audio Coding	Speech Coding	Image Coding	Video Coding	Document Compression/ Summary
Communication	Voice over IP, DAB,etc		4G/5G Networks, DVB, Home Networking, etc		
Security	Multimedia watermarking, encryption, etc.				
Enhancement/ Analysis	De-noising/ Source separation	Speech Enhancement/ Feature extraction	Image/video enhancement (Clear Type), Segmentation, feature extraction		Grammar checking, Text Parsing
Synthesis/ Rendering	Computer Music	Speech Synthesis (text-to-speech)	Computer Graphics/	Video Synthesis	Natural Language Generation
User-Interface	Multi-Modal Human Computer Interaction (HCI --- Input Methods)				
Recognition	Auditory Scene Analysis (Computer audition; e.g. Melody Detection & Singer ID)	Automatic Speech/Speaker Recognition	Image Recognition	Computer Vision (e.g. 3-D object recognition)	Document Recognition
Understanding (Semantic IE)		Spoken Language Understanding	Image Understanding		Natural Language Understanding
Retrieval/Mining	Music Retrieval	Spoken Document Retrieval & Voice/Mobile Search	Image Retrieval	Video Search	Text Search (info retrieval)
translation		Speech translation			Machine translation
Social Media Apps	.	.	Photo Sharing (e.g. flickr)	Video Sharing (e.g. Youtube	Blogs, Wiki, del.icio.us

Machine Learning Basics

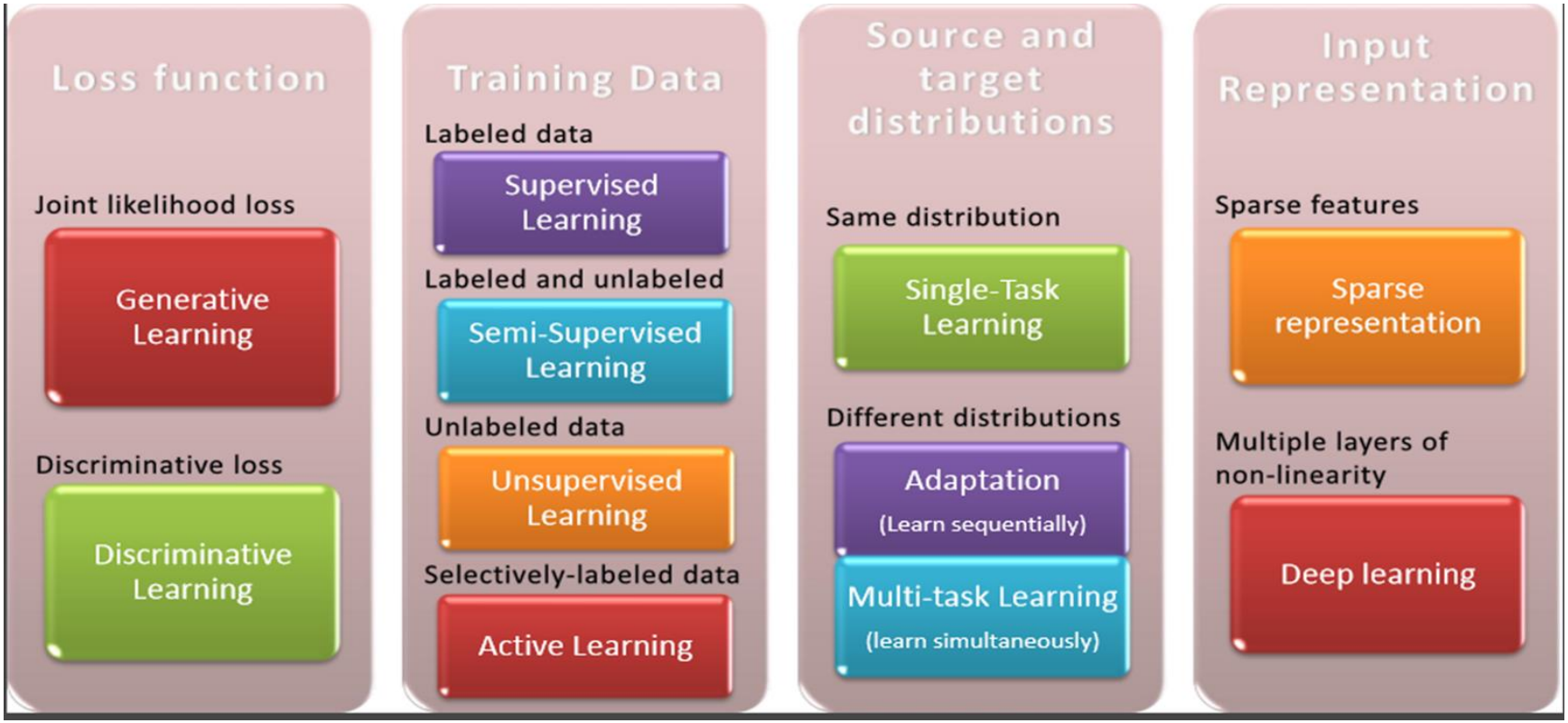
1060-1089 IEEE TRANSACTIONS ON AUDIO, SPEECH, AND LANGUAGE PROCESSING, VOL. 21, NO. 5, MAY 2013

Machine Learning Paradigms for Speech Recognition: An Overview

Li Deng, *Fellow, IEEE*, and Xiao Li, *Member, IEEE*

***Abstract*—Automatic Speech Recognition (ASR) has historically been a driving force behind many machine learning (ML) techniques, including the ubiquitously used hidden Markov model, discriminative learning, structured sequence learning, Bayesian learning, and adaptive learning. Moreover, ML can and**

community to make assumptions about a problem, develop precise mathematical theories and algorithms to tackle the problem given those assumptions, but then evaluate on data sets that are relatively small and sometimes synthetic. ASR research, on the

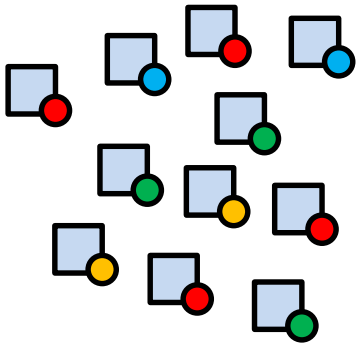


	Input	Output	Decision function	Loss function
Inductive	$\{(x_i, y_i)\}_{i \in TR}$	f^{TS}		
Transductive	$\{(x_i, y_i)\}_{i \in TR}$ and $\{x_i\}_{i \in TS}$	$\{y_i\}_{i \in TS}$		
Generative			Generative models	$L = -\ln p(x, y; \theta)$
Discriminative			Generative *OR* Discriminative models	Discriminative loss (form varies)
Supervised	$\{(x_i, y_i)\}_{i \in TR}$			
Unsupervised	$\{x_i\}_{i \in U}$			
Semi-supervised	$\{(x_i, y_i)\}_{i \in TR}$ and $\{x_i\}_{i \in U}$			
Active	$\{(x_i, y_i)\}_{i \in TR}$ and $\{x_i\}_{i \in U}$ and $\{y_i\}_{i \in L \subseteq U}$			
Adaptive	f^{TR} and $\{(x_i, y_i)\}_{i \in AD}$	f^{TS}		
Multitask	$\{(x_i, y_i)\}_{i \in TR_k}$ for all k	f^k or $\{y_i\}_{i \in TS_k}$ for all k		

Supervised Machine Learning (classification)

Training phase (usually offline)

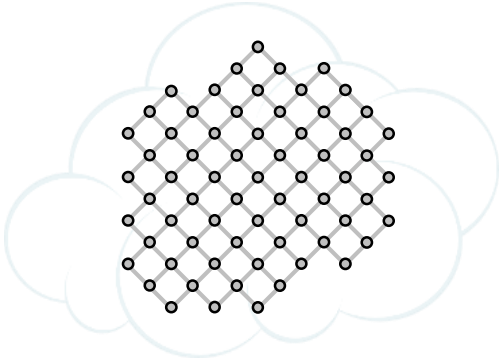
Training data set



measurements (features)
&
associated 'class' labels

(colors used to show class labels)

Learned model

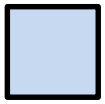


Parameters/weights
(and sometimes structure)

Supervised Machine Learning (classification)

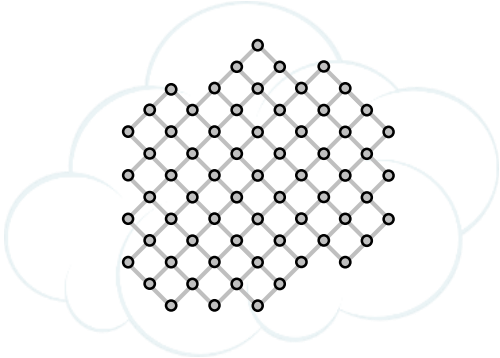
Test phase (run time, online)

Input test data point



measurements (features) only

Learned model

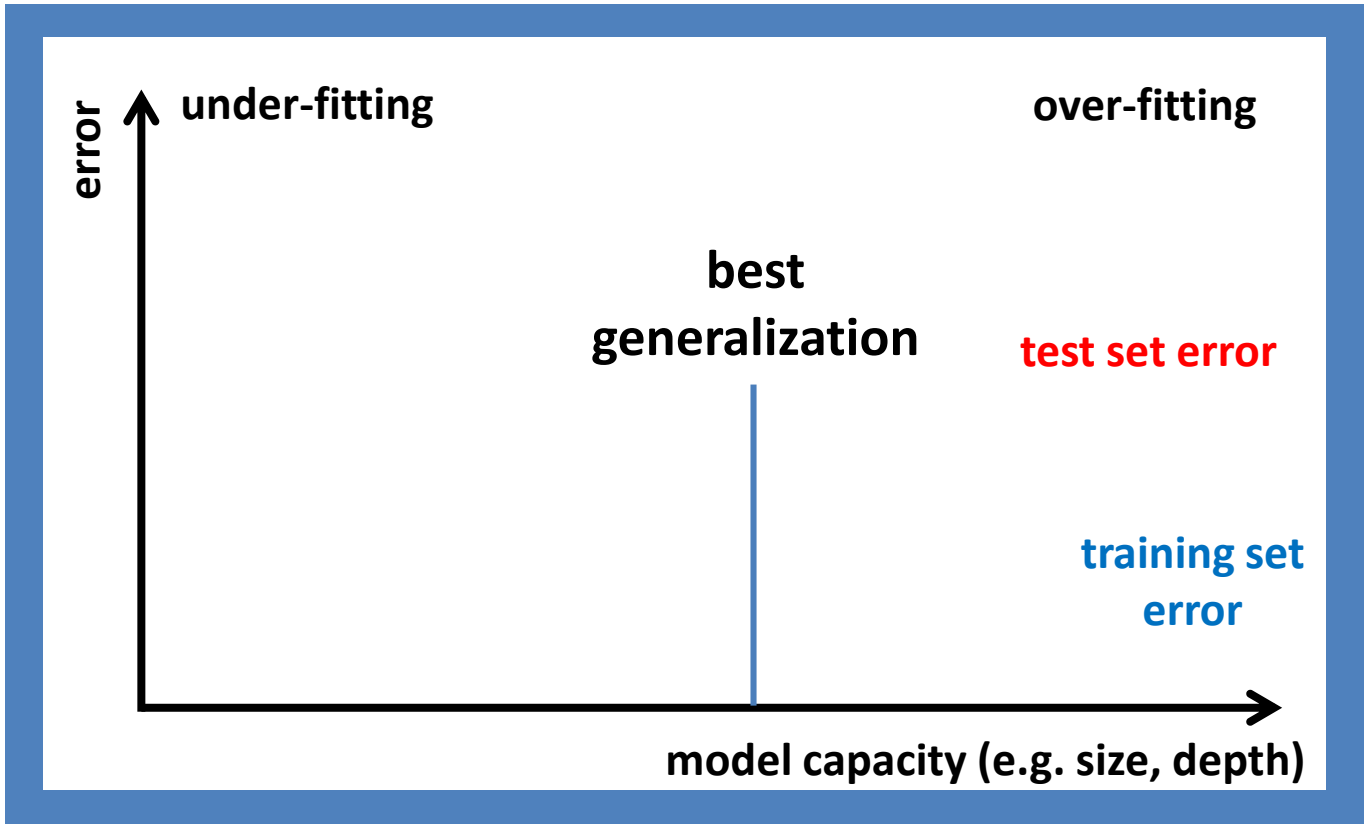


structure + parameters

Output

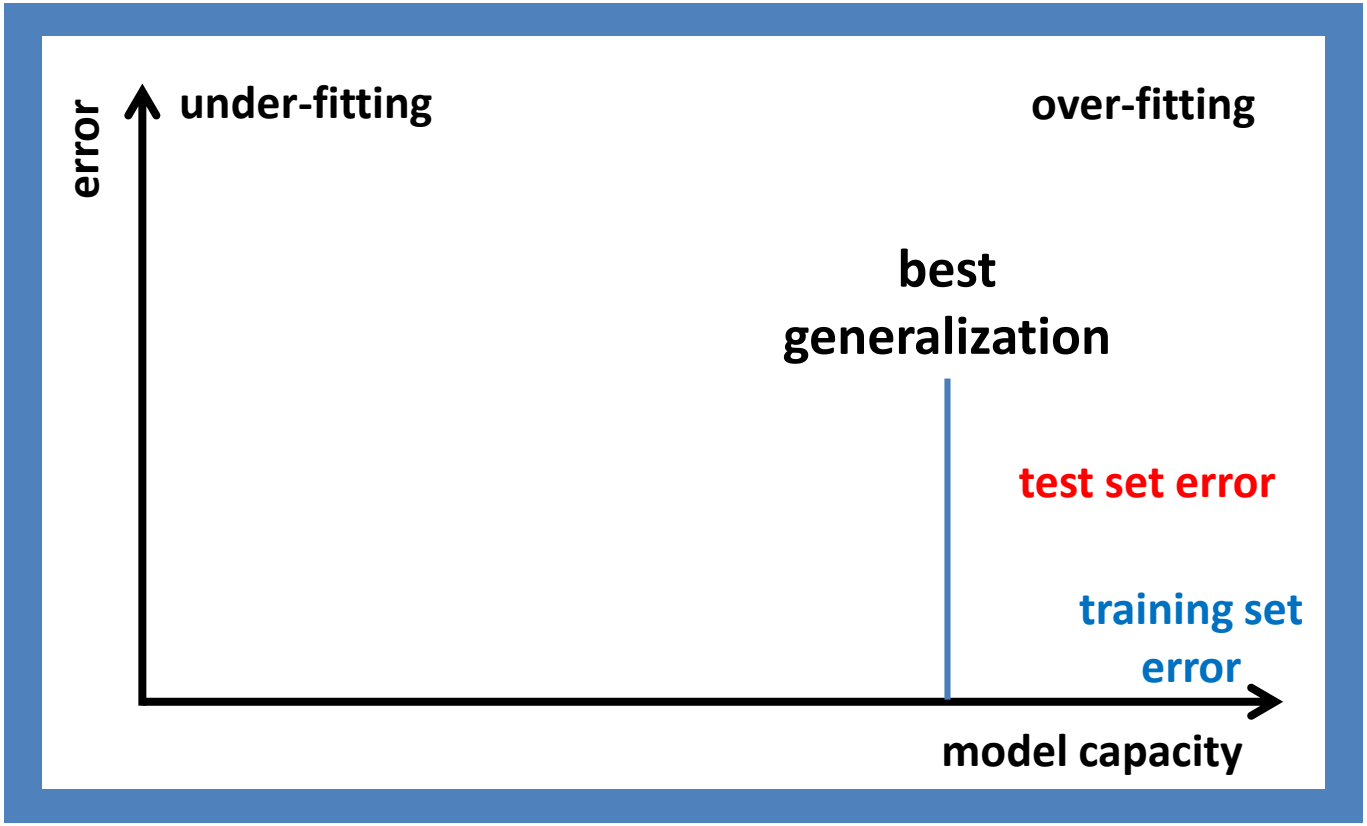
predicted class label or
label sequence (e.g. sentence)

A key ML concept: Generalization



- To avoid over-fitting,
- The need for regularization (or make model simpler, or to add more training)
- → move the training objective away from (empirical) error rate

Generalization – effects of more data



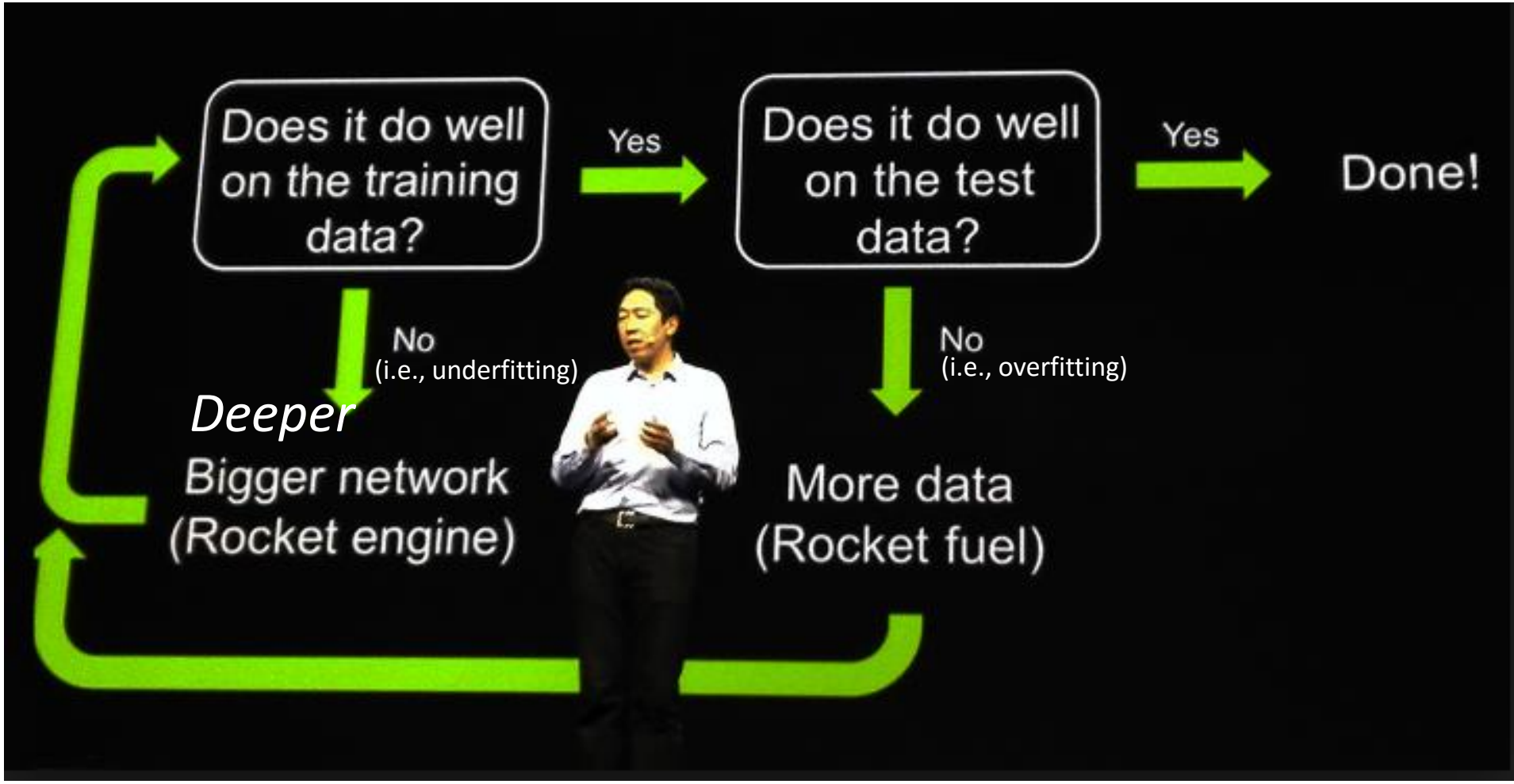
A variety of ML methods

- Decision trees/forests/jungles, Boosting
- Support Vector Machines (SVMs)
- Model-Based (Graphical models, often generative models: sparse connections w. interpretability)
  - model tailored for each new application and incorporates prior knowledge
  - Bayesian statistics exploited to 'invert the model' & infer variables of interest
- Neural Networks (DNN, RNN, dense connections)

These two types of methods can be made DEEP: Deep generative models and DNNs

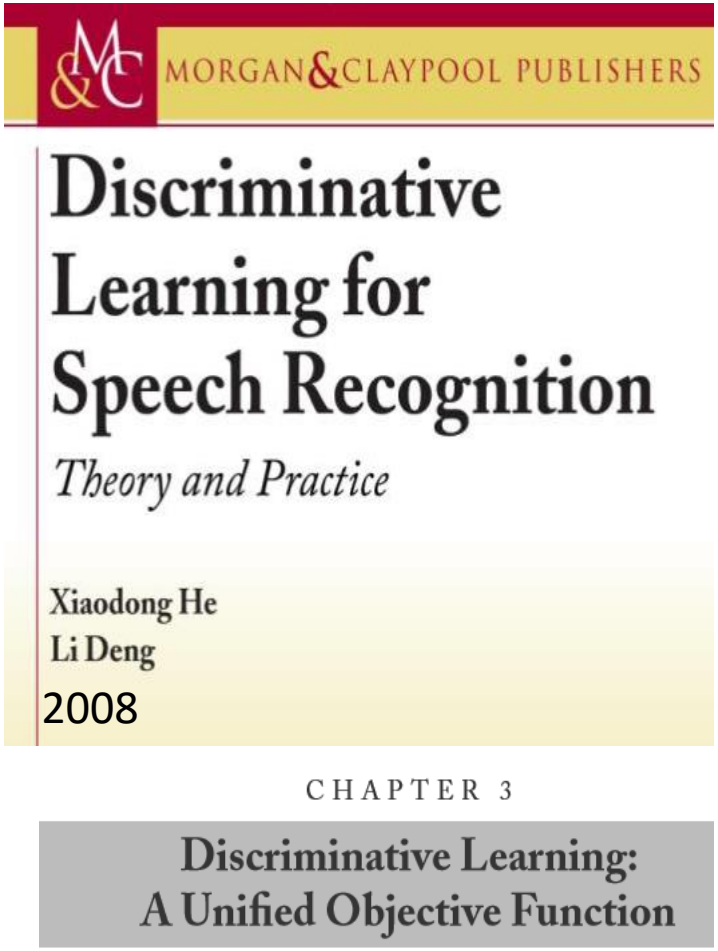
Microsoft Research

Recipe for (Supervised) Deep Learning with Big Data

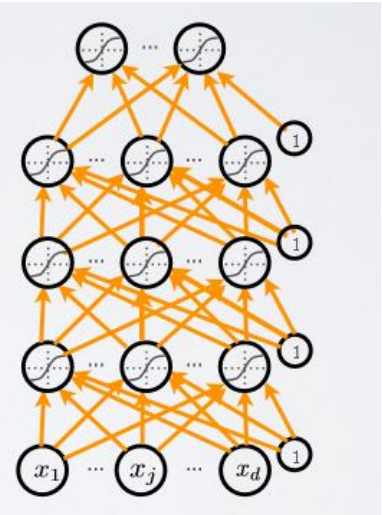


Contrast with Signal Processing Approaches

- Strong focus on sophisticated objective functions for optimization (e.g. MCE, MWE, MPE, MMI, string-level, super-string-level, …)
- Can be regarded as "end2end" learning in ASR
- Almost always non-convex optimization (praised by deep-ML researchers)
- Weaker focus on regularization & overfitting
- Why?
- Our ASR community has been using shallow, low-capacity models for too long (e.g., GMM-HMM)
- Less need for overcoming overfitting
- Now, deep models add a new dimension for increasing model capacity
- Regularization becomes essential for DNN
- E.g. DBN pre-training, "**dropout" method**, STDP spiking neurons, etc.



Deep Neural Net (DNN) Basics
--- why gradient vanishes & how to rescue it



(Shallow) Neural Networks for ASR

(prior to the rise of deep learning)

Temporal & Time-Delay (1-D Convolutional) Neural Nets

- Atlas, Homma, and Marks, "An Artificial Neural Network for Spatio-Temporal Bipolar Patterns, Application to Phoneme Classification," NIPS, 1988. — 1988
- Waibel, Hanazawa, Hinton, Shikano, Lang. "Phoneme recognition using time-delay neural networks." IEEE Transactions on Acoustics, Speech and Signal Processing, 1989. — 1989

Hybrid Neural Nets-HMM

- Morgan and Bourlard. "Continuous speech recognition using MLP with HMMs," ICASSP, 1990. — 1990

Recurrent Neural Nets

- Bengio. "Artificial Neural Networks and their Application to Speech/Sequence Recognition", Ph.D. thesis, 1991. — 1991
- Robinson. "A real-time recurrent error propagation network word recognition system," ICASSP 1992. — 1992

Neural-Net Nonlinear Prediction

- Deng, Hassanein, Elmasry. "Analysis of correlation structure for a neural predictive model with applications to speech recognition," *Neural Networks*, vol. 7, No. 2, 1994. — 1994

Bidirectional Recurrent Neural Nets

- Schuster, Paliwal. "Bidirectional recurrent neural networks," IEEE Trans. Signal Processing, 1997. — 1997

Neural-Net TANDEM

- Hermansky, Ellis, Sharma. "Tandem connectionist feature extraction for conventional HMM systems." ICASSP 2000. — 2000
- Morgan, Zhu, Stolcke, Sonmez, Sivadas, Shinozaki, Ostendorf, Jain, Hermansky, Ellis, Doddington, Chen, Cretin, Bourlard, Athineos, "Pushing the envelope - aside [speech recognition]," IEEE Signal Processing Magazine, vol. 22, no. 5, 2005. — 2005

← **DARPA EARS Program 2001-2004: Novel Approach I** (Novel Approach II: Deep Generative Model)

Bottle-neck Features Extracted from Neural-Nets

- Grezl, Karafiat, Kontar & Cernocky. "Probabilistic and bottle-neck features for LVCSR of meetings," ICASSP, 2007. — 2007

One-hidden-layer neural networks

- Starting from (nonlinear) regression

$$y(\mathbf{x}, \mathbf{w}) = f\left(\sum_{j=1}^{M} w_j \phi_j(\mathbf{x})\right)$$

- Replace each ϕ_j with a variable z_j, where

$$z_j = h(a_j) \qquad a_j = \sum_{i=1}^{D} w_{ji}^{(1)} x_i + w_{j0}^{(1)}$$

and $h()$ is a fixed activation function

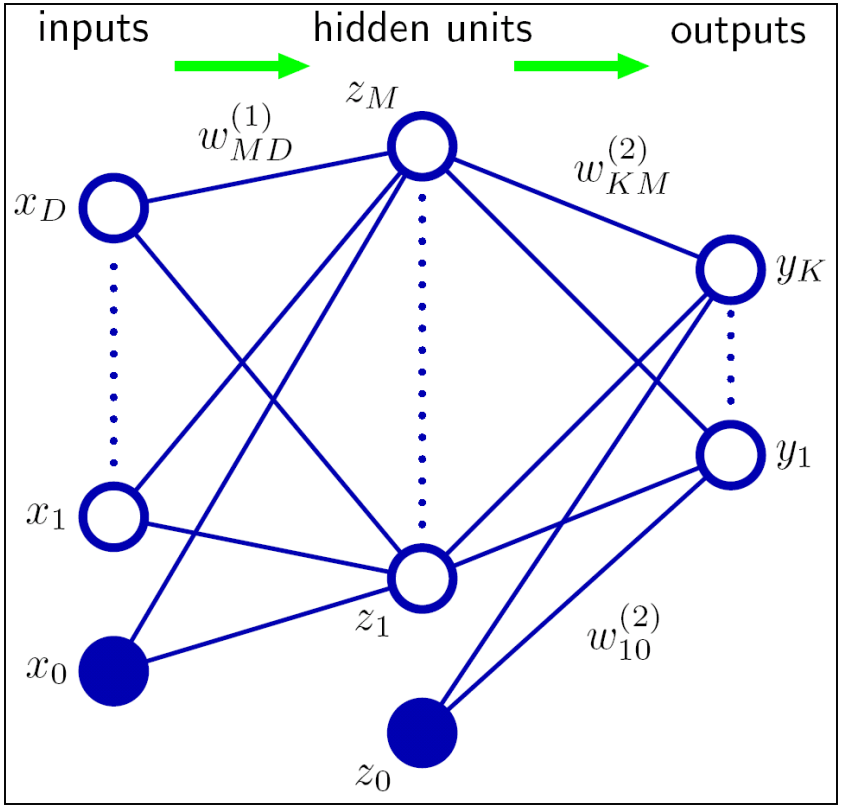
- The outputs obtained from

$$y_k = \sigma(a_k) \quad a_k = \sum_{j=1}^{M} w_{kj}^{(2)} z_j + w_{k0}^{(2)}$$

where $\sigma()$ is another fixed function

- In all, we have (simplifying biases):

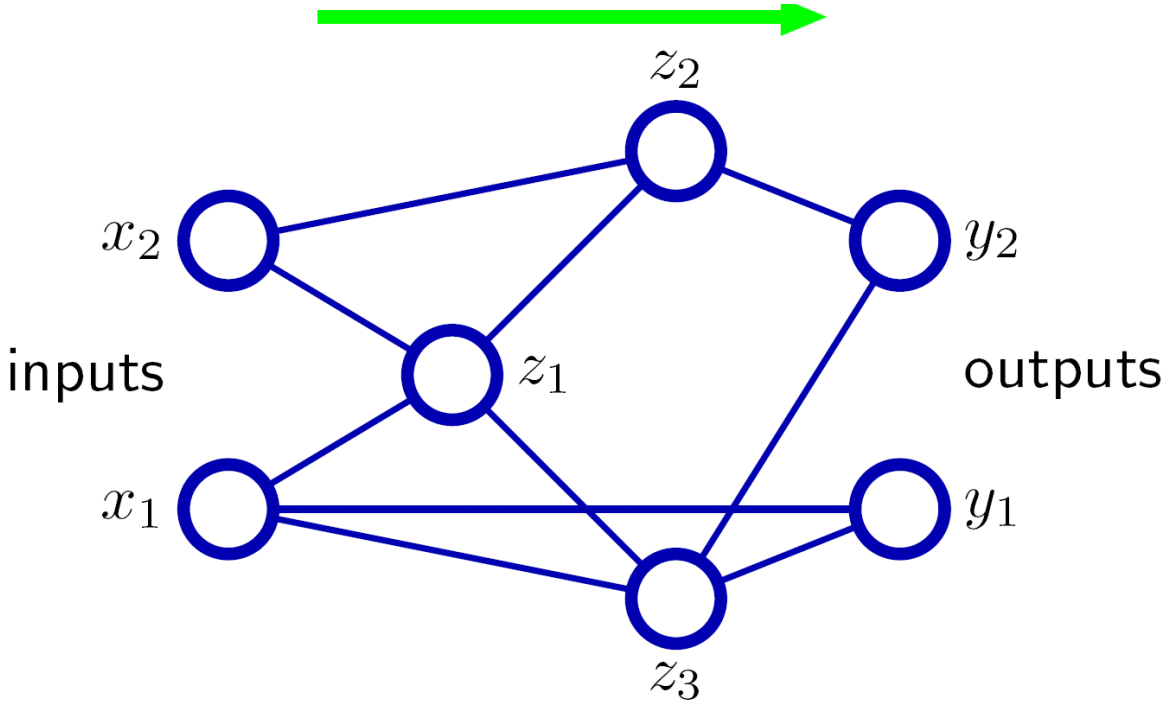
$$y_k(\mathbf{x}, \mathbf{w}) = \sigma\left(\sum_{j=0}^{M} w_{kj}^{(2)} h\left(\sum_{i=0}^{D} w_{ji}^{(1)} x_i\right)\right)$$



Multi-layer neural networks

Denote all activation functions by *h*

$$z_k = h\left(\sum_j w_{kj} z_j\right)$$



- The sum is over those values of *j* with instantiated weights w_{kj}

Unchallenged learning algorithm: Back propagation (BP)

- For regression, we consider a squared error cost function:

$$E(\mathbf{w}) = \tfrac{1}{2} \sum_n \sum_k (t_{nk} - y_k(\mathbf{x}_n, \mathbf{w}))^2$$

  which corresponds to a Gaussian density $p(\mathbf{t}|\mathbf{x})$

- We can substitute

$$y_k(\mathbf{x}, \mathbf{w}) = \sigma \left(\sum_{j=0}^{M} w_{kj}^{(2)} h \left(\sum_{i=0}^{D} w_{ji}^{(1)} x_i \right) \right)$$

  and use a general purpose optimizer to estimate $\mathbf{w}$, but it is much more efficient to exploit derivatives of E, the essence of BP

Learning neural networks

$$E(\mathbf{w}) = \tfrac{1}{2} \sum_n \sum_k (t_{nk} - y_k(\mathbf{x}_n, \mathbf{w}))^2$$

- Recall that for linear regression:

$$\partial E(\mathbf{w})/\partial w_m = -\sum_n (t_n - y_n) x_{nm}$$

Weight in-between error signal and input signal

Error signal

Input signal

- We'll use the chain rule of differentiation to derive a similar-looking expression, where
  - Local input signals are forward-propagated from the input
  - Local error signals are back-propagated from the output

Local signals needed for learning

- For clarity, consider the error for one training case:

$$E_n = \frac{1}{2}\sum_k (t_{nk} - y_{nk})^2$$

- To compute $\partial E_n / \partial w_{ji}$, note that w_{ji} appears in only one term of the overall expression, namely

$$a_j = \sum_i w_{ji} z_i$$

if w_{ji} is in the 1st layer, z_i is actually input x_i

- Using the chain rule of differentiation, we have

$$\frac{\partial E_n}{\partial w_{ji}} = \frac{\partial E_n}{\partial a_j}\frac{\partial a_j}{\partial w_{ji}} = \delta_j z_i$$

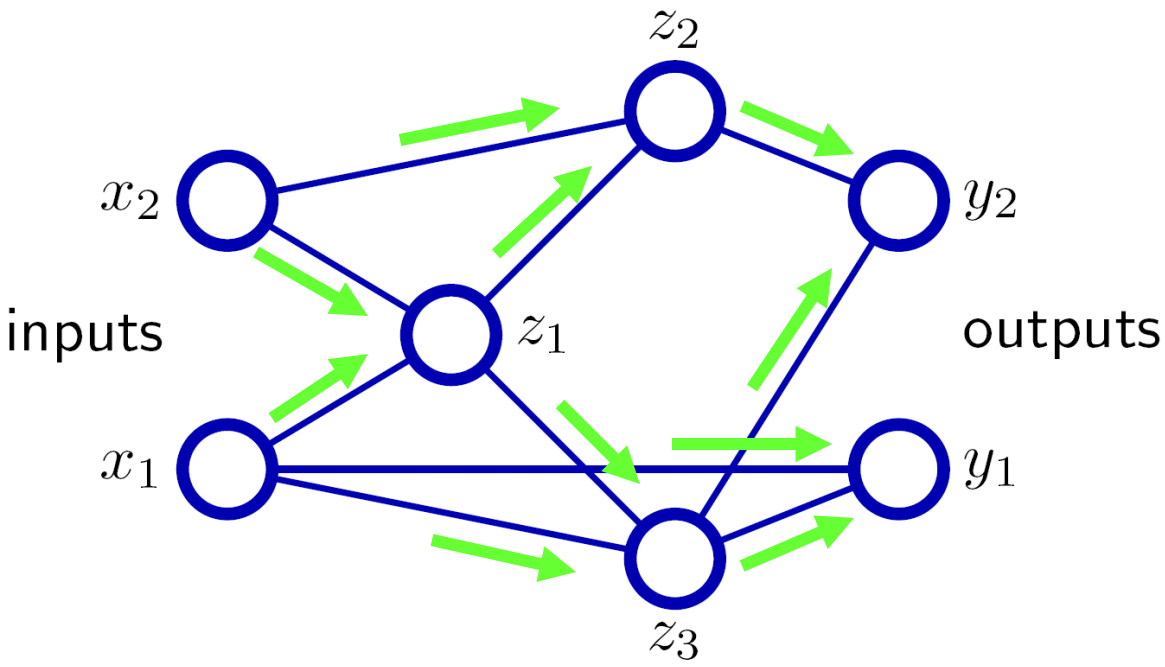
Weight

Local error signal

Local input signal

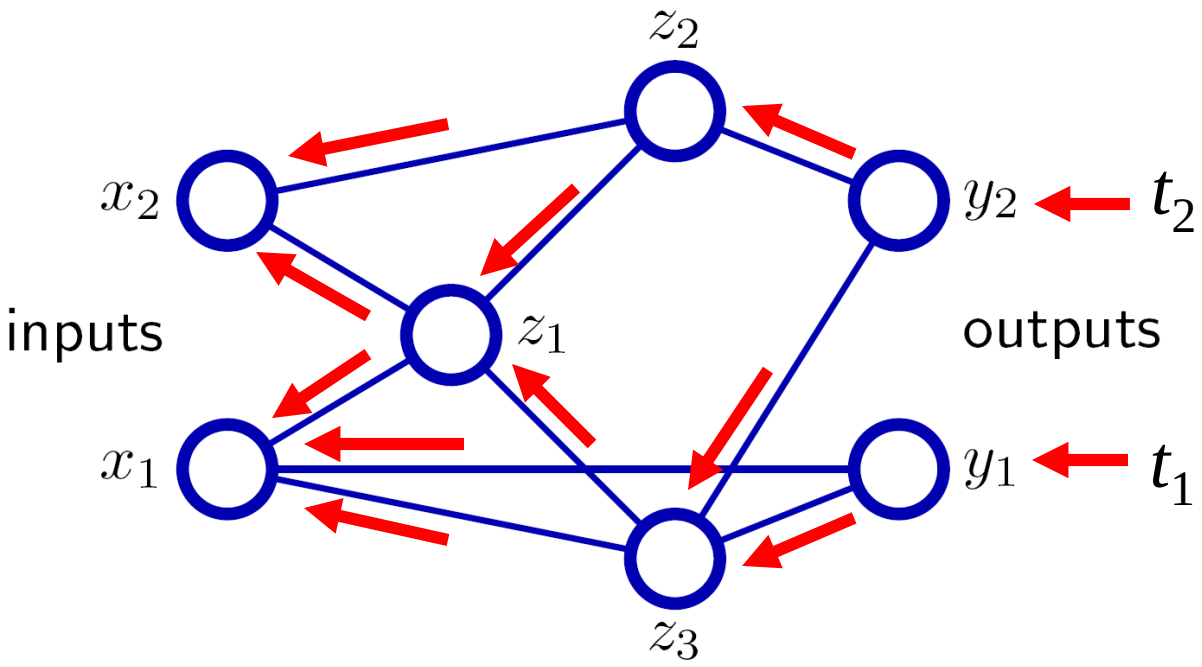
where $\delta_j \equiv \dfrac{\partial E_n}{\partial a_j}$

Forward-propagating local input signals



- Forward propagation gives all the a's and z's

Back-propagating local error signals



- Back-propagation gives all the δ's

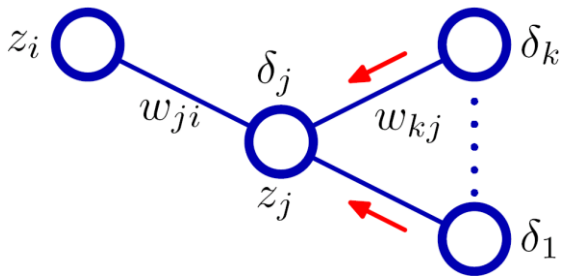
Back-propagating error signals

- To compute $\partial E_n/\partial a_j$ (i.e., δ_j), a_j (also called "logit") appears in all those expressions $a_k = \sum_i w_{ki}\, h(a_i)$ that depend on a_j
- Using the chain rule, we have

$$\delta_j \equiv \frac{\partial E_n}{\partial a_j} = \sum_k \frac{\partial E_n}{\partial a_k}\frac{\partial a_k}{\partial a_j}$$

- The sum is over k s.t. unit j is connected to unit k and for each such term, $\partial a_k/\partial a_j = w_{kj}\, h'(a_j)$
- Noting that $\partial E_n/\partial a_k = \delta_k$, we get the back-propagation rule:

$$\delta_j = h'(a_j)\sum_k w_{kj}\delta_k$$



- For output units: $\delta_j = -h'(a_j)(t_{nk} - y_{nk})$

Putting the propagations together

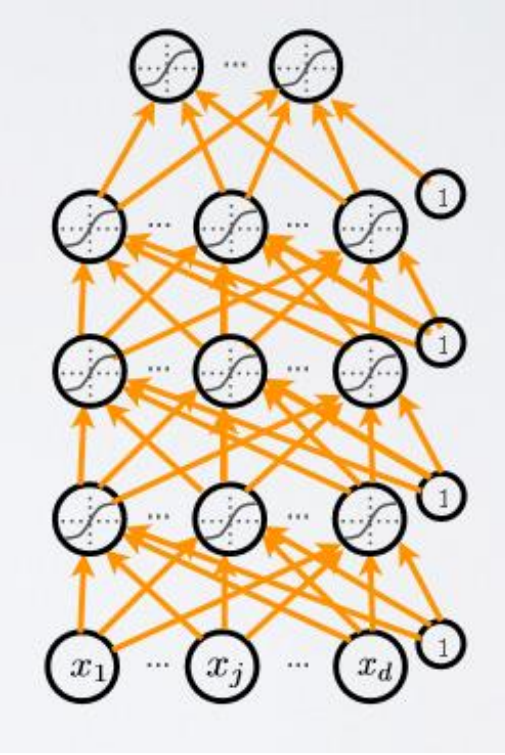
- For each training case *n*, apply forward propagation and back-propagation to compute

$$\frac{\partial E_n}{\partial w_{ji}} = \delta_j z_i$$

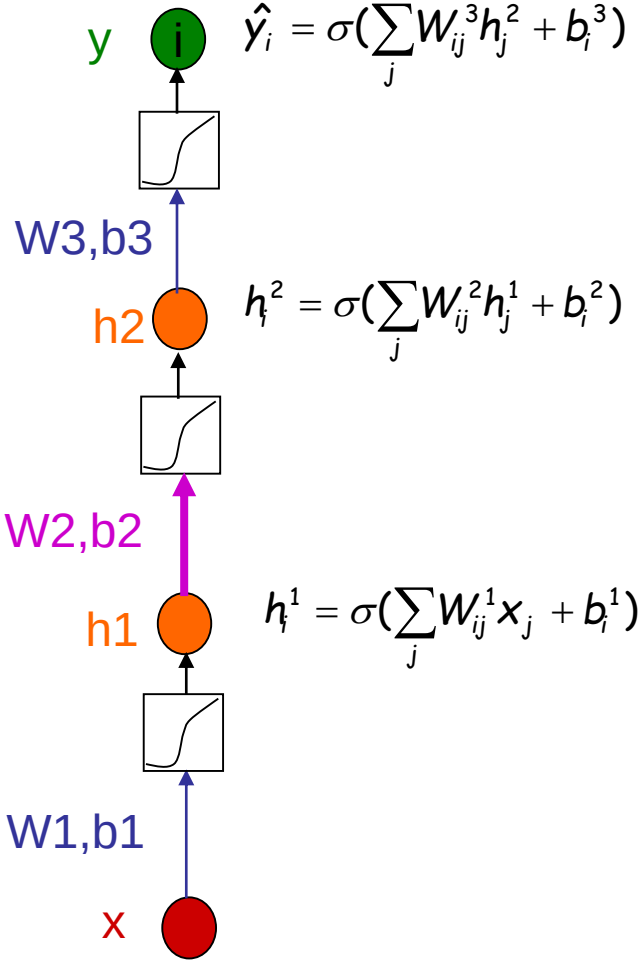
  for each weight w_{ji}
- Sum these over training cases to compute $\frac{\partial E}{\partial w_{ji}}$
- Use these derivatives for steepest descent learning (too slow for large set of training data)
- Minibatch learning: After a small set of input samples, use the above gradient to update the weights (so update more often)

Why gradients tend to vanish for DNN

- Recall BP for adjacent layer pair: $\delta_j = h'(a_j) \sum_k w_{kj} \delta_k$
- For sigmoid units: $h'(a_j) = h(a_j)\,(1-h(a_j))$
- If there is one hidden layer, gradients are not likely to vanish
- Problem becomes serious when nets get deep



Why gradients tend to vanish for DNN

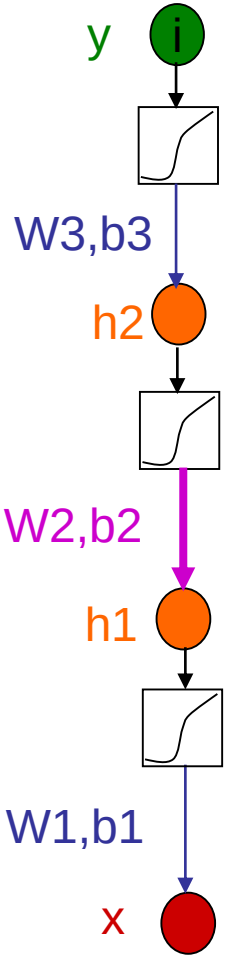


$$\hat{y}_i = \sigma\left(\sum_j W_{ij}^3 h_j^2 + b_i^3\right)$$

$$h_i^2 = \sigma\left(\sum_j W_{ij}^2 h_j^1 + b_i^2\right)$$

$$h_i^1 = \sigma\left(\sum_j W_{ij}^1 x_j + b_i^1\right)$$

Upward pass



$$\delta_{in}^3 = \hat{y}_{in}(1-\hat{y}_{in})\frac{d\,error_{in}}{d\,\sigma_{in}}$$

$$\delta_{jn}^2 = h_{jn}^2(1-h_{jn}^2)\sum_{upstream\ i} W_{ij}^3\delta_{in}^3$$

$$\delta_{kn}^1 = h_{kn}^1(1-h_{kn}^1)\sum_{upstream\ j} W_{jk}^2\delta_{jn}^2$$

downward pass

Why gradients tend to vanish for DNN

- To illustrate to problem, let's use matrix form of error BP:

$$\delta^l = ((w^{l+1})^T \delta^{l+1}) \odot \sigma'(z^l)$$

$$= ((w^{l+1})^T \left[((w^{l+2})^T \delta^{l+2}) \odot \sigma'(z^{l+1}) \right]) \odot \sigma'(z^l)$$

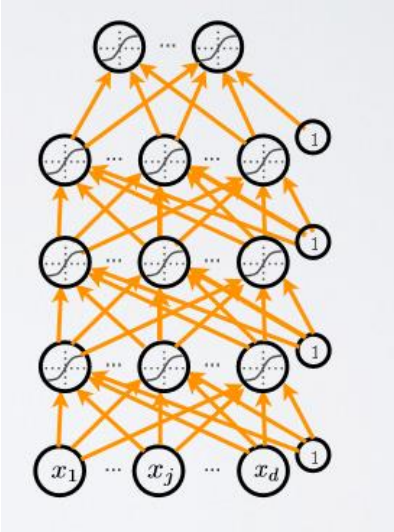
..., ..., ...

..., ..., ...

- So even if forward pass is nonlinear, **error backprop is a linear process**
- It suffers from all problems associated with linear processes
- Many terms of $\sigma(1-\sigma)$ for sigmoid units
- In addition, many terms in the product of W's
- If any sigmoid unit saturates in either direction, the error gradient becomes zero
- If ||W||<1, the product will shrink fast for high depths
- If ||W||>1, the product may grow fast for high depths

How to rescue it

- Pre-train the DNN by generative DBN (solution by 2010)
  - Complicated process
- Discriminative pre-training (much easier to do)
- Still random initialization but with carefully set variance values; e.g.
  - Layer-dependent variance values
  - For lower layers (with more terms in the product), make the variance closer to 0 (e.g. < 0.1)
- Use **bigger training data** to reduce the chance of vanishing gradients for each epoch
- Use ReLU units: only one side of zero gradient instead of two as for sigmoid units



The power of understanding root causes!!!
(mid 2010 at MSR Redmond)

An alternative way of training NN

- Backprop takes partial derivatives: $\frac{\partial E_n}{\partial w_{ji}} = \delta_j z_i$

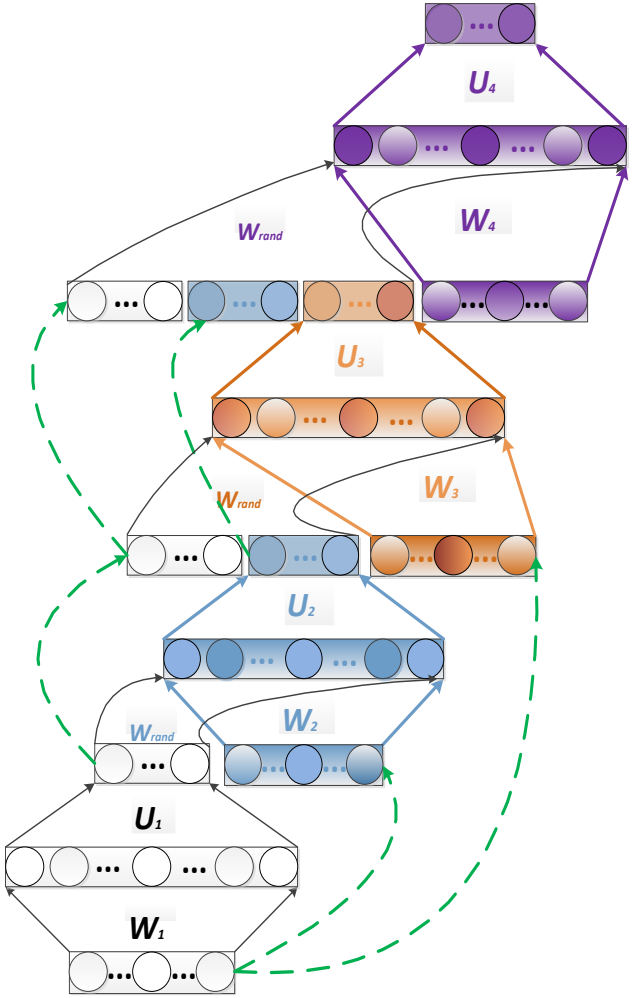
- If the **output layer is linear**, total (instead of partial) derivative can be computed

$$\frac{dE}{d\mathbf{W}} = \frac{\partial E}{\partial \mathbf{W}} + \frac{\partial E}{\partial \mathbf{U}^*}\frac{\partial \mathbf{U}^*}{\partial \mathbf{W}}.$$

- This is basic learning method for Deep Convex/Stacking Net (DSN), designed to be easily parallelizable by batch training
- Using the total derivative is equivalent to coordinate descent algorithm with an "infinite" step size to achieve the global optimum along the "coordinate" of updating U while fixing W.

Deep Stacking Nets

- Learn weight matrices U and W in individual modules separately.
- Given W and linear output layer, U can be expressed as explicit nonlinear function of W.
- This nonlinear function is used as the constraint in solving nonlinear least square for learning W.
- Initializing W with RBM (bottom layer)
- For higher layers, part of W is initialized with the optimized W from the immediately lower layer and part of it with random numbers



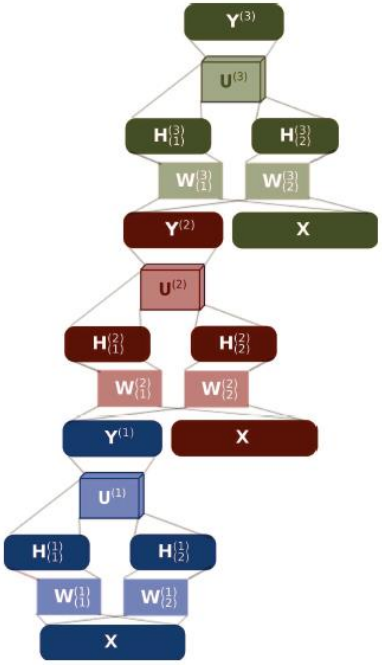


Fig. 1. An example T-DSN architecture with three stacking blocks, where each block consists of three layers, and

(Deng, Yu, Platt, ICASSP-2012; Hutchinson, Deng, Yu, IEEE T-PAMI, 2013)

Microsoft Research

A neat way of learning DSN weights

$$\mathrm{E} = \frac{1}{2}\sum_n ||\boldsymbol{y}_n - \boldsymbol{t}_n||^2, \qquad \text{where } \boldsymbol{y}_n = \boldsymbol{U}^T\boldsymbol{h}_n = \boldsymbol{U}^T\sigma(\boldsymbol{W}^T\boldsymbol{x}_n) = G_n(\boldsymbol{U},\boldsymbol{W})$$

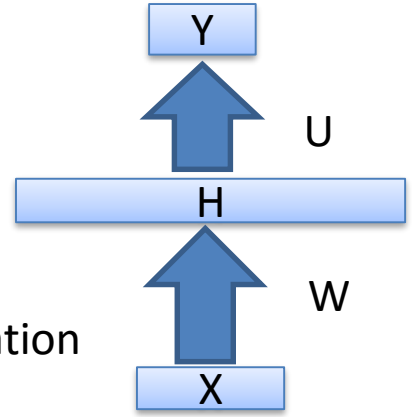
$$\frac{\partial E}{\partial \boldsymbol{U}} = 2\boldsymbol{H}(\boldsymbol{U}^T\boldsymbol{H} - \boldsymbol{T})^T \rightarrow \quad \boldsymbol{U} = \left(\boldsymbol{H}\boldsymbol{H}^{\boldsymbol{T}}\right)^{-1}\boldsymbol{H}\boldsymbol{T}^T = \mathrm{F}(\boldsymbol{W}), \text{ where } \boldsymbol{h}_n = \sigma(\boldsymbol{W}^T\boldsymbol{x}_n)$$

$$\mathrm{E} = \tfrac{1}{2}\textstyle\sum_n ||G_n(\boldsymbol{U},\boldsymbol{W}) - \boldsymbol{t}_n||^2, \text{ subject to } \boldsymbol{U} = \mathrm{F}(\boldsymbol{W}),$$

Use of Lagrange multiplier method:

$$\mathrm{E} = \tfrac{1}{2}\textstyle\sum_n ||G_n(\boldsymbol{U},\boldsymbol{W}) - \boldsymbol{t}_n||^2 + \lambda\, ||\boldsymbol{U} - \mathrm{F}(\boldsymbol{W})||$$

to learn ***W*** (& then ***U***) → full derivation $\frac{dE}{d\mathbf{W}}$ in closed form
(i.e. no longer recursion on partial derivation as in backpropagation



- Advantages found:
  --- less noise in gradient than using chain rule which ignores explicit constraint $\boldsymbol{U} = \mathrm{F}(\boldsymbol{W})$
  --- batch learning is effective, aiding parallel training

Microsoft Research

How the Brain May Do BackProp

- Canadian Psychology, Vol 44, pp 10-13, 2003.

The Ups and Downs of Hebb Synapses

GEOFFREY HINTON
University of Toronto

Abstract
Modelers have come up with many different learning rules for neural networks. When a teacher specifies the correct

This approach led to effective "error-driven" learning rules such as the Widrow-Hoff rule (Widrow & Hoff, 1960) and the perceptron convergence procedure (Rosenblatt, 1961) and it was later generalized to multilayer networks by using backpropagation of the errors to get training signals for intermediate "hidden" layers (Rumelhart, Hinton, & Williams, 1986).

- Feedback system in biological neural nets
- Key roles of STDP (Spike-Time-Dependent Plasticity) --- temporal derivative
- To provide a way to encode error derivatives

Microsoft Research

How the Brain May Do BackProp

- Backprop algorithm requires that feedforward and feedback weights are the same
- This is clearly not true for biological neural nets
- How to reconcile this discrepancy?
- Recent studies showed that use of random feedback weights in BP performs close to rigorous BP
- Implications for regularizing BP learning (like dropout, which may not make sense at 1st glance)

Random feedback weights support learning in deep neural networks

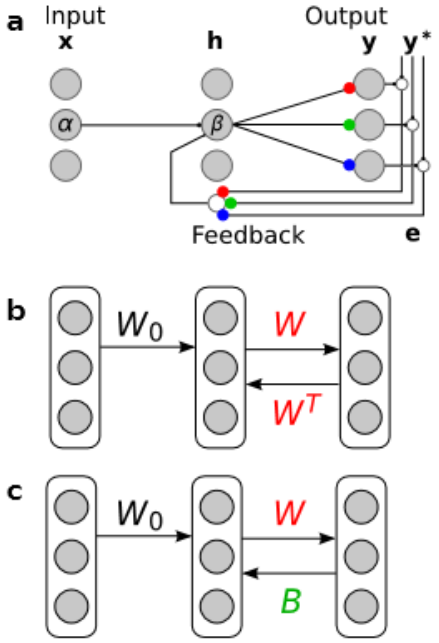
Timothy P. Lillicrap[1*], Daniel Cownden[2], Douglas B. Tweed[3,4], Colin J. Akerman[1]

[1]Department of Pharmacology, University of Oxford, Oxford, United Kingdom
[2]Centre for the Study of Cultural Evolution, Stockholm University, Stockholm, Sweden
[3]Departments of Physiology and Medicine, University of Toronto, Toronto, Canada
[4]Centre for Vision Research, York University, Toronto, Canada

*To whom correspondence should be addressed:
timothy.lillicrap@pharm.ox.ac.uk
colin.akerman@pharm.ox.ac.uk

Abstract

The brain processes information through many layers of neurons. This deep architecture is representationally powerful[1,2,3,4], but it complicates learning by making it hard to identify the responsible neurons when a mistake is made[1,5]. In machine learning, the backpropagation algorithm[1] assigns blame to a neu



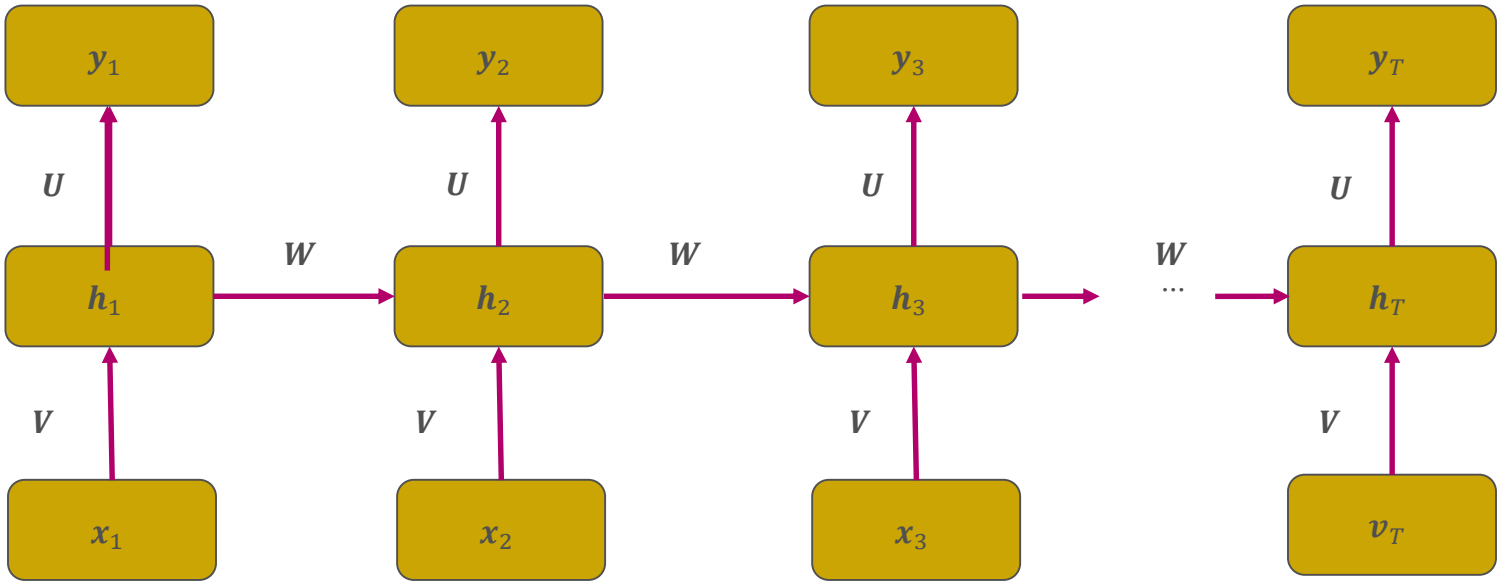
Recurrent Neural Net (RNN) Basics

--- why memory decays fast or explode (1990's)

--- how to rescue it (2010's, in the new deep learning era)

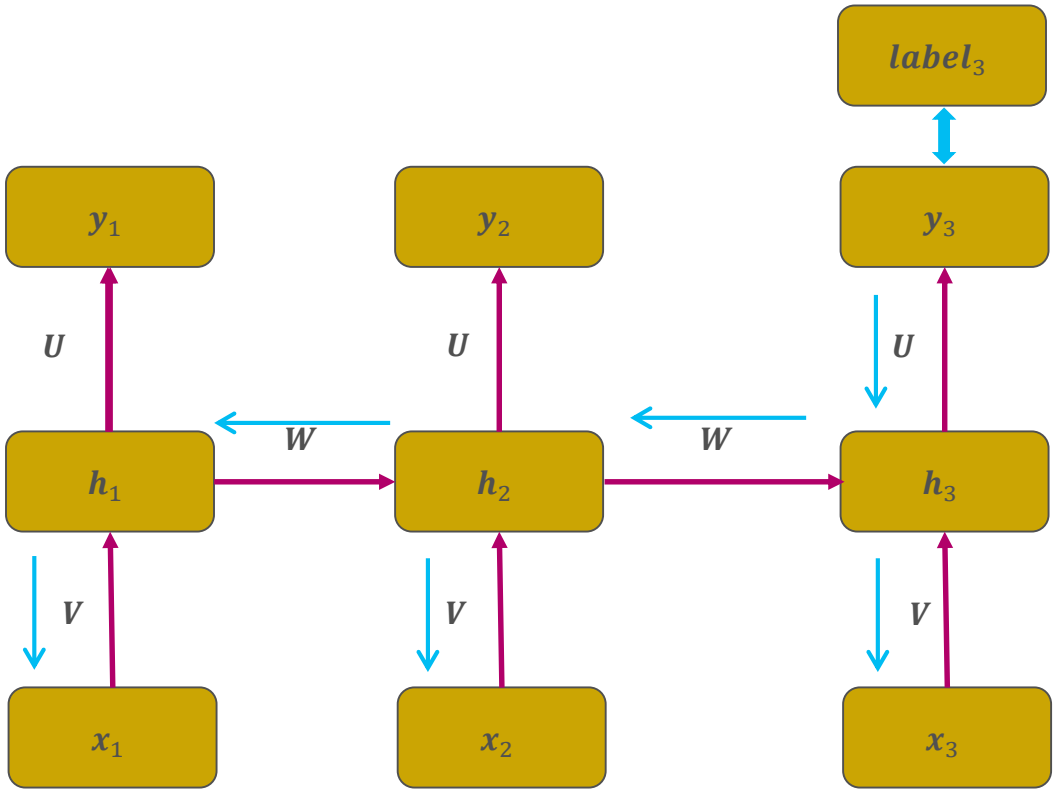
Basic architecture of an RNN

h_t is the hidden layer that carries the information from time $0 \sim t$
where x_t: *the input word* , y_t: *the output tag*
$y_t = SoftMax(U \cdot h_t), where\ h_t = \sigma(W \cdot h_{t-1} + V \cdot x_t)$



Used for slot filling in SLU [Mesnil, He, Deng, Bengio, 2013; Yao, Zweig, Hwang, Shi, Yu, 2013] IEEE TASLP

Back-propagation through time (BPTT)



at time $t = 3$

1. Forward propagation
2. Generate output
3. Calculate error
4. Back propagation
5. Back prop. through time

A Good Read to appreciate the "Magic" of modern RNNs

The Unreasonable Effectiveness of Recurrent Neural Networks

May 21, 2015

There's something magical about Recurrent Neural Networks (RNNs). I still remember when I trained my first recurrent network for Image Captioning. Within a few dozen minutes of training my first baby model (with rather arbitrarily-chosen hyperparameters) started to generate very nice looking descriptions of images that were on the edge of making sense. Sometimes the ratio of how simple your model is to the quality of the results you get out of it blows past your expectations, and this was one of those times. What made this result so shocking at the time was that the common wisdom was that RNNs were supposed to be difficult to train (with more experience I've in fact reached the opposite conclusion). Fast forward about a year: I'm training RNNs all the time and I've witnessed their power and robustness many times, and yet their magical outputs still find ways of amusing me. This post is about sharing some of that magic with you.

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

Some early attempts to examine difficulties in learning RNNs

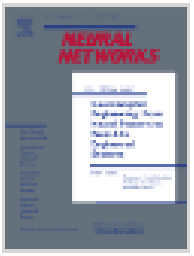
- Yoshua Bengio's Ph.D. thesis at McGill University (1991)
- Based on attractor properties of nonlinear dynamic systems
- His recent, more intuitive explanation --- in extreme of nonlinearity, discrete functions and gradients vanish or explode:

- An alternative analysis: based on perturbation analysis of nonlinear differential equations (next several slides)

ELSEVIER

Neural Networks

Volume 7, Issue 2, 1994, Pages 331–339



Contributed article

Analysis of the correlation structure for a neural predictive model with application to speech recognition ☆

L. Deng, K. Hassanein, M. Elmasry

University of Waterloo Canada

- NN for nonlinear sequence prediction (like NN language model used today)
- Memory (temporal correlation) proved to be stronger than linear prediction
- No GPUs to use; very slow to train with BP; did not make NN big and deep, etc.

- Conceptually easy to make it deep using (state-space) signal processing & graphical models.. by moving away from NN…

The method we use to derive the correlation function for (2) resembles the perturbation analysis for the study of nonlinear differential equations [13]. To proceed, we construct a family of models which is parameterized by α:

$$Y_{t+1}(\alpha) = \alpha f(Y_t(\alpha)) + \epsilon_{t+1}, \tag{3}$$

Once the model is parameterized, the autoregression on the data Y_t can be removed by performing power-series expansion of the nonlinear function $f(\cdot)$:

$$\begin{aligned}
Y_1(\alpha) &= \epsilon_1 + \alpha f(Y_0(\alpha)), \\
Y_2(\alpha) &= \epsilon_2 + \alpha f(Y_1(\alpha)) \\
&= \epsilon_2 + \alpha f(\epsilon_1) + \alpha^2 f(Y_0) f'(\epsilon_1) + \frac{1}{2}\alpha^3 f^2(Y_0) f''(\epsilon_1) + \cdots, \\
Y_3(\alpha) &= \epsilon_3 + \alpha f(Y_2(\alpha)) \\
&= \epsilon_3 + \alpha f(\epsilon_2) + \alpha^2 f(\epsilon_1) f'(\epsilon_2) + \alpha^3 f(Y_0) f'(\epsilon_1) f'(\epsilon_2) + \cdots, \\
&\vdots
\end{aligned}$$

and in general,

$$Y_t(\alpha) = \epsilon_t + \alpha f(\epsilon_{t-1}) + \alpha^2 f(\epsilon_{t-2}) f'(\epsilon_{t-1}) + \alpha^3 f(\epsilon_{t-3}) f'(\epsilon_{t-2}) f'(\epsilon_{t-1}) + \cdots \quad (4)$$

(In the above, $f'(\cdot)$ denotes the derivative of $f(\cdot)$ with respect to its argument.)

From (4) the covariance function for model (3) is calculated to give

$$\begin{aligned} & Cov[Y_t(\alpha), Y_{t+\tau}(\alpha)] \\ \approx\ & Cov(\epsilon_t, \epsilon_{t+\tau}) + \alpha Cov[f(\epsilon_t), \epsilon_{t+\tau}] + \alpha Cov[\epsilon_t, f(\epsilon_{t+\tau-1})] \\ & + \alpha^2 Cov[f(\epsilon_{t-2}) f'(\epsilon_{t-1}), \epsilon_{t+\tau}] + \alpha^2 Cov[\epsilon_t, f(\epsilon_{t+\tau-2}) f'(\epsilon_{t+\tau-1})] \\ & + \alpha^3 Cov[f(\epsilon_{t-2}) f'(\epsilon_{t-1}), f(\epsilon_{t+\tau-1})] \\ & + \alpha^3 Cov[f(\epsilon_{t-1}), f(\epsilon_{t+\tau-2}) f'(\epsilon_{t+\tau-1})] \\ & + \alpha^4 Cov[f(\epsilon_{t-2}) f'(\epsilon_{t-1}), f(\epsilon_{t+\tau-2}) f'(\epsilon_{t+\tau-1})]. \end{aligned} \quad (5)$$

Among the eight terms in (5), the first, second, fourth, and sixth terms are zero for $\tau \geq 0$. This is due to the IID assumption for ϵ_t and to the fact that $f(\cdot)$ is a static function containing no memory. The fifth term, $Cov[\epsilon_t, f(\epsilon_{t+\tau-2}) f'(\epsilon_{t+\tau-1})]$, is non-zero only for $\tau = 1$ and $\tau = 2$. The seventh and the eighth terms are non-zero only for $\tau = 1$. Likewise, any higher order terms of α in the covariance function which are omitted due to cutoff in the power-series expansion of $Y_t(\alpha)$ would contain non-zero values only for small time lags.

We conclude from the above analysis that prediction of a time series with a single nonlinear term alone does not produce long-term temporal correlations in the model's output.

III.3. Joint prediction with nonlinear and linear terms

In this section we investigate correlation properties of the data generated from the stationary time series model

$$Y_{t+1} = \phi Y_t + f(Y_t) + \epsilon_{t+1}, \qquad t = 1, 2, ..., T, \tag{6}$$

We now decompose the stationary random process $Y_t(\alpha)$ into its stationary component processes by representing it as a power-series expansion on α

$$Y_{t+1}(\alpha) = Y_{t+1,0} + \alpha Y_{t+1,1} + \frac{1}{2!}\alpha^2 Y_{t+1,2} + \frac{1}{3!}\alpha^3 Y_{t+1,3} + \cdots. \tag{8}$$

In order to identify the component processes $Y_{t,i}, i = 0, 1, 2, ...$, we substitute (8) into (7) and approximate the nonlinear function $f(\cdot)$ by truncating its power-series expansion. This gives

$$\begin{aligned} Y_{t+1}(\alpha) \approx\ & \phi(Y_{t,0} + \alpha Y_{t,1} + \frac{1}{2!}\alpha^2 Y_{t,2} + \frac{1}{3!}\alpha^3 Y_{t,3}) \\ & + \alpha[f(Y_{t,0}) + f'(Y_{t,0})(\alpha Y_{t,1} + \frac{1}{2!}\alpha^2 Y_{t,2} + \frac{1}{3!}\alpha^3 Y_{t,3})] + \epsilon_{t+1} \\ =\ & (\phi Y_{t,0} + \epsilon_{t+1}) + \alpha[\phi Y_{t,1} + f(Y_{t,0})] + \alpha^2[\frac{1}{2}\phi Y_{t,2} + f'(Y_{t,0})Y_{t,1}] \\ & + \alpha^3[\frac{1}{6}\phi Y_{t,3} + \frac{1}{2}f'(Y_{t,0})Y_{t,2}] + \cdots. \end{aligned} \tag{9}$$

By equating the coefficients of α^i in (8) and in (9), we obtain the following recursive relations among the component processes $Y_{t,k}, k = 0, 1, 2, ...$:

$$\begin{aligned} Y_{t+1,0} &= \phi Y_{t,0} + \epsilon_{t+1}, \\ Y_{t+1,1} &= \phi Y_{t,1} + f(Y_{t,0}), \\ Y_{t+1,2} &= \phi Y_{t,2} + 2f'(Y_{t,0})Y_{t,1}, \\ Y_{t+1,3} &= \phi Y_{t,3} + 3f'(Y_{t,0})Y_{t,2}, \\ &\vdots \end{aligned} \tag{10}$$

According to (10), we can proceed to derive the autocovariance function for $Y_t(\alpha)$ denoted by

$$\gamma = Cov[Y_t(\alpha), Y_{t+\tau}(\alpha)].$$

Using (8) and truncating the expansion up to the first order, we have

$$\gamma \approx Cov[Y_{t,0} + \alpha Y_{t,1}, Y_{t+\tau,0} + \alpha Y_{t+\tau,1}]. \quad (11)$$

Use of the stationarity property of $Y_{t,0}$ and $Y_{t,1}$ leads to

$$\begin{aligned}\gamma &= \phi^2\gamma + \alpha^2 Cov[f(Y_{t-1,0}), f(Y_{t+\tau-1,0})] \\ &\quad +\phi\alpha Cov[Y_{t-1,0} + \alpha Y_{t-1,1}, f(Y_{t+\tau-1,0})] \\ &\quad +\phi\alpha Cov[Y_{t+\tau-1,0} + \alpha Y_{t+\tau-1,1}, f(Y_{t-1,0})].\end{aligned}$$

Re-arranging terms and using the stationarity property of $Y_{t,0}$ and $Y_{t,1}$ again give γ which is equal to

$$\frac{1}{(1-\phi^2)}\{\alpha^2 Cov[f(Y_{t-1,0}), f(Y_{t+\tau-1,0})] + 2\phi\alpha Cov[Y_{t,0} + \alpha Y_{t,1}, f(Y_{t+\tau,0})]\}. \quad (12)$$

$Y_{t,0}$, the zero-th order expansion of $Y_t(\alpha)$, is a linear process and its properties are well understood (Section III.1). To obtain the desired

$$\begin{aligned}\gamma &= \frac{1}{(1-\phi^2)}\{\alpha^2 Cov[f(Y_{t-1,0}), f(Y_{t+\tau-1,0})] + 2\phi\alpha Cov[Y_{t,0}, f(Y_{t+\tau,0})] \\ &\quad +2\phi\alpha^2 \sum_{i=0}^{t-1} \phi^i Cov[f(Y_{t-i-1,0}), f(Y_{t+\tau,0})]\}.\end{aligned}$$

The first two terms in the above expression are exponentially declining as a function of time lag τ because the component processes involved are just static functions of linear processes. The remaining summation, however, would in general decay more slowly because of the many contributing terms.

Why gradients vanish/explode for RNN

- The easiest account is to follow the analysis for DNN
- except that “depth” of RNN is much larger: the length of input sequence
- Especially serious for speech sequence (not as bad for text input)
- Tony Robinson group was the only one that made RNN work for TIMIT phone recognition (1994)

Microsoft Research

How to rescue it

- Echo state nets ("lazy" approach)
  - avoiding problems by not training input & recurrent weights
  - H. Jaeger. "Short term memory in echo state networks", 2001
- But if you figure out smart ways to train them, you get much better results

Learning Input and Recurrent Weight Matrices in Echo State Networks

Hamid Palangi
University of British Columbia
Vancouver, BC, Canada
hamidp@ece.ubc.ca

Li Deng
Microsoft Research
Redmond, WA, USA
deng@microsoft.com

Rabab K Ward
University of British Columbia
Vancouver, BC, Canada
rababw@ece.ubc.ca

Abstract

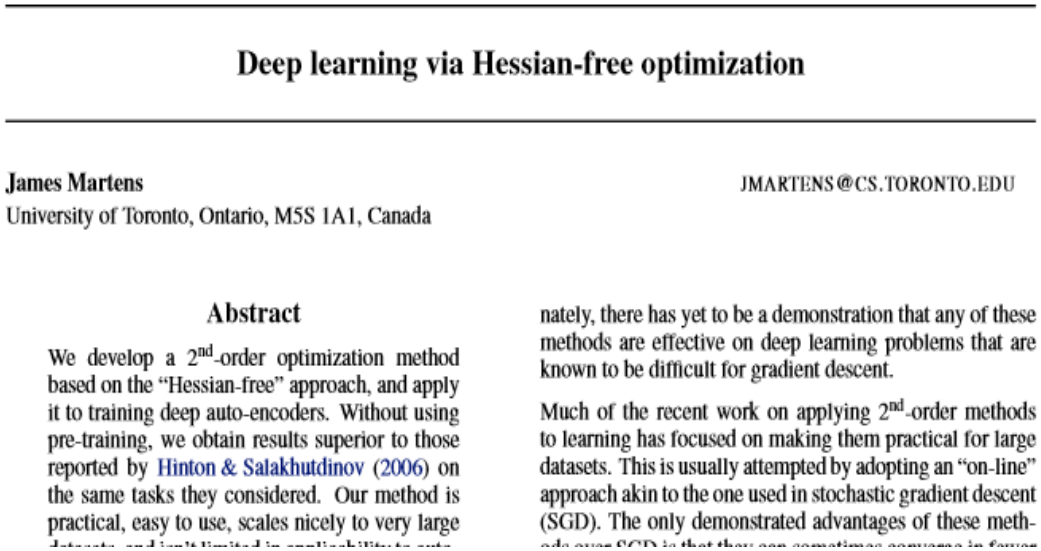
The traditional echo state network (ESN) is a special type of a temporally deep model, the recurrent network (RNN), which carefully designs the recurrent matrix and fixes both the recurrent and input matrices in the RNN. The ESN also adopts the linear output (or readout) units to simplify the leanring of the only output matrix in the RNN. In this paper, we devise a special technique that takes advantage of the linearity in the output units in the ESN to learn the input and

(NIPS-WS, 2013)

Microsoft Research

How to rescue it

- By Better optimization
  - Hessian-free method
  - Primal-dual method



Deep learning via Hessian-free optimization

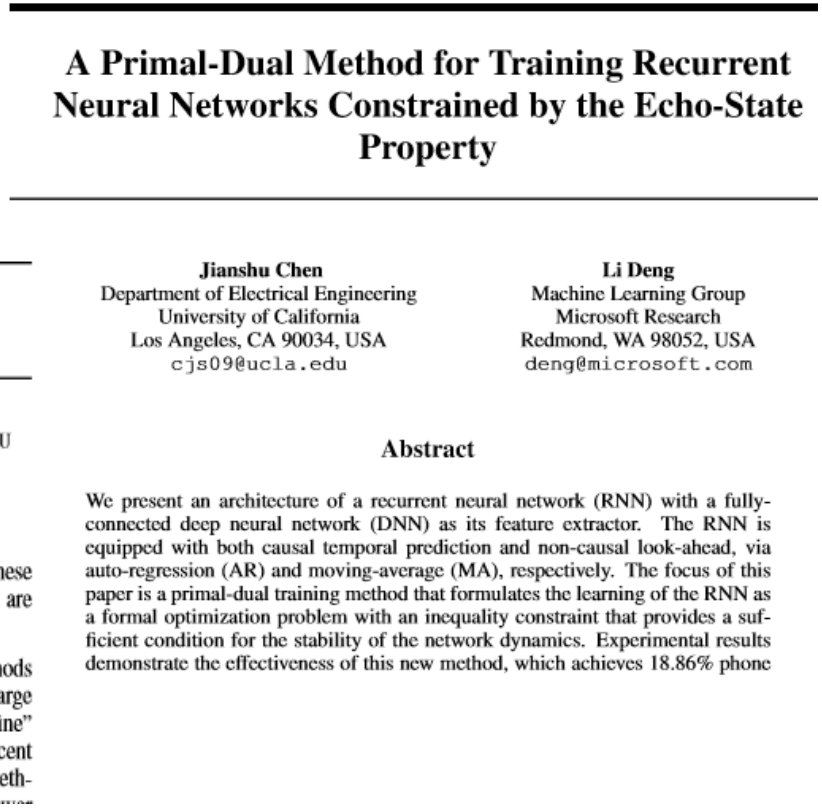
James Martens JMARTENS@CS.TORONTO.EDU
University of Toronto, Ontario, M5S 1A1, Canada

Abstract

We develop a 2nd-order optimization method based on the "Hessian-free" approach, and apply it to training deep auto-encoders. Without using pre-training, we obtain results superior to those reported by Hinton & Salakhutdinov (2006) on the same tasks they considered. Our method is practical, easy to use, scales nicely to very large

nately, there has yet to be a demonstration that any of these methods are effective on deep learning problems that are known to be difficult for gradient descent.

Much of the recent work on applying 2nd-order methods to learning has focused on making them practical for large datasets. This is usually attempted by adopting an "on-line" approach akin to the one used in stochastic gradient descent (SGD). The only demonstrated advantages of these meth-



A Primal-Dual Method for Training Recurrent Neural Networks Constrained by the Echo-State Property

Jianshu Chen
Department of Electrical Engineering
University of California
Los Angeles, CA 90034, USA
cjs09@ucla.edu

Li Deng
Machine Learning Group
Microsoft Research
Redmond, WA 98052, USA
deng@microsoft.com

Abstract

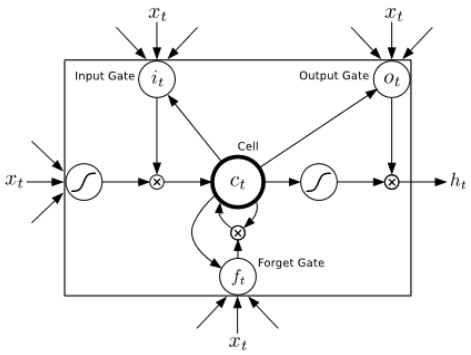
We present an architecture of a recurrent neural network (RNN) with a fully-connected deep neural network (DNN) as its feature extractor. The RNN is equipped with both causal temporal prediction and non-causal look-ahead, via auto-regression (AR) and moving-average (MA), respectively. The focus of this paper is a primal-dual training method that formulates the learning of the RNN as a formal optimization problem with an inequality constraint that provides a sufficient condition for the stability of the network dynamics. Experimental results demonstrate the effectiveness of this new method, which achieves 18.86% phone

How to rescue it

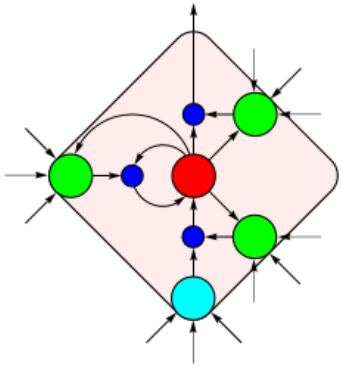
- Use of LSTM (long short-term memory) cells
  - Sepp Hochreiter & Jürgen Schmidhuber (1997). "Long short-term memory" *Neural Computation* **9** (8): 1735–1780.
  - Many earlier-to-read materials, especially after 2013
- The best way so far to train RNNs well
  - Increasingly popular in speech/language processing
  - Attracted big attention from ASR community at ICASSP-2013's DNN special session (Graves et al.)
  - Huge progress since then

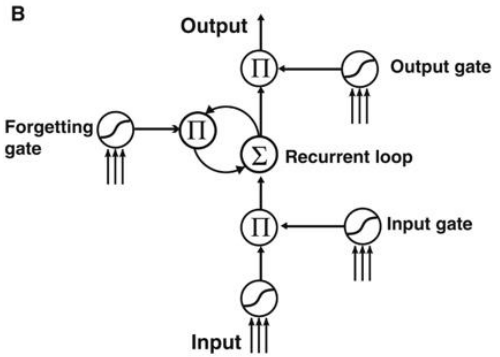
Microsoft Research

Many ways to show an LSTM cell

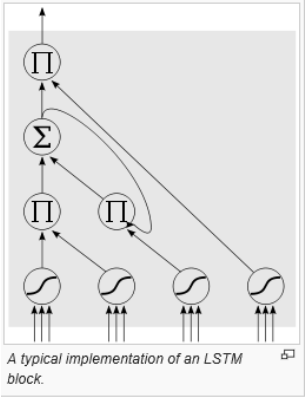


$$i_t = \sigma\left(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i\right)$$
$$f_t = \sigma\left(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f\right)$$
$$c_t = f_t c_{t-1} + i_t \tanh\left(W_{xc}x_t + W_{hc}h_{t-1} + b_c\right)$$
$$o_t = \sigma\left(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t + b_o\right)$$
$$h_t = o_t \tanh(c_t)$$





- Long Short-Term Memory (LSTM)[52,53]
- Provides vanishing gradient problem solution
- Input, Output and Forget[54] gates flow information through
- Linear memory cell (called Constant Error Carousel – CEC)


A typical implementation of an LSTM block.

[52] **Hochreiter:95fki207r.**
[53] **Hochreiter:97lstm.**
[54] **Gers:99a.**

(Slide revised from: Koutnik & Schmidhuber, 2015)

Microsoft Research

Many ways to show an LSTM cell

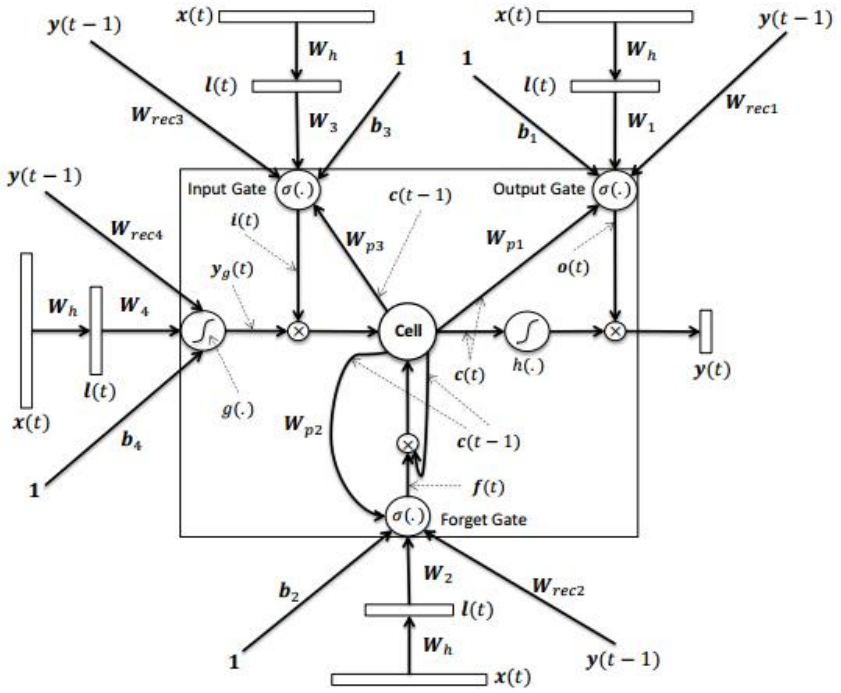


Fig. 2. The basic LSTM architecture used for sentence embedding

Deep Sentence Embedding Using Long Short-Term Memory Networks

Hamid Palangi, Li Deng, Yelong Shen, Jianfeng Gao, Xiaodong He, Jianshu Chen, Xinying Song, Rabab Ward

Jul 2015

***Abstract*—This paper develops a model that addresses sentence embedding, a hot topic in current natural language processing research, using recurrent neural networks (RNN) with Long Short-Term Memory (LSTM) cells. The proposed LSTM-RNN model sequentially takes each word in a sentence, extracts its information, and embeds it into a semantic vector. Due to its ability to capture long term memory, the LSTM-RNN accumulates increasingly richer information as it goes through the sentence, and when it reaches the last word, the hidden layer of the network** embedding is learned using sentence pairs. As a result, sentence embedding can better discover salient words and topics in a sentence, and thus is more suitable for tasks that require computing semantic similarities between text strings. By mapping texts into a unified semantic representation, the embedding vector can be further used for different language processing applications, such as machine translation [1], sentiment analysis [2], and information retrieval [3]. In machine translation

$$
\begin{aligned}
\mathbf{y}_g(t) &= g(\mathbf{W}_4\mathbf{l}(t) + \mathbf{W}_{\text{rec4}}\mathbf{y}(t-1) + \mathbf{b}_4) \\
\mathbf{i}(t) &= \sigma(\mathbf{W}_3\mathbf{l}(t) + \mathbf{W}_{\text{rec3}}\mathbf{y}(t-1) + \mathbf{W}_{p3}\mathbf{c}(t-1) + \mathbf{b}_3) \\
\mathbf{f}(t) &= \sigma(\mathbf{W}_2\mathbf{l}(t) + \mathbf{W}_{\text{rec2}}\mathbf{y}(t-1) + \mathbf{W}_{p2}\mathbf{c}(t-1) + \mathbf{b}_2) \\
\mathbf{c}(t) &= \mathbf{f}(t) \circ \mathbf{c}(t-1) + \mathbf{i}(t) \circ \mathbf{y}_g(t) \\
\mathbf{o}(t) &= \sigma(\mathbf{W}_1\mathbf{l}(t) + \mathbf{W}_{\text{rec1}}\mathbf{y}(t-1) + \mathbf{W}_{p1}\mathbf{c}(t) + \mathbf{b}_1) \\
\mathbf{y}(t) &= \mathbf{o}(t) \circ h(\mathbf{c}(t)) \qquad (2)
\end{aligned}
$$

Microsoft Research

Many ways to show an LSTM cell

Distributed Compressive Sensing: A Deep Learning Approach

Hamid Palangi, Rabab Ward, Li Deng

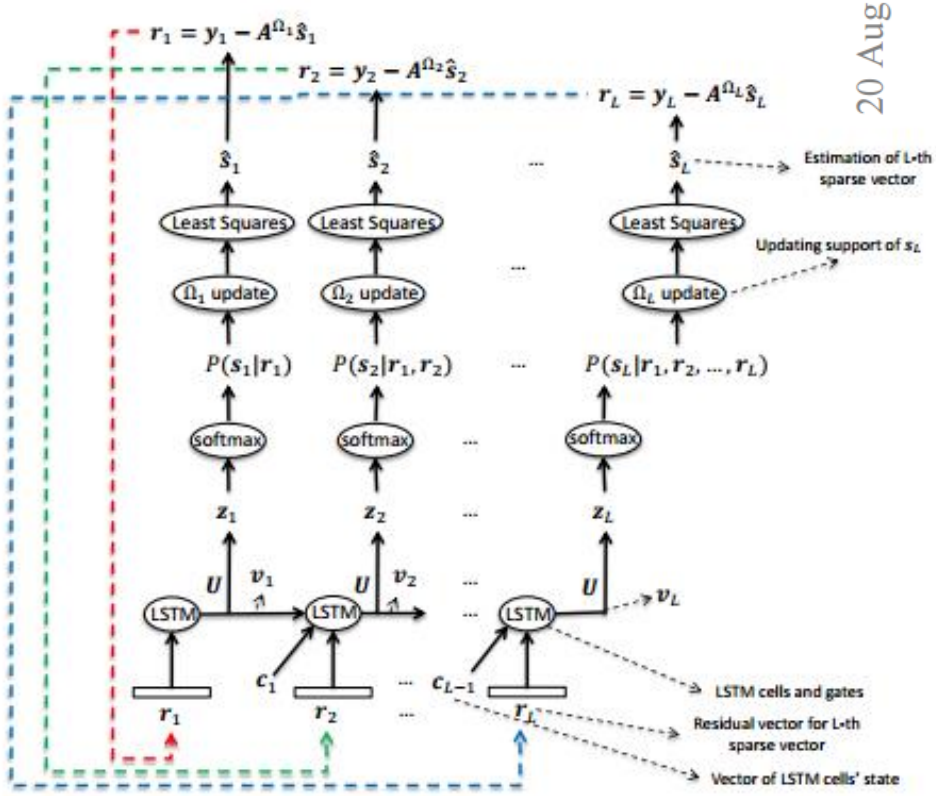
***Abstract*—We address the problem of compressed sensing with Multiple Measurement Vectors (MMVs) when the structure of sparse vectors in different channels depend on each other. *The sparse vectors are not necessarily joint sparse*. We capture this dependency by computing the conditional probability of each entry of each sparse vector to be non-zero given "residuals" of all previous sparse vectors. To compute these probabilities, we propose to use Long Short-Term Memory (LSTM) [1], a bottom up data driven model for sequence modelling. To compute model parameters we minimize a cross entropy cost function. We propose a greedy solver that uses above probabilities at**

where $\mathbf{y} \in \Re^{M\times 1}$ is the known measured vector and $\mathbf{\Phi} \in \Re^{M\times N}$ is a random measurement matrix. An important assumption needed by the decoder to uniquely recover $\mathbf{x}$ given $\mathbf{y}$ and $\mathbf{\Phi}$, is that $\mathbf{x}$ is sparse in a given basis $\mathbf{\Psi}$. This means that

$$\mathbf{x} = \mathbf{\Psi}\mathbf{s} \quad (2)$$

where $\mathbf{s}$ is $K-sparse$, i.e., $\mathbf{s}$ has at most K non-zero elements. The basis $\mathbf{\Psi}$ can be complete; i.e., $\mathbf{\Psi} \in \Re^{N\times N}$, or over-complete; i.e., $\mathbf{\Psi} \in \Re^{N\times N_1}$ where $N < N_1$ (compressed sensing for over-complete dictionaries is

20 Aug 2015



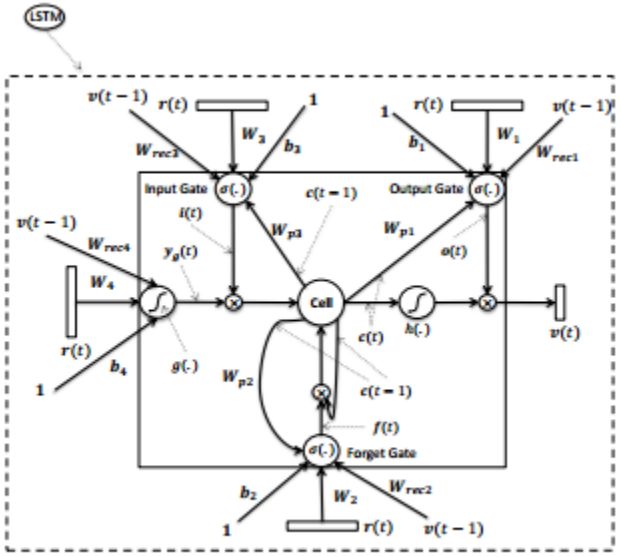
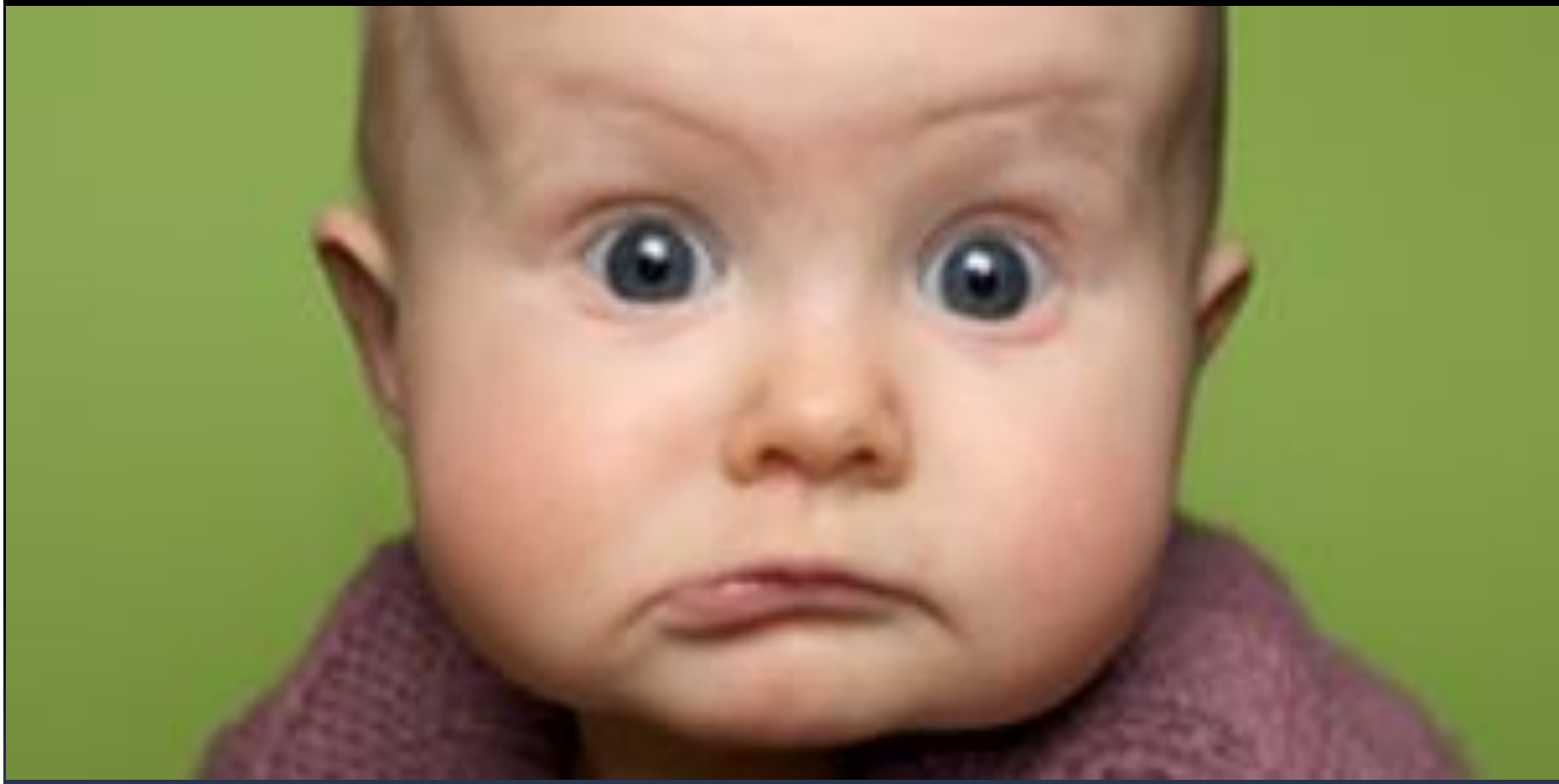


Fig. 2. Block diagram of the Long Short-Term Memory (LSTM).

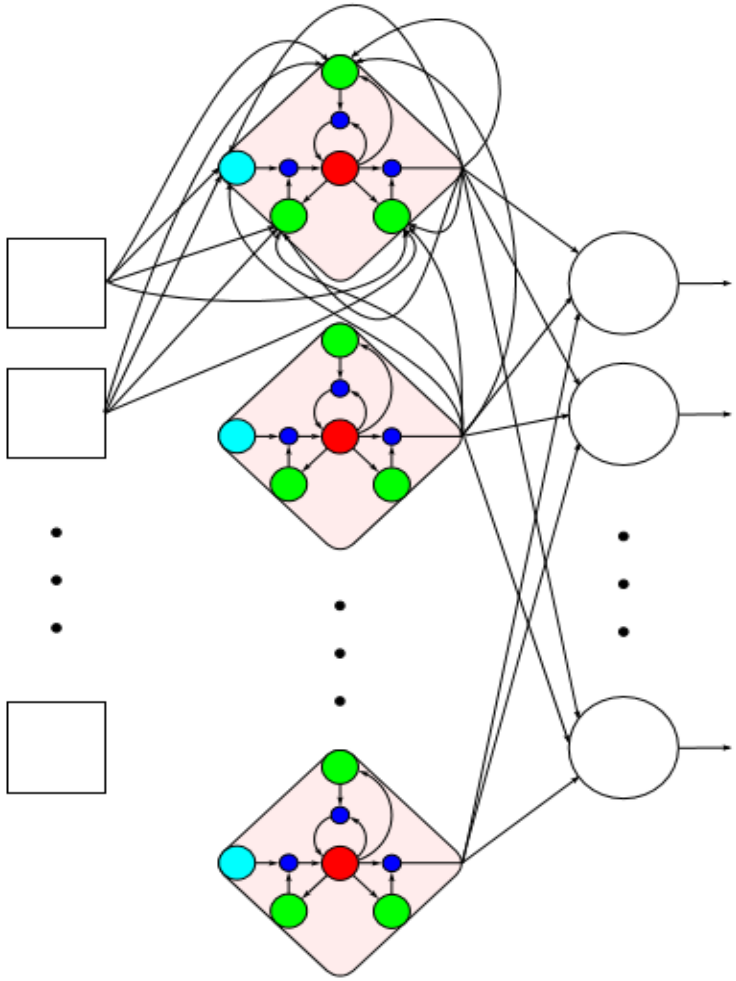
Huh?....



Don’t worry, we’ll come back to this shortly

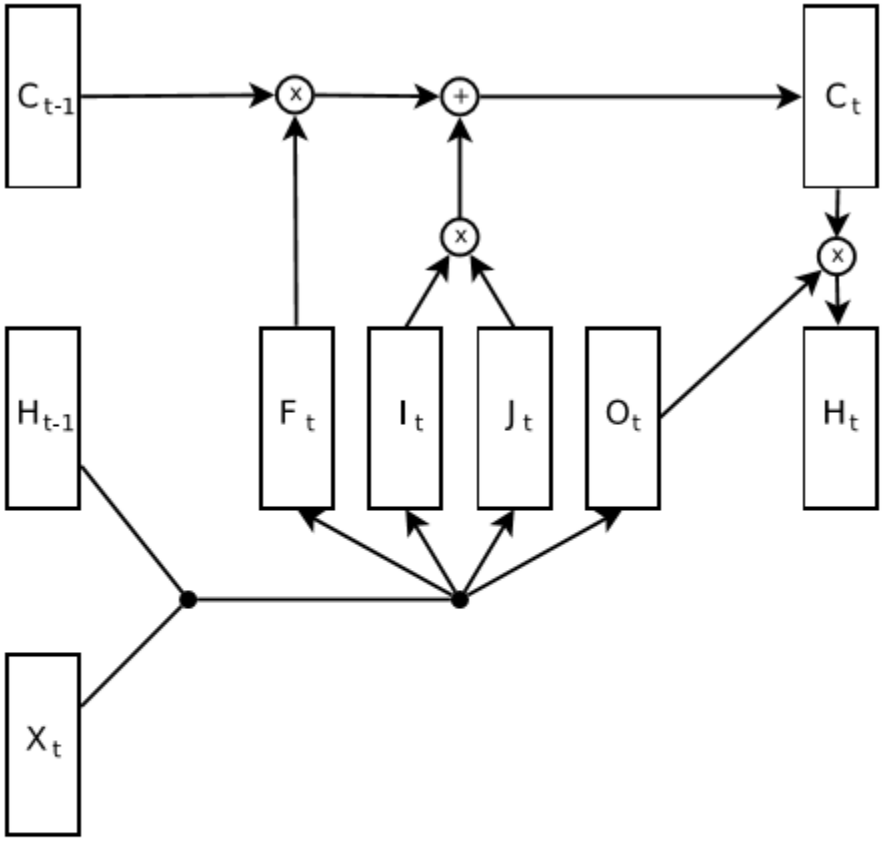
Microsoft Research

LSTM Cells in an RNN



Microsoft Research

LSTM cell unfolding over time



$$
\begin{aligned}
i_t &= \tanh(W_{\mathrm{xi}}x_t + W_{\mathrm{hi}}h_{t-1} + b_{\mathrm{i}}) \\
j_t &= \mathrm{sigm}(W_{\mathrm{xj}}x_t + W_{\mathrm{hj}}h_{t-1} + b_{\mathrm{j}}) \\
f_t &= \mathrm{sigm}(W_{\mathrm{xf}}x_t + W_{\mathrm{hf}}h_{t-1} + b_{\mathrm{f}}) \\
o_t &= \tanh(W_{\mathrm{xo}}x_t + W_{\mathrm{ho}}h_{t-1} + b_{\mathrm{o}}) \\
c_t &= c_{t-1} \odot f_t + i_t \odot j_t \\
h_t &= \tanh(c_t) \odot o_t
\end{aligned}
$$

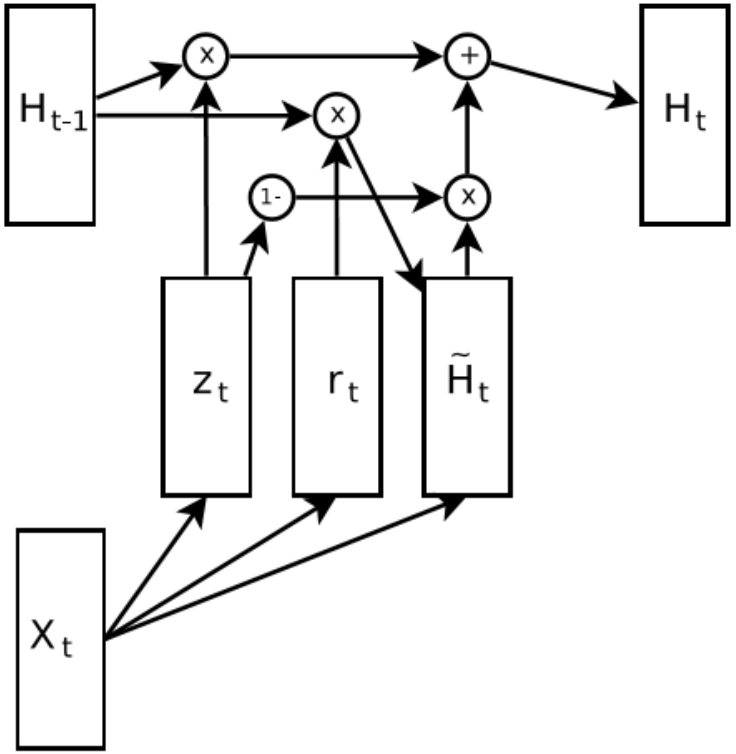
Figure 1. The LSTM architecture. The value of the cell is increased by $i_t \odot j_t$, where $\odot$ is element-wise product. The LSTM's output is typically taken to be h_t, and c_t is not exposed. The forget gate f_t allows the LSTM to easily reset the value of the cell.

(Jozefowics, Zarembe, Sutskever, ICML 2015)

Microsoft Research

Gated Recurrent Unit (GRU)

(simpler than LSTM; no output gates)



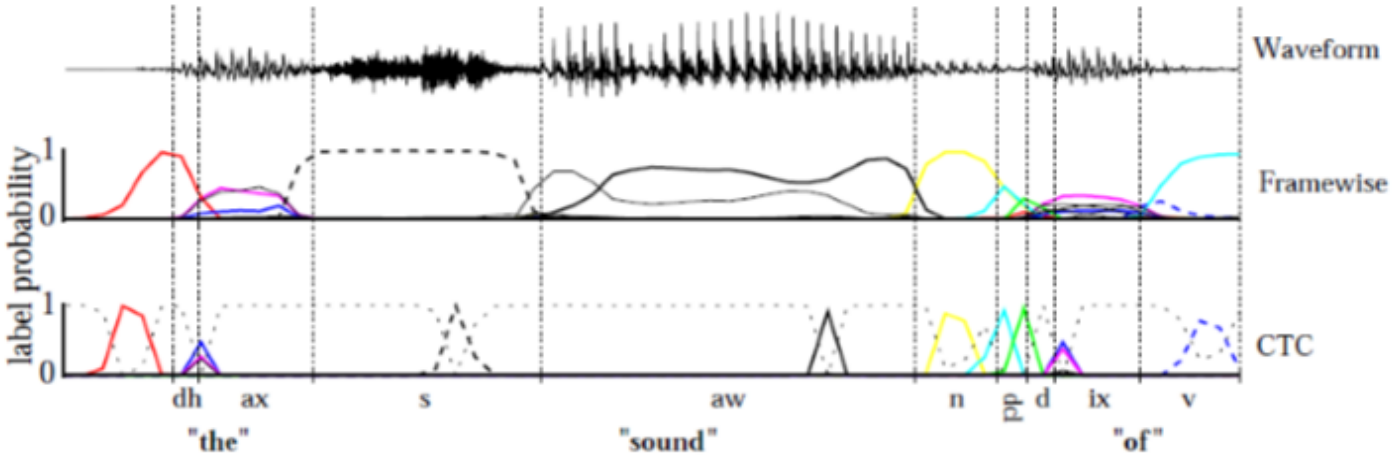
$$
\begin{aligned}
r_t &= \text{sigm}\left(W_{\text{xr}} x_t + W_{\text{hr}} h_{t-1} + b_{\text{r}}\right) \\
z_t &= \text{sigm}\left(W_{\text{xz}} x_t + W_{\text{hz}} h_{t-1} + b_{\text{z}}\right) \\
\tilde{h}_t &= \tanh\left(W_{\text{xh}} x_t + W_{\text{hh}} (r_t \odot h_{t-1}) + b_{\text{h}}\right) \\
h_t &= z_t \odot h_{t-1} + (1 - z_t) \odot \tilde{h}_t
\end{aligned}
$$

Figure 2. The Gated Recurrent Unit. Like the LSTM, it is hard to tell, at a glance, which part of the GRU is essential for its functioning.

(Jozefowics, Zarembe, Sutskever, ICML 2015; Google Kumar et al., arXiv, July, 2015; Metamind)

Microsoft Research

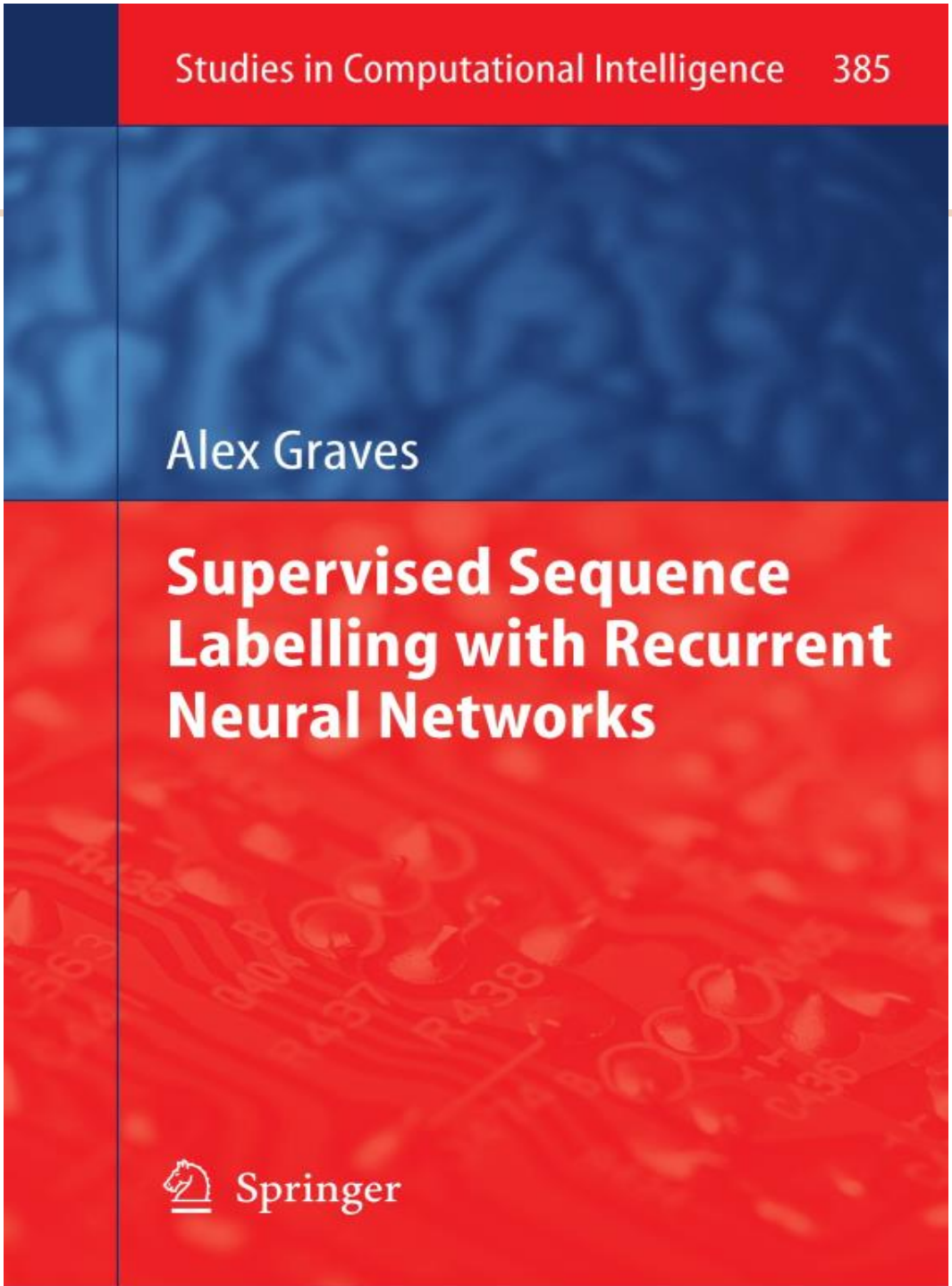
Connectionist Temporal Classification



- Connectionist Temporal Classification (CTC)[60]
- Automatically transforms frame labels to classification segments
- Provides labels for all time steps
- Allows to back-propagate the error from sparse labels

[60] **Graves:06icml**.

Microsoft Research



(2006, 2012)

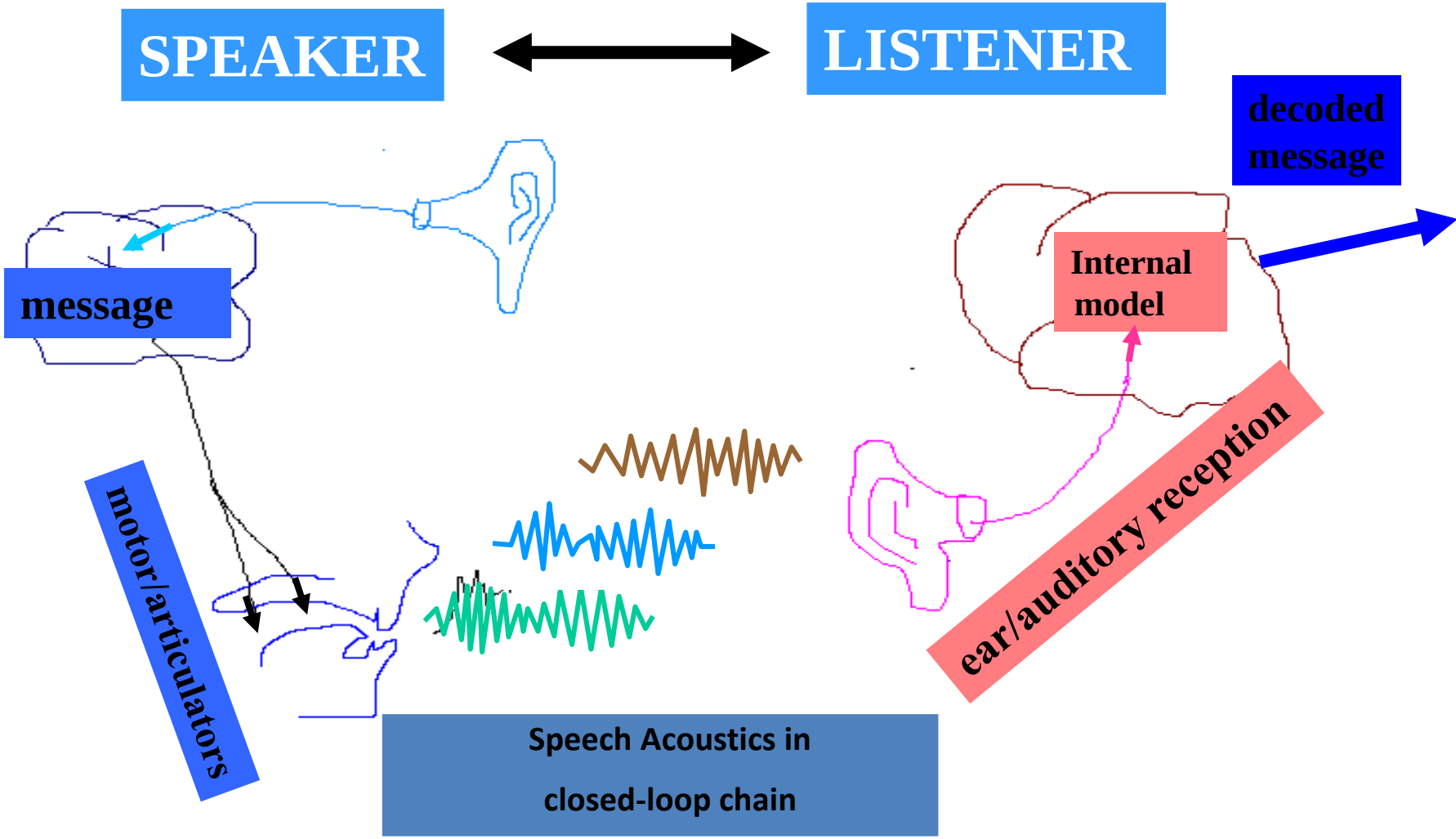
Part II: Speech

Deep/Dynamic Structure in Human Speech Production and Perception

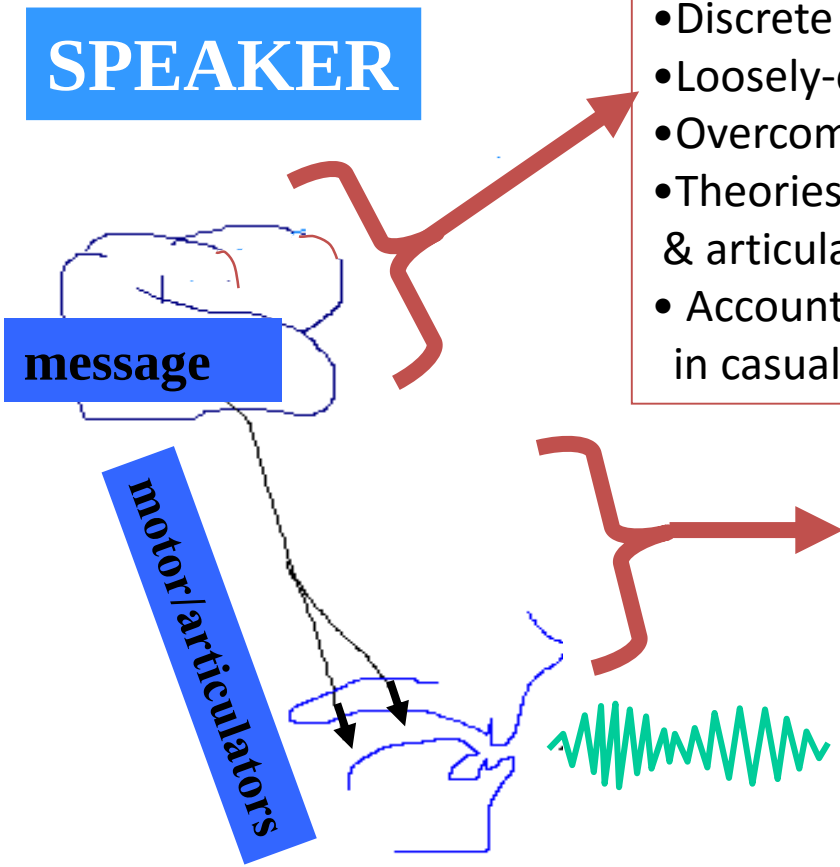
(part of my tutorial at 2009 NIPS WS)

Microsoft Research

Production & Perception: Closed-Loop Chain



Encoder: Two-Stage Production Mechanisms



Phonology (higher level):

- Symbolic encoding of linguistic message
- Discrete representation by phonological features
- Loosely-coupled multiple feature tiers
- Overcome beads-on-a-string phone model
- Theories of distinctive features, feature geometry & articulatory phonology
- Account for partial/full sound deletion/modification in casual speech

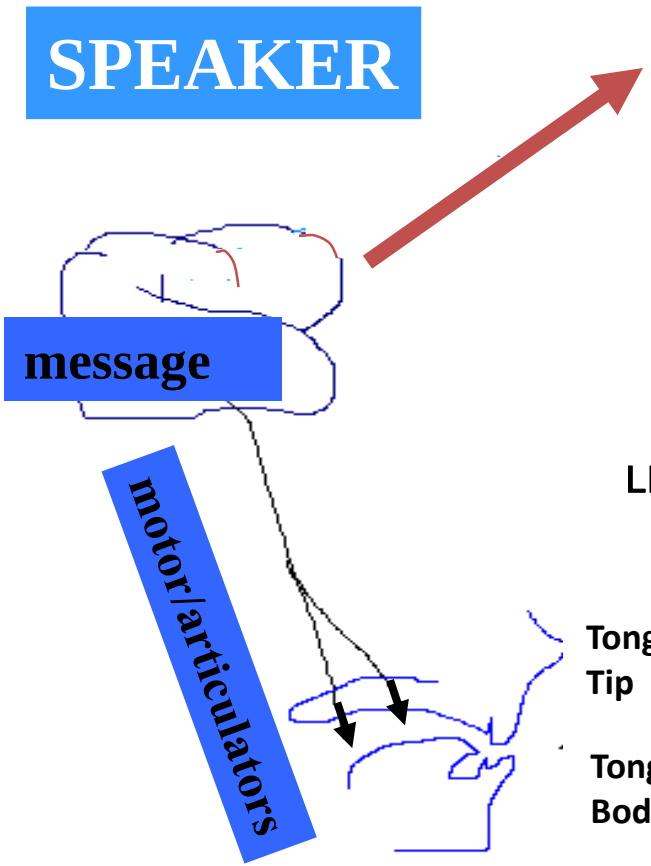
Phonetics (lower level):

- Convert discrete linguistic features to continuous acoustics
- Mediated by motor control & articulatory dynamics
- Mapping from articulatory variables to VT area function to acoustics
- Account for co-articulation and reduction (target undershoot), etc.

Encoder: Phonological Modeling

Computational phonology:

- **Represent pronunciation variations as constrained factorial Markov chain**
- **Constraint: from articulatory phonology**
- **Language-universal representation**



ten themes

/ t ɛ n ɵ i: m z /

LIPS:

Labial-closure

Tongue Tip

Alveolar closure

Alveolar closure

dental constr.

Alveolar constr.

Tongue Body

Mid / Front

High / Front

VEL:

Nasality

Nasality

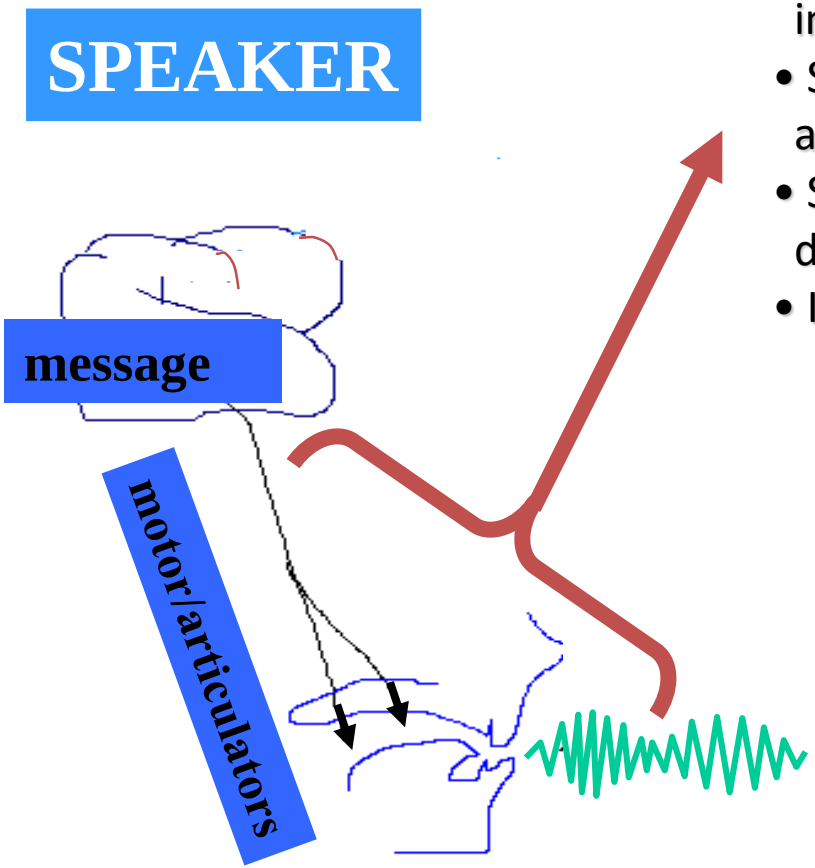
GLO:

Aspiration

Voicing

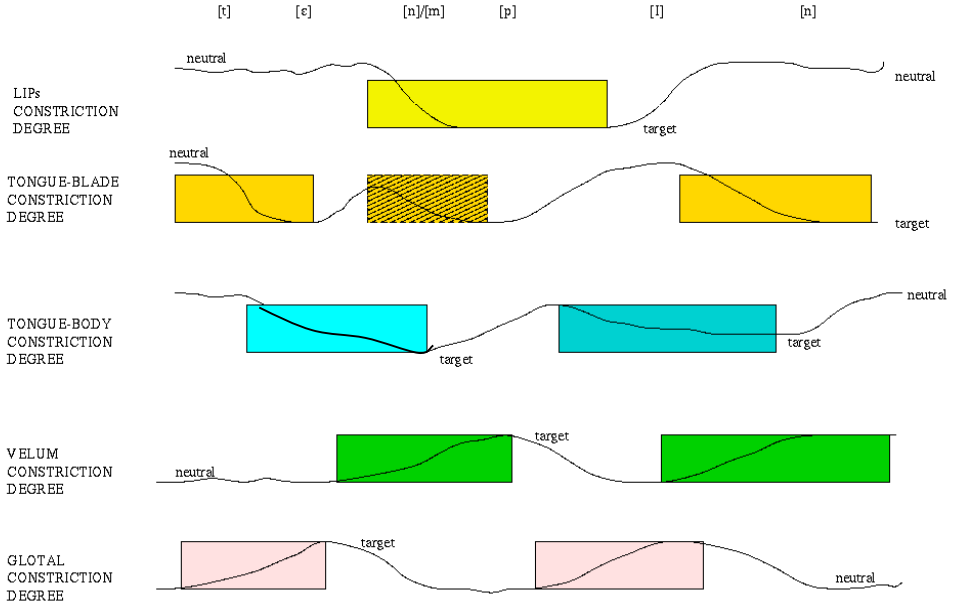
Voicing

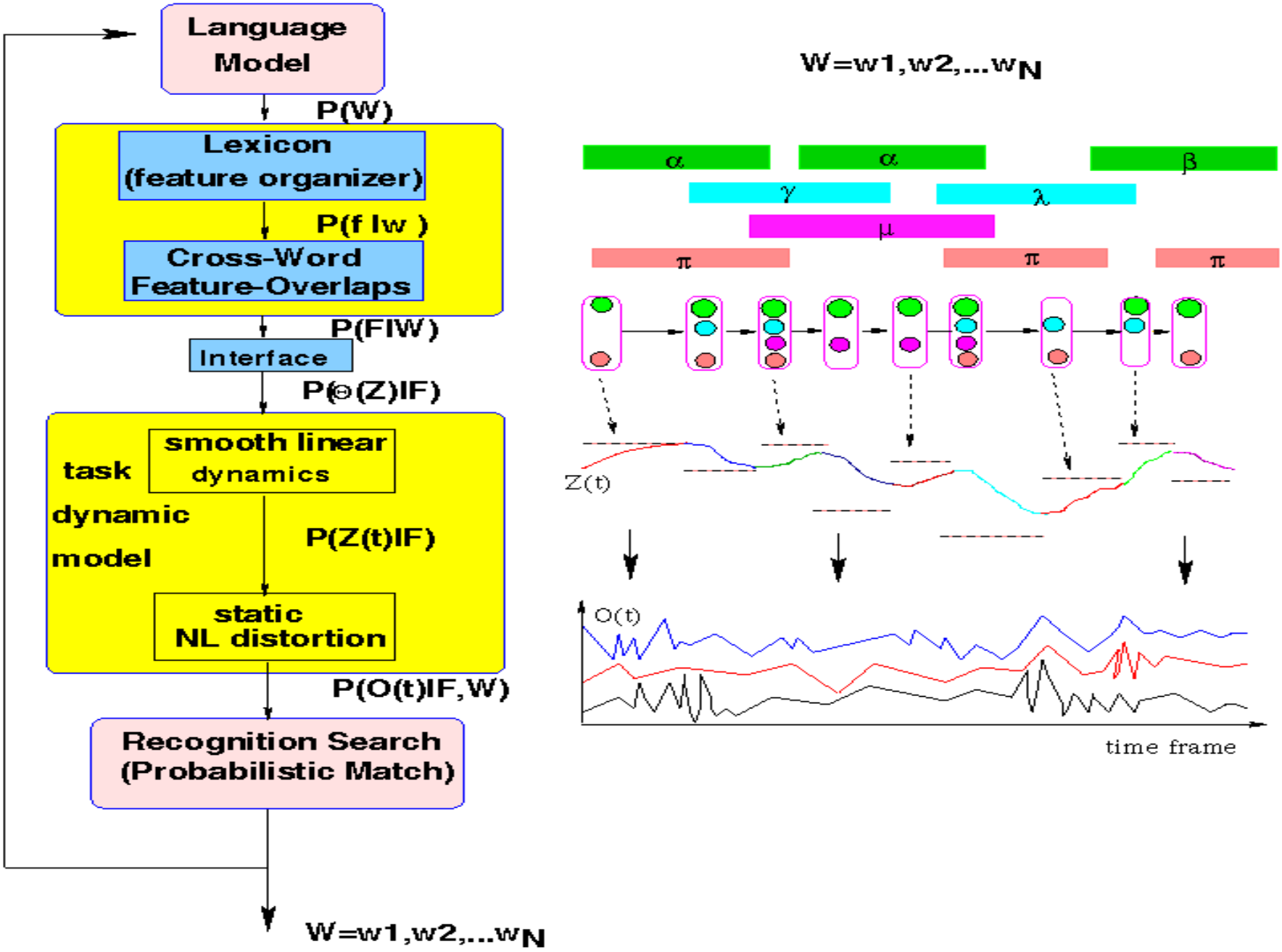
Encoder: Phonetic Modeling



Computational phonetics:

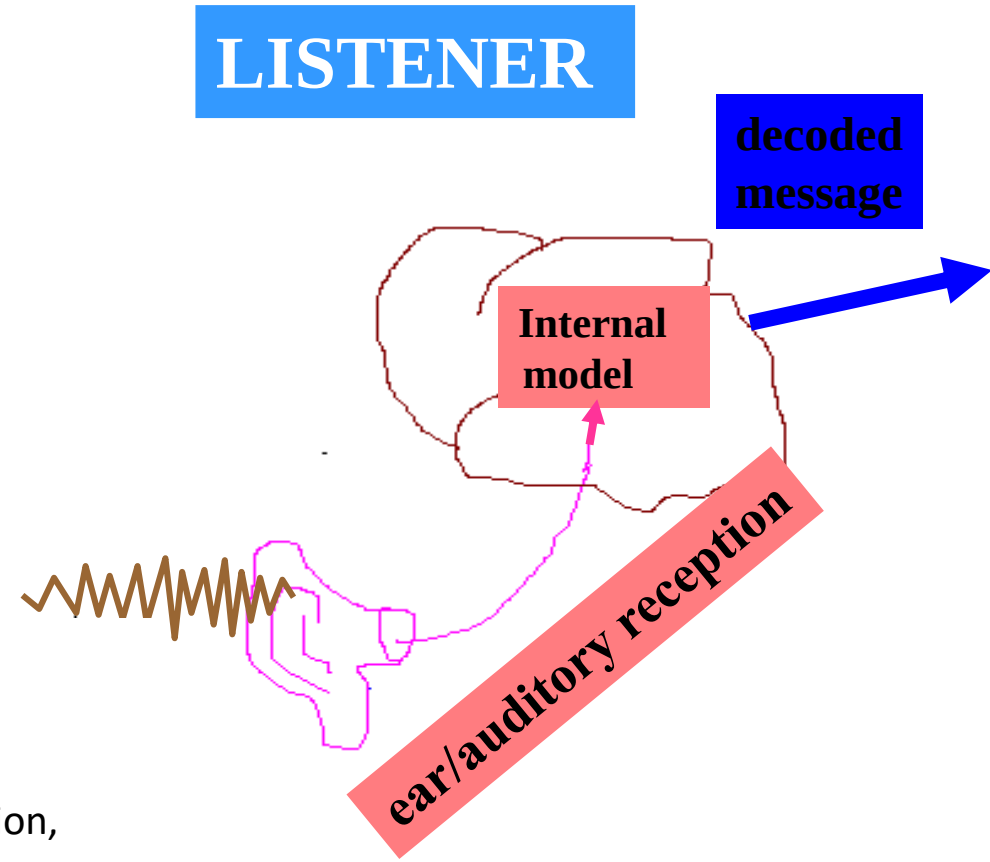
- Segmental factorial HMM for sequential target in articulatory or vocal tract resonance domain
- Switching trajectory model for target-directed articulatory dynamics
- Switching nonlinear state-space model for dynamics in speech acoustics
- Illustration:




Language Model
P(W)
Lexicon (feature organizer)
P(f lw)
Cross-Word Feature-Overlaps
P(FIW)
Interface
P(Θ(Z)IF)
task dynamic model
smooth linear dynamics
P(Z(t)IF)
static NL distortion
P(O(t)IF,W)
Recognition Search (Probabilistic Match)
W=w1,w2,...wN
W=w1,w2,...wN
α
α
β
γ
λ
μ
π
π
π
Z(t)
O(t)
time frame

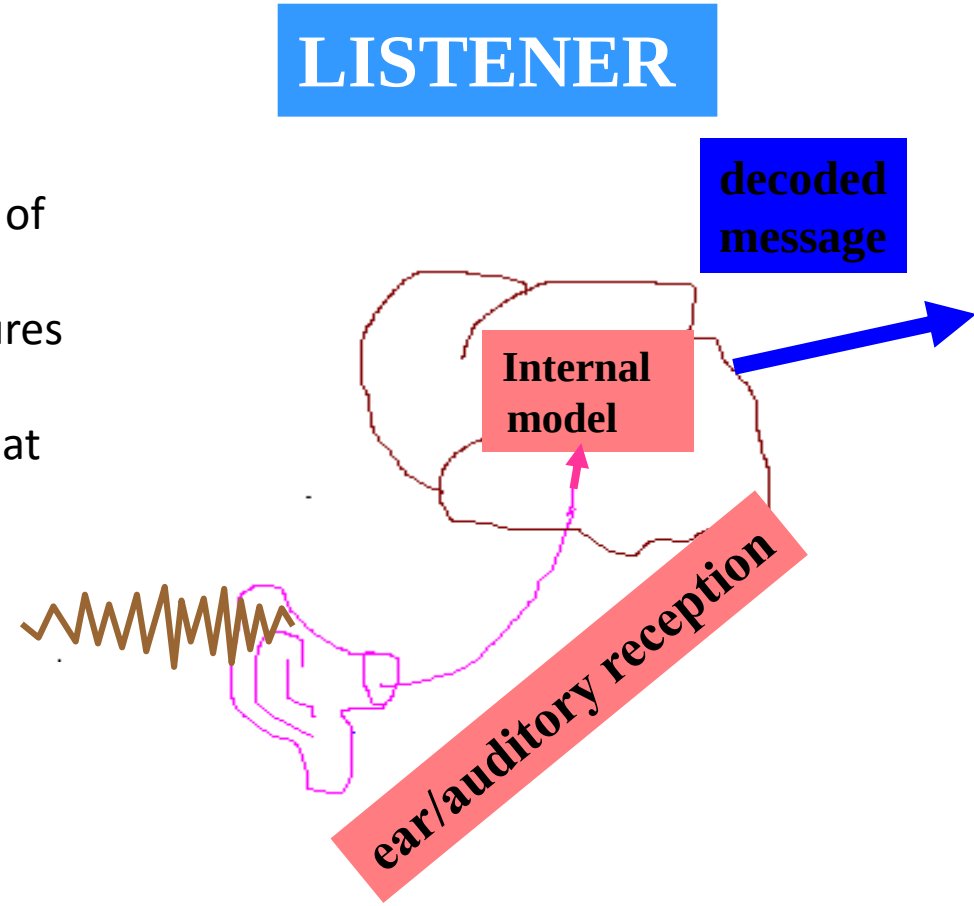
Decoder I: Auditory Reception

- Convert speech acoustic waves into efficient & robust auditory representation
- This processing is largely independent of phonological units
- Involves processing stages in cochlea (ear), cochlear nucleus, SOC, IC,…, all the way to A1 cortex
- Principal roles:
  1) combat environmental acoustic distortion;
  2) detect relevant speech features
  3) provide temporal landmarks to aid decoding
- Key properties:
  1) Critical-band freq scale, logarithmic compression,
  2) adapt freq selectivity, cross-channel correlation,
  3) sharp response to transient sounds
  4) modulation in independent frequency bands,
  5) binaural noise suppression, etc.



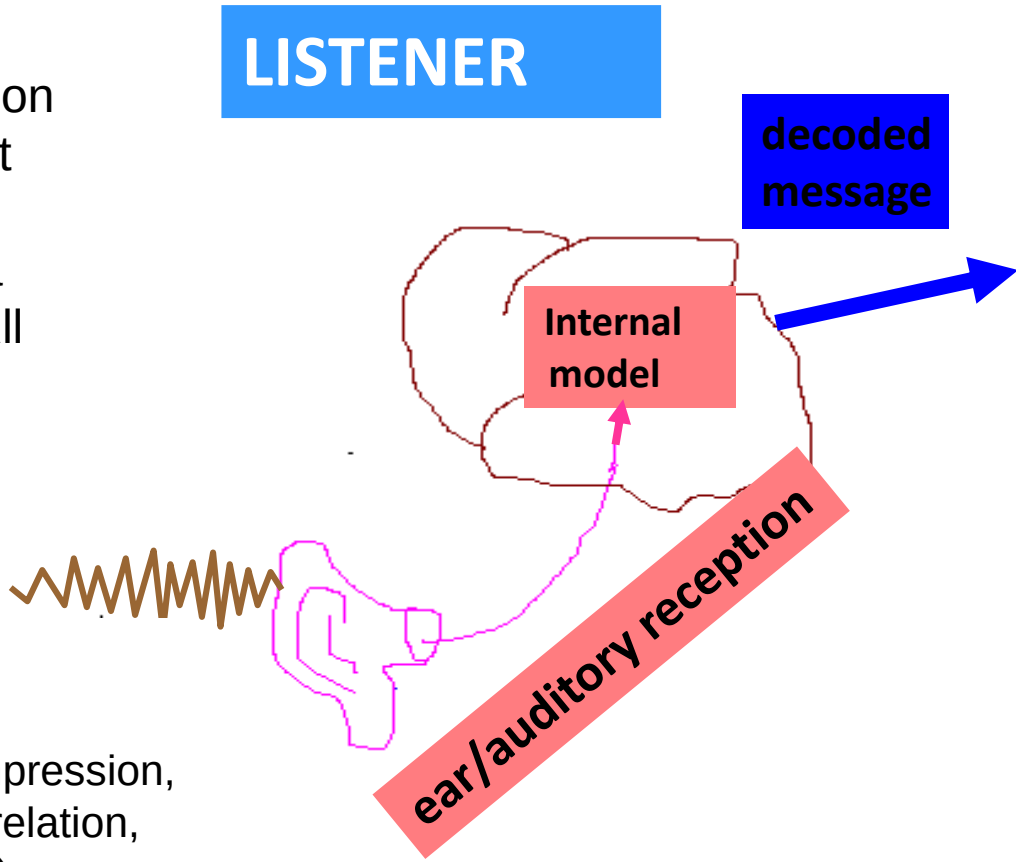
Decoder II: Cognitive Perception

- Cognitive process: recovery of linguistic message
- Relies on
  1) "Internal" model: structural knowledge of the encoder (production system)
  2) Robust auditory representation of features
  3) Temporal landmarks
- Child speech acquisition process is one that gradually establishes the "internal" model
- Strategy: analysis by synthesis
- i.e., Probabilistic inference on (deeply) hidden linguistic units using the internal model
- No motor theory: the above strategy requires no articulatory recovery from speech acoustics



Human Speech Perception (decoder)

- Convert speech acoustic waves into efficient & robust auditory representation
- This processing is largely independent of phonological units
- Involves processing stages in cochlea (ear), cochlear nucleus, SOC, IC,…, all the way to A1 cortex
- Two principal roles:
  1) combat environmental acoustic distortion;
  2) provide temporal landmarks to aid decoding
- Key properties:
  1) Critical-band freq scale, logarithmic compression,
  2) adapt freq selectivity, cross-channel correlation,
  3) sharp response to transient sounds (CN),
  4) modulation in independent frequency bands,
  5) binaural noise suppression, etc.



Types of Speech Perception Theories

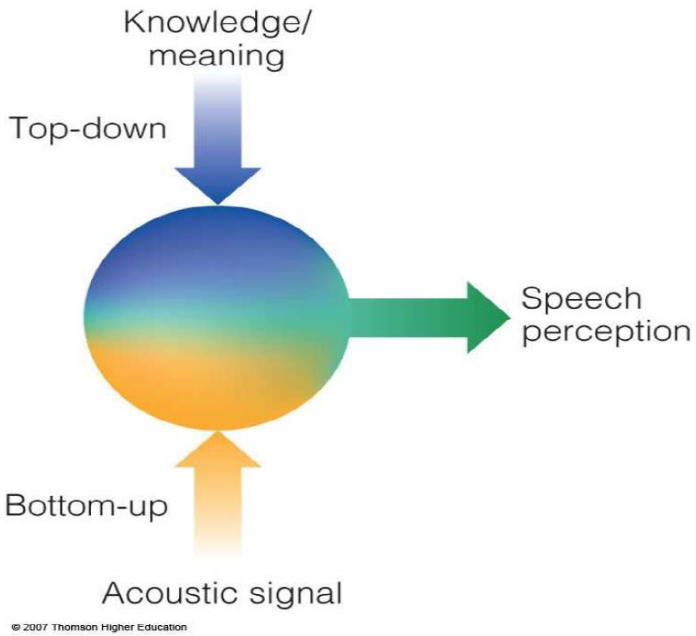
- Active vs. Passive
- Bottom up vs./and Top Down
- Autonomous vs. Interactive

Active vs. Passive

- **Active theories** suggests that speech perception and production are closely related
  - Listener knowledge of how sounds are produced facilitates recognition of sounds
- **Passive theories** emphasizes the sensory aspects of speech perception
  - Listeners utilize internal filtering mechanisms
  - Knowledge of vocal tract characteristics plays a minor role, for example when listening in noise conditions

Bottom up ~~vs.~~ & Top Down

- Top-down processing works with knowledge a listener has about a language, context, experience, etc.
  - Listeners use stored information about language and the world to make sense of the speech
- Bottom-up processing works in the absence of a knowledge base providing top-down information
  - listeners receive auditory information, convert it into a neural signal and process the phonetic feature information

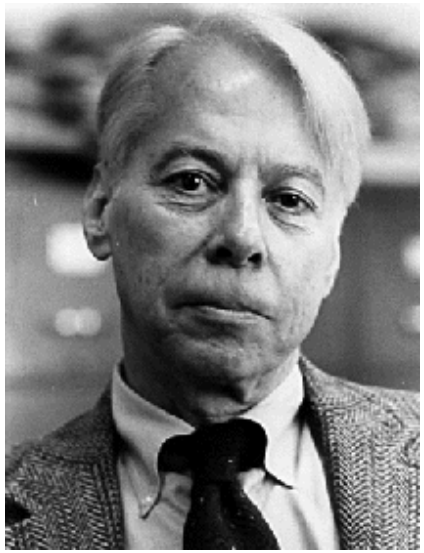


Specific Speech Perception Theories

- Motor Theory
- Acoustic Invariance Theory
- Direct Realism
- Trace Model (based on neural nets)
- Cohort Theory
- Fuzzy Logic Model of Perception
- Native Language Magnet Theory

Motor Theory

- Postulates speech is perceived by reference to how it is produced
  - when perceiving speech, listeners access their own knowledge of how phonemes are articulated
  - Articulatory gestures (such as rounding or pressing the lips together) are units of perception that directly provide the listener with phonetic information



Liberman, Cooper, Shankweiler, & Studdert-Kennedy, 1967

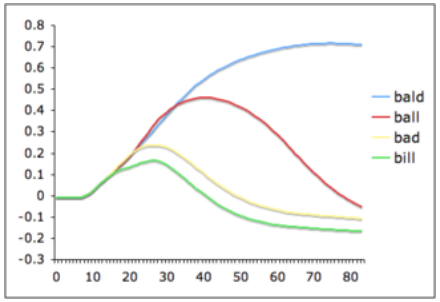
Acoustic Invariance Theory

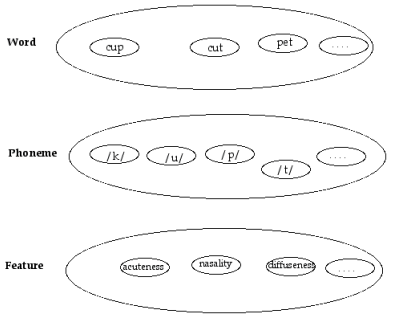
- Listeners inspect the incoming signal for the so-called acoustic landmarks which are particular events in the spectrum carrying information about gestures which produced them.
- Gestures are limited by the capacities of humans' articulators and listeners are sensitive to their auditory correlates, the lack of invariance simply does not exist in this model.
- The acoustic properties of the landmarks constitute the basis for establishing the distinctive features.
- Bundles of the distinctive features uniquely specify phonetic segments (phonemes, syllables, words).

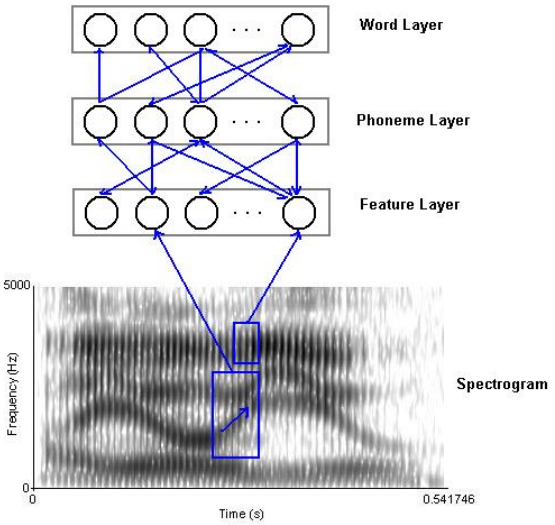
Stevens, K.N. (2002). "Toward a model of lexical access based on acoustic landmarks and distinctive features" (PDF). *Journal of the Acoustical Society of America* **111** (4): 1872–1891.

TRACE Model

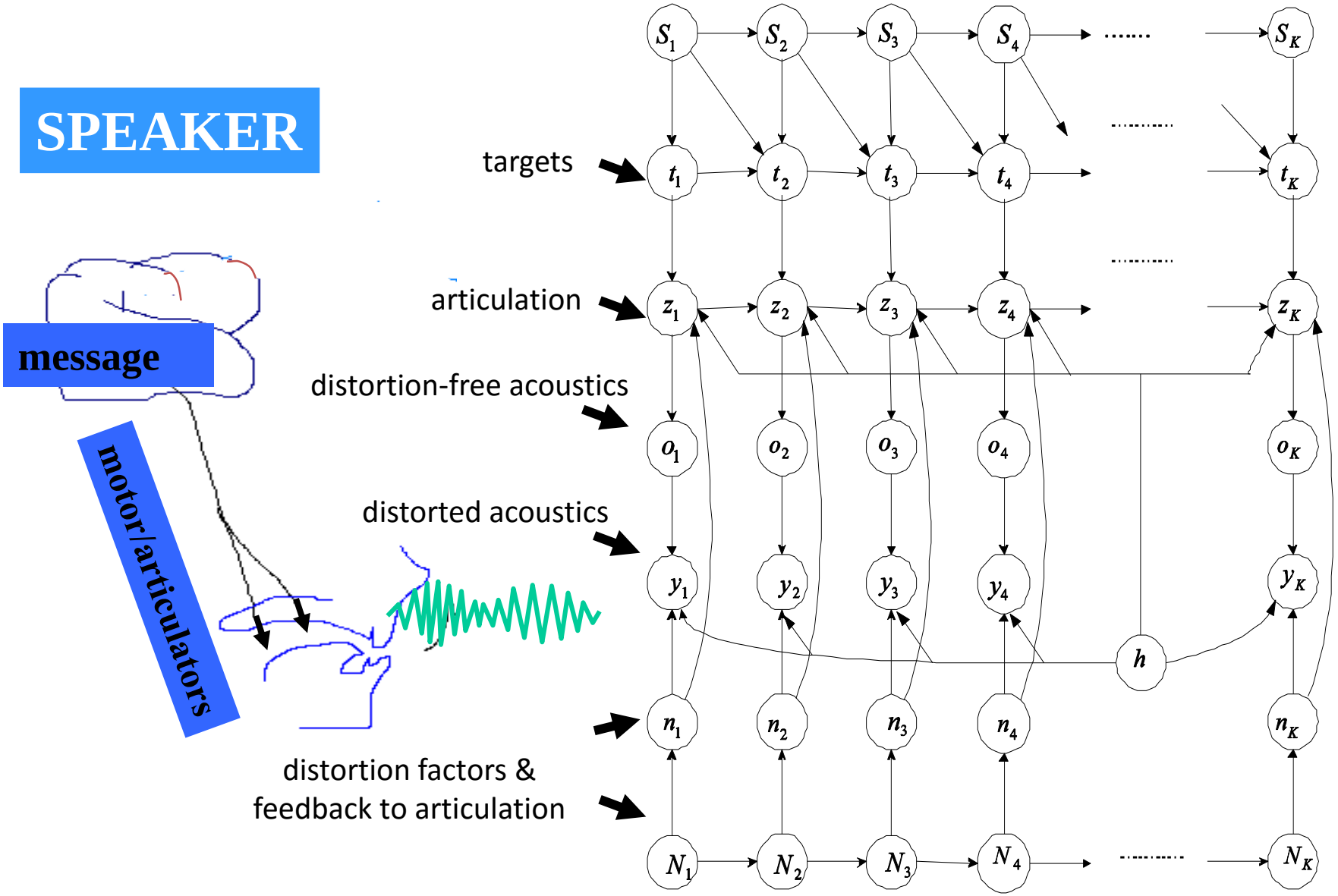
- For example, a listener hears the beginning of *bald*, and the words bald, ball, bad, bill become active in memory. Then, soon after, only bald and ball remain in competition (bad, bill have been eliminated because the vowel sound doesn't match the input).
  - Soon after, bald is recognized.
- TRACE simulates this process by representing the temporal dimension of speech, allowing words in the lexicon to vary in activation strength, and by having words compete during processing.







A Deep/Generative Model of Speech Production/Perception
--- Perception as "variational inference"



Deep Generative Models, Variational Interference/Learning, & Applications to Speech

Deep Learning

≈

Neural Networks, Deep

in space & **time (recurrent** LSTM), & deep RNN

+

Generative Models, Deep

in space & **time (dynamic**), & deep/hidden dynamic models

+

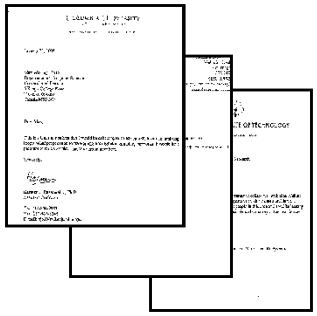
…, …, …

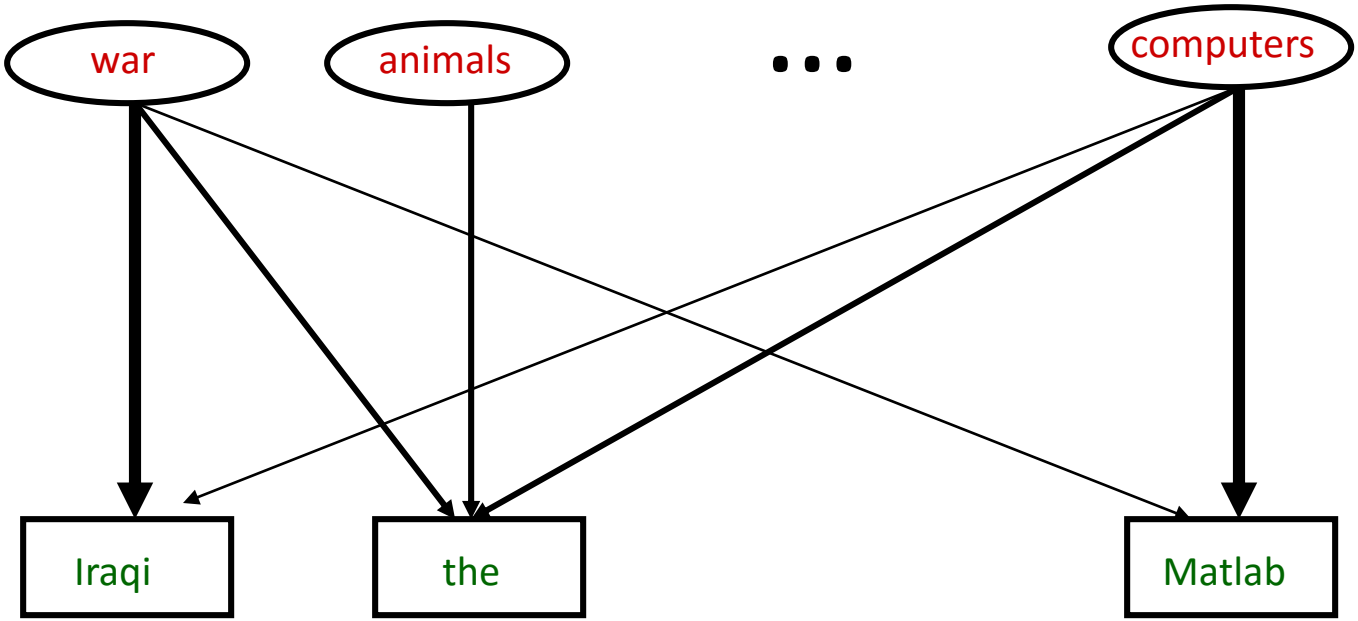
	Deep Neural Nets	Deep Generative Models
Structure	Graphical; info flow: **bottom-up**	Graphical; info flow: **top-down**
Incorp constraints & domain knowledge	Hard	**Easy**
Semi/*unsupervised*	Harder or impossible	*Easier, at least possible*
Interpretation	Harder	**Easy** (generative "story" on data and hidden variables)
Representation	**Distributed**	Localist (mostly); can be distributed also
Inference/decode	Easy	Harder (but note **recent progress**)
Scalability/compute	**Easier (regular computes/GPU)**	Harder (but note **recent progress**)
Incorp. uncertainty	Hard	**Easy**
Empirical goal	Classification, feature learning, …	Classification (via Bayes rule), latent variable inference…
Terminology	Neurons, activation/gate functions, weights …	Random vars, stochastic "neurons", potential function, parameters …
Learning algorithm	A single, unchallenged, algorithm -- BackProp	A major focus of open research, many algorithms, & more to come
Evaluation	On a black-box score – end performance	On almost every intermediate quantity
Implementation	Hard (but increasingly easier)	Standardized but insights needed
Experiments	Massive, real data	Modest, often simulated data
Parameterization	Dense matrices	Sparse (often PDFs); can be dense

Example: (Shallow) Generative Model



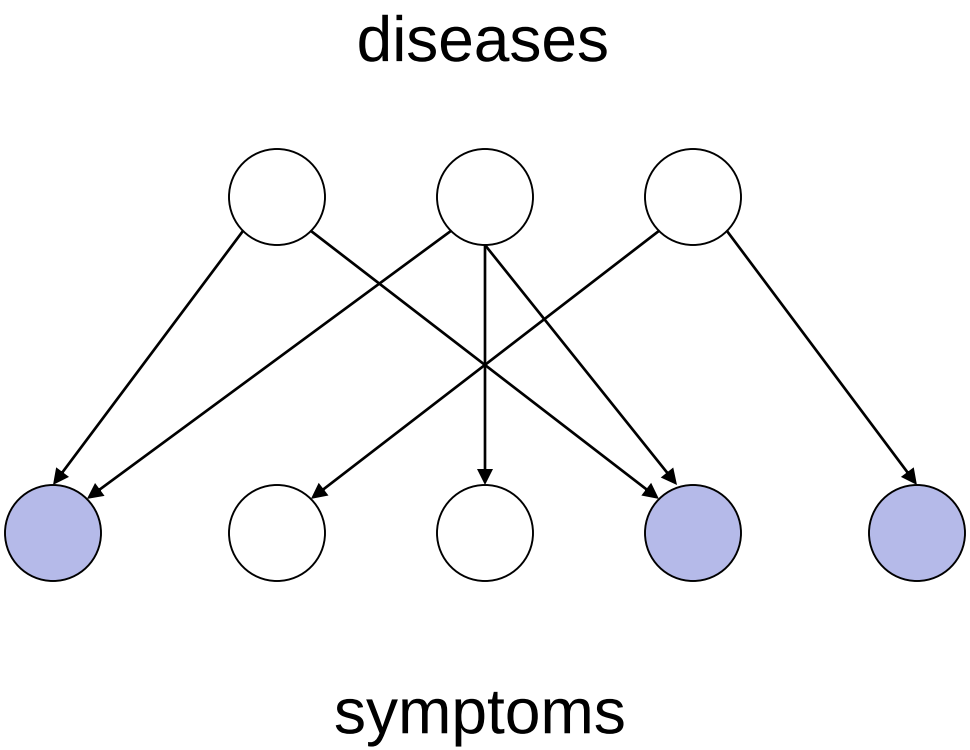
"TOPICS"
as hidden layer





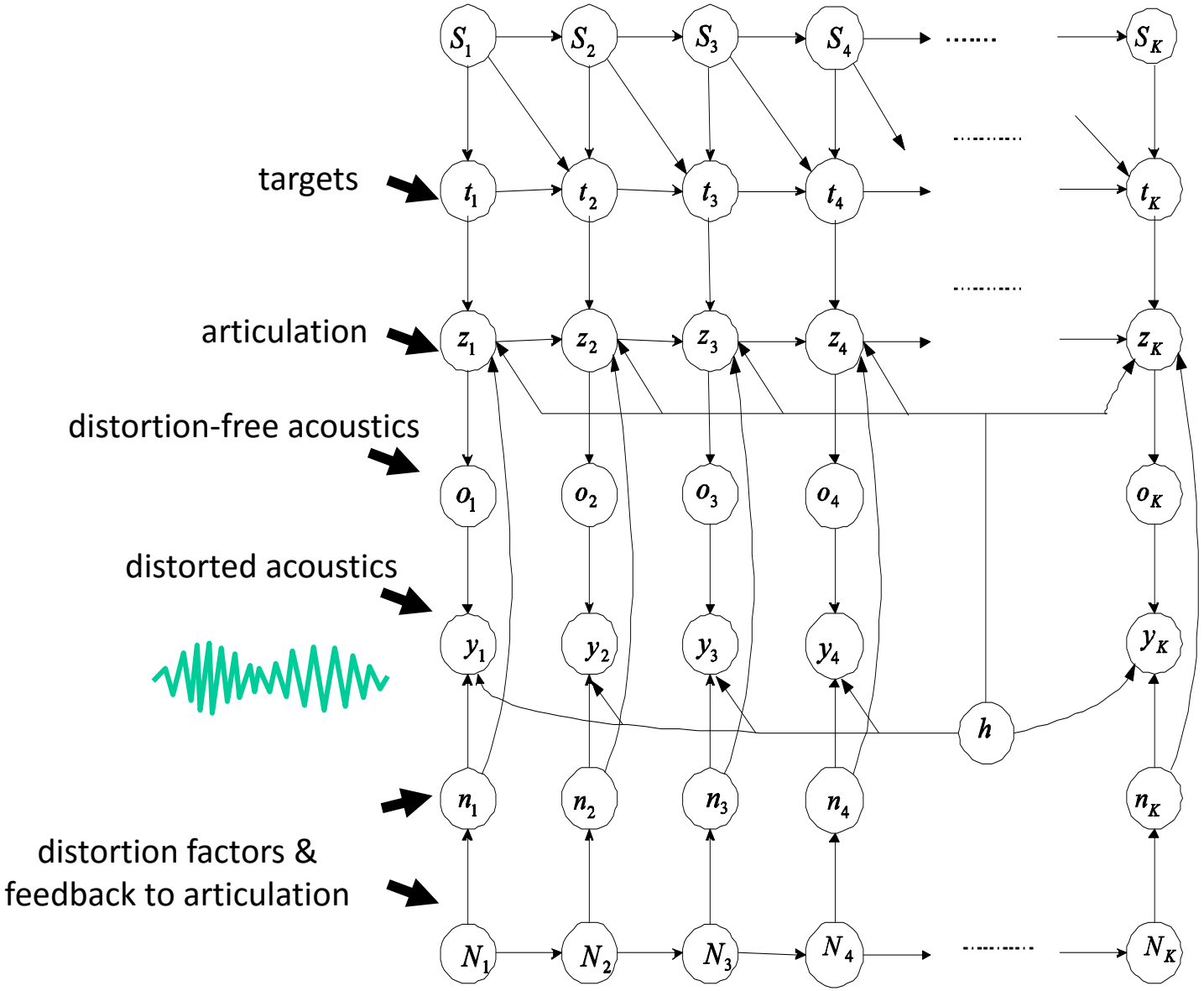
slide revised from: Max Welling

Another example: Medical Diagnosis



Inference problem: What is the most probable disease given the symptoms?

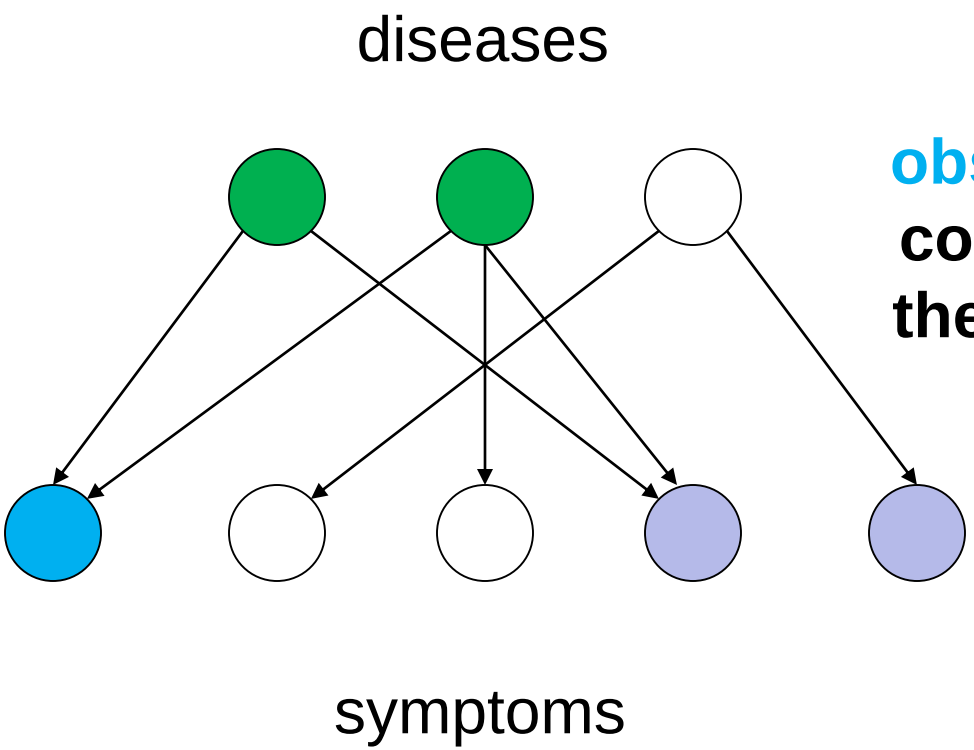
Example: (Deep) Generative Model



Deep Generative/Graphical Model Inference

- Key issues:
  - **Representation**: syntax and semantics (directed/undirected,variables/factors,..)
  - **Inference**: computing probabilities and most likely assignments/explanations
  - **Learning**: of model parameters based on observed data. *Relies on inference!*
- Inference is NP-hard (incl. approximation hardness)
- Exact inference: works for very limited subset of models/structures
  - E.g., chains or low-treewidth trees
- Approximate inference: highly computationally intensive
  - Deterministic: **Variational**, loopy belief propagation, expectation propagation
  - Numerical sampling (Monte Carlo): Gibbs sampling
- Variational learning:
  - EM algorithm
  - E-step uses variational inference (recent new advances in ML)

Variational inference/learning is not trivial: Example



Difficulty:

Explaining away:

observation introduces correlation of nodes in the parent hidden layer

Variational EM

Step 1: Maximize the bound with respect to Q

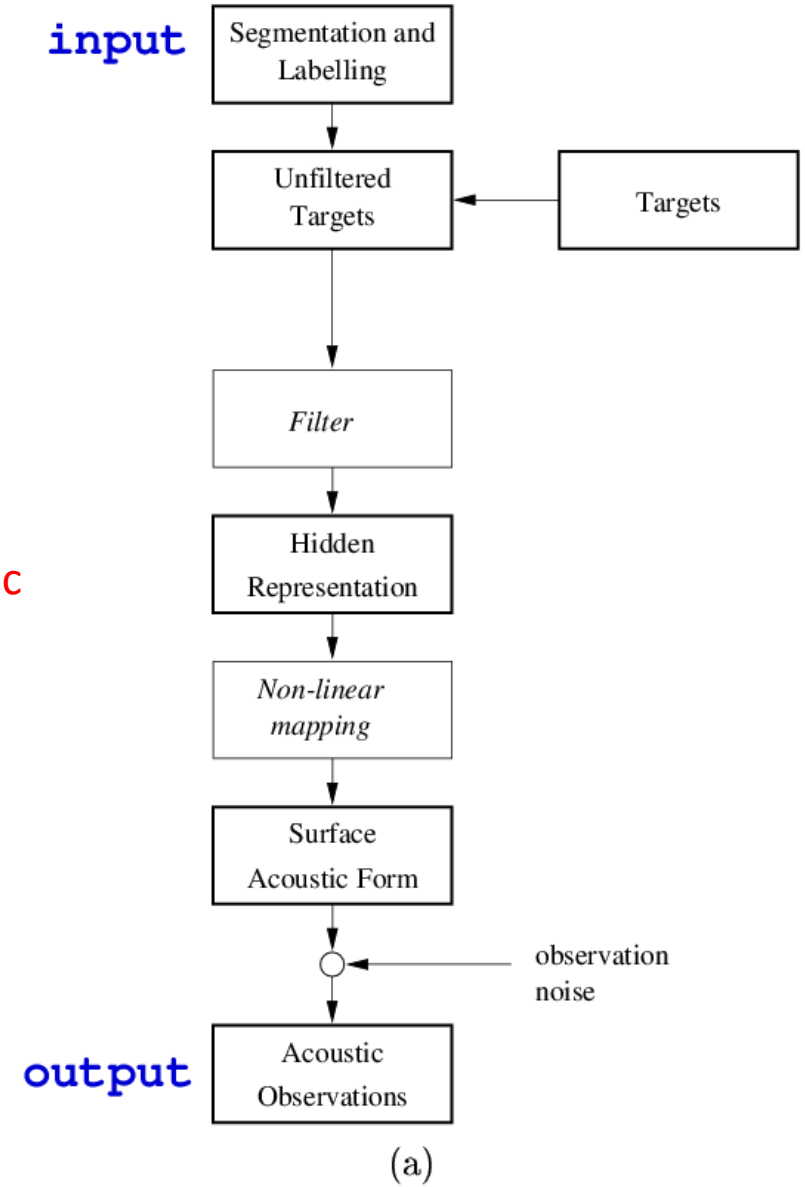
$$(\text{E step}): \; Q^{(k+1)} = \arg\max_Q L(Q, \theta^{(k)})$$

(Q approximates true posterior, often by factorizing it)

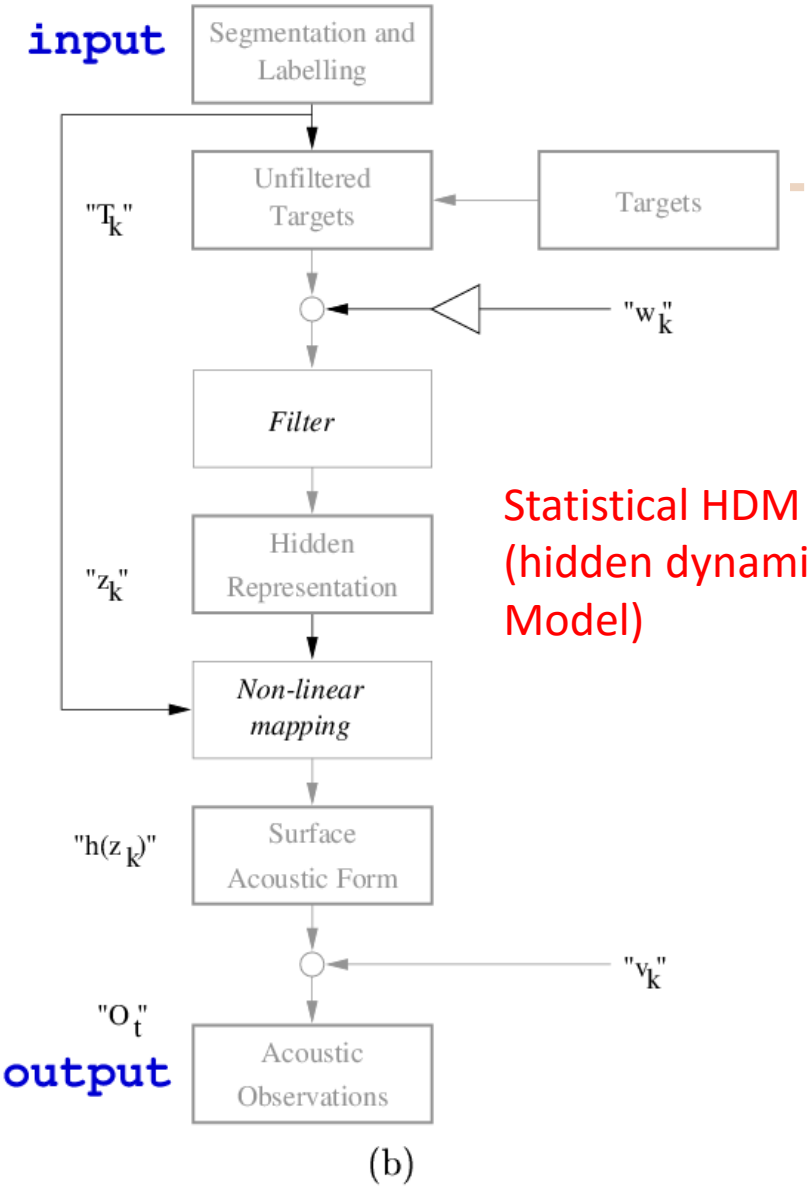
Step 2: Fix Q, maximize with respect to

$$(\text{M step}): \; \theta^{(k+1)} = \arg\max_\theta L(Q^{(k+1)}, \theta)$$

Note in traditional EM, Q is precise; e.g. posteriors computed by forward/backward algorithm for HMMs



Deterministic HDM



Statistical HDM (hidden dynamic Model)

Bridle, Deng, Picone, Richards, Ma, Kamm, Schuster, Pike, Reagan. Final Report for Workshop on Language Engineering, Johns Hopkins University, 1998. (experiments on Switchboard tasks)

ICASSP-2004

A MULTIMODAL VARIATIONAL APPROACH TO LEARNING AND INFERENCE IN SWITCHING STATE SPACE MODELS

Leo J. Lee[1,2], *Hagai Attias*[2], *Li Deng*[2] *and Paul Fieguth*[3]

University of Waterloo
[1]Electrical & Computer Engineering
[3]Systems Design Engineering
Waterloo, ON, N2L 3G1
Canada

[2]Microsoft Corporation
Microsoft Research
One Microsoft Way
Redmond, WA 98052-6339
USA

Auxiliary function:

$$\mathcal{F}[q] = \sum_{s_{1:N}} \int d\mathbf{x}_{1:N}\; q(s_{1:N}, \mathbf{x}_{1:N}) \cdot [\log p(\mathbf{y}_{1:N}, \mathbf{x}_{1:N}, s_{1:N}) - \log q(s_{1:N}, \mathbf{x}_{1:N})]$$

In the variational approach we approximate the exact posterior $p(s_{1:N}, \mathbf{x}_{1:N} \mid \mathbf{y}_{1:N})$ by a distribution with a tractable structure, denoted by q. Here we choose the following partially factorized structure shown graphically in Fig. 1:

$$p(s_{0:N}, \mathbf{x}_{0:N} \mid \mathbf{y}_{1:N}) \approx q(s_{0:N}, \mathbf{x}_{0:N} \mid \mathbf{y}_{1:N}) = \prod_{n=1}^{N} q(\mathbf{x}_n \mid s_n) q(s_n \mid s_{n-1}) \cdot q(x_0 \mid s_0) q(s_0) \; . \tag{5}$$

E-step: sufficient statistics. As usual, the variational equations above are coupled, with the equations for $\rho_{s,n}$, $\Gamma_{s,n}$ depend on $\eta_{s's,n}$, $\gamma_{s,n}$ and vice versa. These equations are solved iteratively starting from a random or more suitable initialization if available. The solution is the set of sufficient statistics

$$\varphi = \{\rho_{s,n}, \Gamma_{s,n}, \eta_{ss',n}, \gamma_{s,n}\} \tag{16}$$

which are moments of the variational posterior.

M-step: parameter estimation. Given the sufficient statistics φ, the derivation of the M-step is achieved by taking derivatives of $\mathcal{F}$ w.r.t. the model parameters (details omitted).

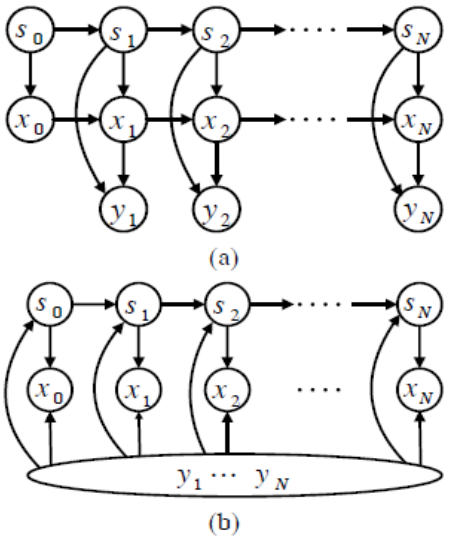
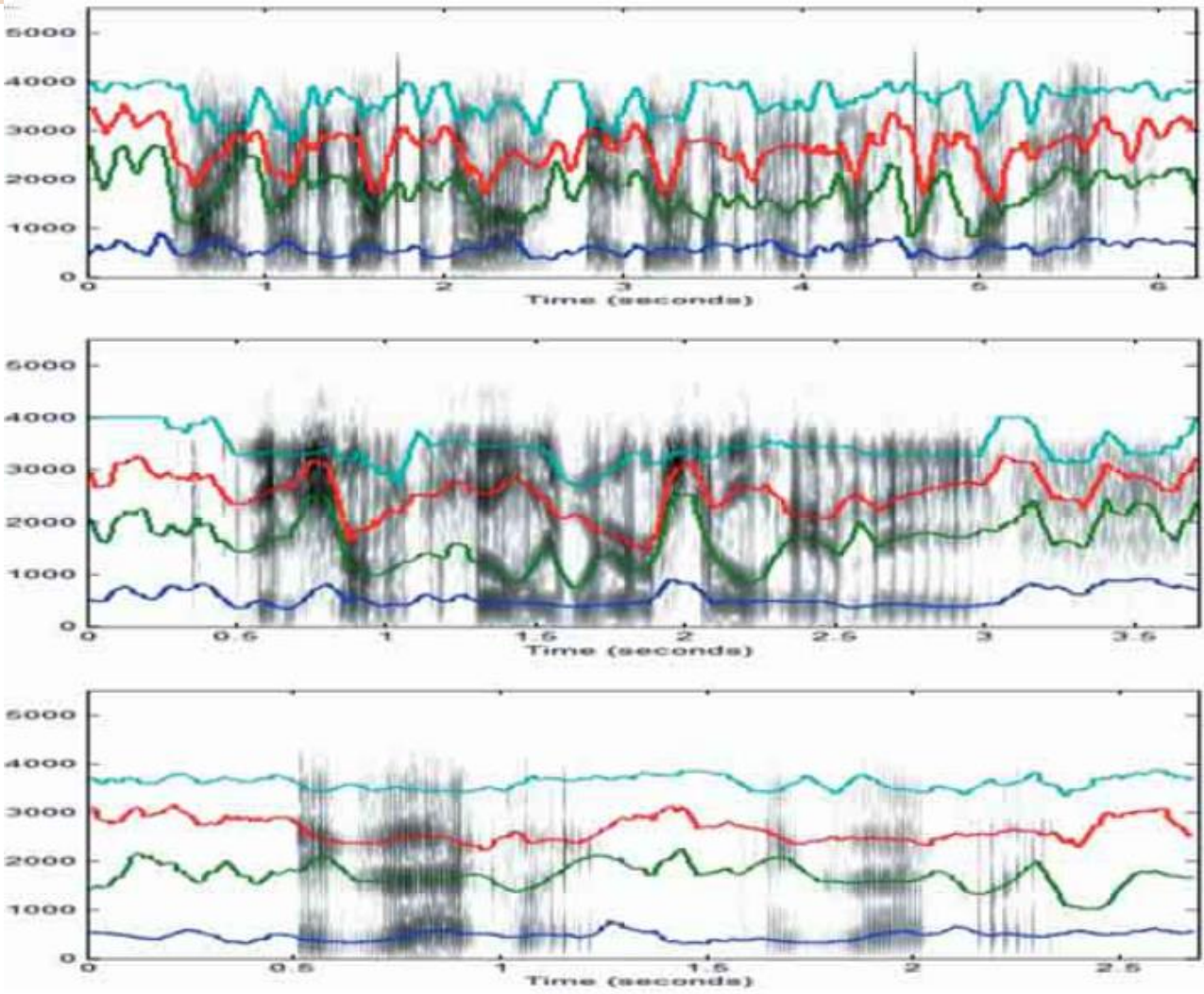


Fig. 1. The model (a) and the variational posterior (b) represented as Bayesian networks.

Surprisingly Good Inference Results for Continuous Hidden States

- By-product: accurately tracking dynamics of resonances (formants) in vocal tract (TIMIT & SWBD).
- Best formant tracker by then in speech analysis; used as basis to form a formant database as "ground truth"
- We thought we solved the ASR problem, except
- "Intractable" for decoding



Deng & Huang, Challenges in Adopting Speech Recognition, *Communications of the ACM*, vol. 47, pp. 69-75, 2004.
Deng, Cui, Pruvenok, Huang, Momen, Chen, Alwan, A Database of Vocal Tract Resonance Trajectories for Research in Speech , ICASSP, 2006.

Deep Generative Models in Speech Recognition

(prior to the rising of deep learning)

Segment & Nonstationary-State Models

- Digalakis, Rohlicek, Ostendorf. "ML estimation of a stochastic linear system with the EM alg & application to speech recognition," IEEE T-SAP, 1993 — 1993
- Deng, Aksmanovic, Sun, Wu, Speech recognition using HMM with polynomial regression functions as nonstationary states," IEEE T-SAP, 1994. — 1994

Hidden Dynamic Models (HDM)

- Deng, Ramsay, Sun. "Production models as a structural basis for automatic speech recognition," Speech Communication, vol. 33, pp. 93–111, 1997. — 1997
- Bridle et al. "An investigation of segmental hidden dynamic models of speech coarticulation for speech recognition," Final Report Workshop on Language Engineering, Johns Hopkins U, 1998. — 1998
- Picone et al. "Initial evaluation of hidden dynamic models on conversational speech," ICASSP, 1999. — 1999
- Deng and Ma. "Spontaneous speech recognition using a statistical co-articulatory model for the vocal tract resonance dynamics," JASA, 2000. — 2000

Structured Hidden Trajectory Models (HTM)

- Zhou, et al. "Coarticulation modeling by embedding a target-directed hidden trajectory model into HMM," ICASSP, 2003. ← **DARPA EARS Program 2001-2004: Novel Approach II** — 2003
- Deng, Yu, Acero. "Structured speech modeling," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 14, no. 5, 2006. — 2006

Switching Nonlinear State-Space Models

- Deng. "Switching Dynamic System Models for Speech Articulation and Acoustics," in Mathematical Foundations of Speech and Language Processing, vol. 138, pp. 115 - 134, Springer, 2003.
- Lee et al. "A Multimodal Variational Approach to Learning and Inference in Switching State Space Models," ICASSP, 2004.

Other Deep Generative Models

(developed outside speech)

- Sigmoid belief nets & wake/sleep alg. (1992)
- Deep belief nets (DBN, 2006);

→ Start of deep learning

- Totally non-obvious result:
  Stacking many RBMs (undirected)

→ not Deep Boltzmann Machine (**DBM**, undirected)

→ but a DBN (directed, generative model)

- Excellent in generating images & speech synthesis

- Similar type of deep generative models to HDM
- But simpler: no temporal dynamics
- With very different parameterization
- Most intriguing of DBN: inference is easy
  (i.e. no need for approximate variational Bayes)
  ← "Restriction" of connections in RBM

- Pros/cons analysis→ Hinton coming to MSR 2009

LETTER Communicated by Yann Le Cun

A Fast Learning Algorithm for Deep Belief Nets

Geoffrey E. Hinton
hinton@cs.toronto.edu
Simon Osindero
osindero@cs.toronto.edu
Department of Computer Science, University of Toronto, Toronto, Canada M5S 3G4

Yee-Whye Teh
tehyw@comp.nus.edu.sg
Department of Computer Science, National University of Singapore, Singapore 117543

We show how to use "complementary priors" to eliminate the explaining-away effects that make inference difficult in densely connected belief nets that have many hidden layers. Using complementary priors, we derive a fast, greedy algorithm that can learn deep, directed belief networks one layer at a time, provided the top two layers form an undirected associative memory. The fast, greedy algorithm is used to initialize a slower learning procedure that fine-tunes the weights using a contrastive version of the wake-sleep algorithm. After fine-tuning, a network with three hidden layers forms a very good generative model of the joint distribution of handwritten digit images and their labels. This generative model gives better digit classification than the best discriminative learning algorithms. The low-dimensional manifolds on which the digits lie are modeled by long ravines in the free-energy landscape of the top-level associative memory, and it is easy to explore these ravines by using the directed connections to display what the associative memory has in mind.

This is a very different kind of deep generative model

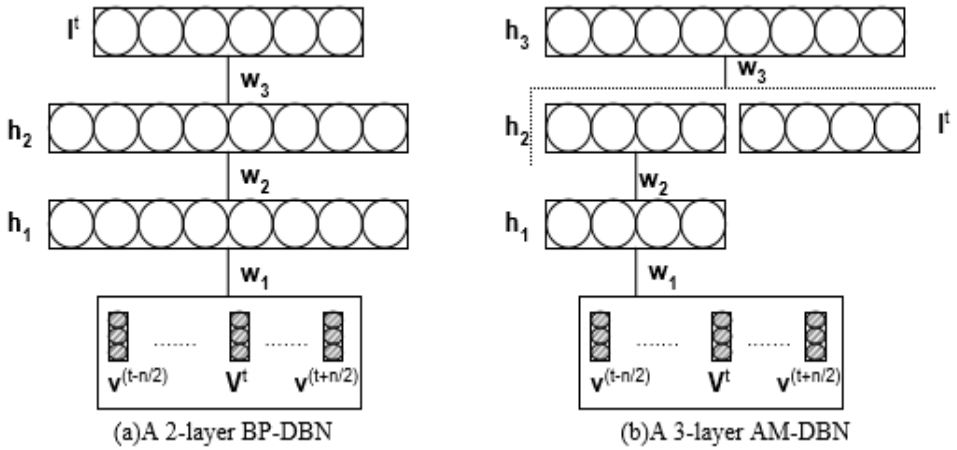
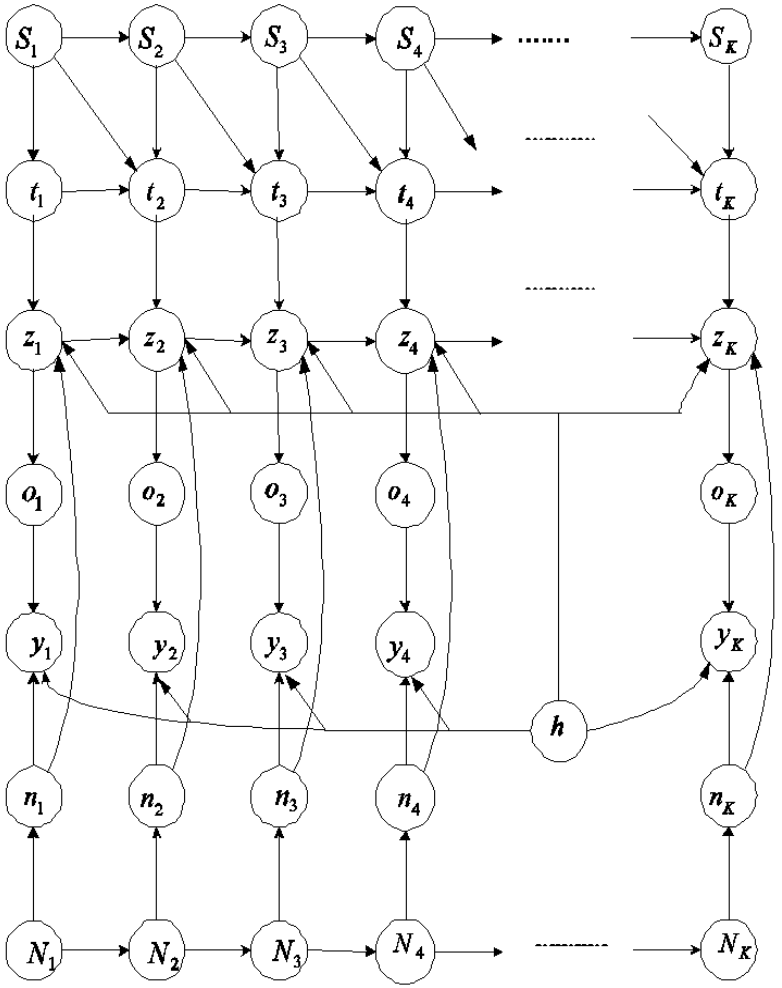


Table 4: *Reported results on TIMIT core test set*

Method	PER
Stochastic Segmental Models [28]	36%
Conditional Random Field [29]	34.8%
Large-Margin GMM [30]	33%
CD-HMM [4]	27.3%
Augmented conditional Random Fields [4]	26.6%
Recurrent Neural Nets [31]	26.1%
Bayesian Triphone HMM [32]	25.6%
Monophone HTMs [33]	24.8%
Heterogeneous Classifiers [34]	24.40%
Deep Belief Networks(DBNs)	**23.0%**

(Mohamed, Dahl, Hinton, 2009, 2012)

(after adding Backprop to the generative DBN)



(Deng et al., 2006; Deng & Yu, 2007)

IEEE TRANSACTIONS ON AUDIO, SPEECH, AND LANGUAGE PROCESSING, VOL. 14, NO. 5, SEPTEMBER 20

Structured Speech Modeling

Li Deng, *Fellow, IEEE*, Dong Yu, *Member, IEEE*, and Alex Acero, *Fellow, IEEE*

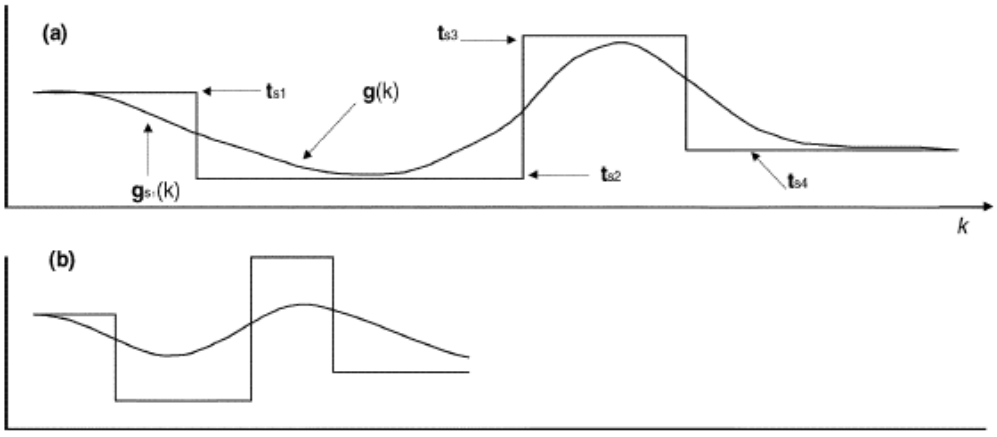


Fig. 1. Illustrations of the various VTR quantities in model Stage-I in utterance with four phone segments. (a) and (b) are for the same four VT targets and their filtered results, but the durations of the four segments a shorter in (b) than in (a).

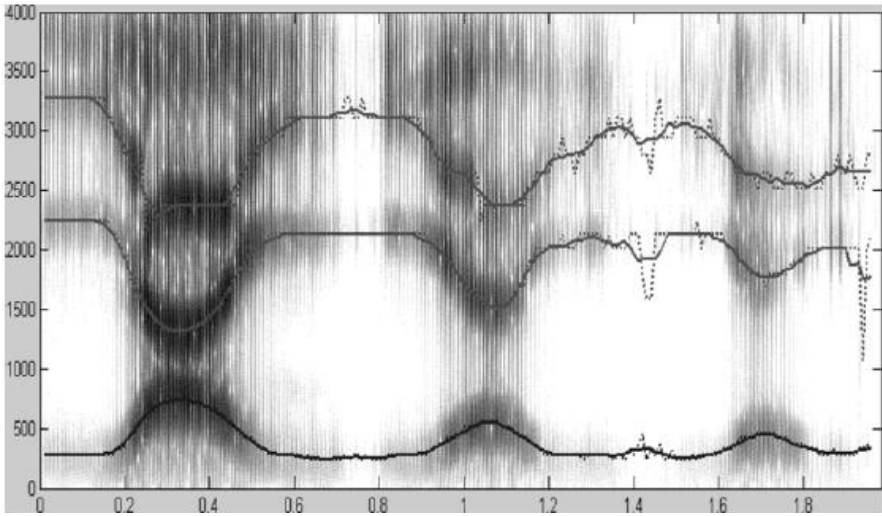


TABLE II
COMPARISONS OF HMM AND HTM PERFORMANCES (PERCENT CORRE
WITHIN EACH OF FOUR BROAD PHONE CLASSES

	Sonorants	Stops	Fricatives	Closures
Occurrences	3814	889	1252	1578
HMM	64.05	72.10	75.64	88.72
HTM	72.42	76.27	75.74	90.94

- Elegant model formulation & knowledge incorporation
- Strong empirical results: **96%** TIMIT accuracy with Nbest=1001; **75.2%** lattice decoding w. monophones; fast approx. training
- Still very expensive for decoding; could not ship (very frustrating!)

-- DBN/DNN made many new errors on short, undershoot vowels

-- 11 frames contain too much "noise"

Early Successes of Deep Learning in Speech Recognition

Academic-Industrial Collaboration (2009,2010)

- I invited Geoff Hinton to work with me at MSR, Redmond
- **Well-timed** academic-industrial collaboration:
  - ASR industry searching for new solutions when "principled" deep generative approaches could not deliver
  - Academia developed deep learning tools (e.g. **DBN** 2006) looking for applications
  - Add Backprop to deep generative models (DBN) → DNN (hybrid generative/discriminative)
  - Advent of GPU computing (Nvidia CUDA library released 2007/08)
  - Big training data in speech recognition were already available

- NIPS Home
- Overview
- Conference Videos
- Workshop Videos
- Program Highlights
- Tutorials
- Conference Sessions
- Workshops
- Publication Models
- Demonstrations
- Mini Symposia
- Accepted Papers
- Dates
- Committees
- Sponsors
- Awards
- Board

Li Deng, Dong Yu, Geoffrey Hinton

Microsoft Research; Microsoft Research; University of Toronto

Deep Learning for Speech Recognition and Related Applications

7:30am - 6:30pm Saturday, December 12, 2009

Location: Hilton: Cheakamus

Abstract: Over the past 25 years or so, speech recognition technology has been dominated by a "shallow" architecture --- hidden Markov models (HMMs). Significant technological success has been achieved using complex and carefully engineered variants

Invitee 1: give me one week to decide …,…

Not worth my time to fly to Vancouver for this…

there has been virtually no effective communication between machine learning researchers and speech recognition researchers who are both advocating the use of deep architecture and learning. One goal of the proposed workshop is to bring together these two groups of researchers to review the progress in both fields and to identify promising and synergistic research directions for potential future cross-fertilization and collaboration.

Mohamed, Dahl, Hinton, Deep belief networks for phone recognition, *NIPS 2009 Workshop on Deep Learning,* 2009

Yu, Deng, Wang, Learning in the Deep-Structured Conditional Random Fields, *NIPS 2009 Workshop on Deep Learning,* 2009

…, …, …

Expanding DNN at Industry Scale

- Scale DNN's success to large speech tasks (2010-2011)
  - Grew output neurons from context-independent phone states (100-200) to context-dependent ones (1k-30k) → CD-DNN-HMM for Bing Voice Search and then to SWBD tasks
  - Motivated initially by saving huge MSFT investment in the speech decoder software infrastructure
  - CD-DNN-HMM also gave much higher accuracy than CI-DNN-HMM
  - Earlier NNs made use of context only as appended inputs, not coded directly as outputs
  - Discovered that ***with large training data Backprop works well without DBN pre-training*** by understanding why gradients often vanish (patent filed for "discriminative pre-training" 2011)

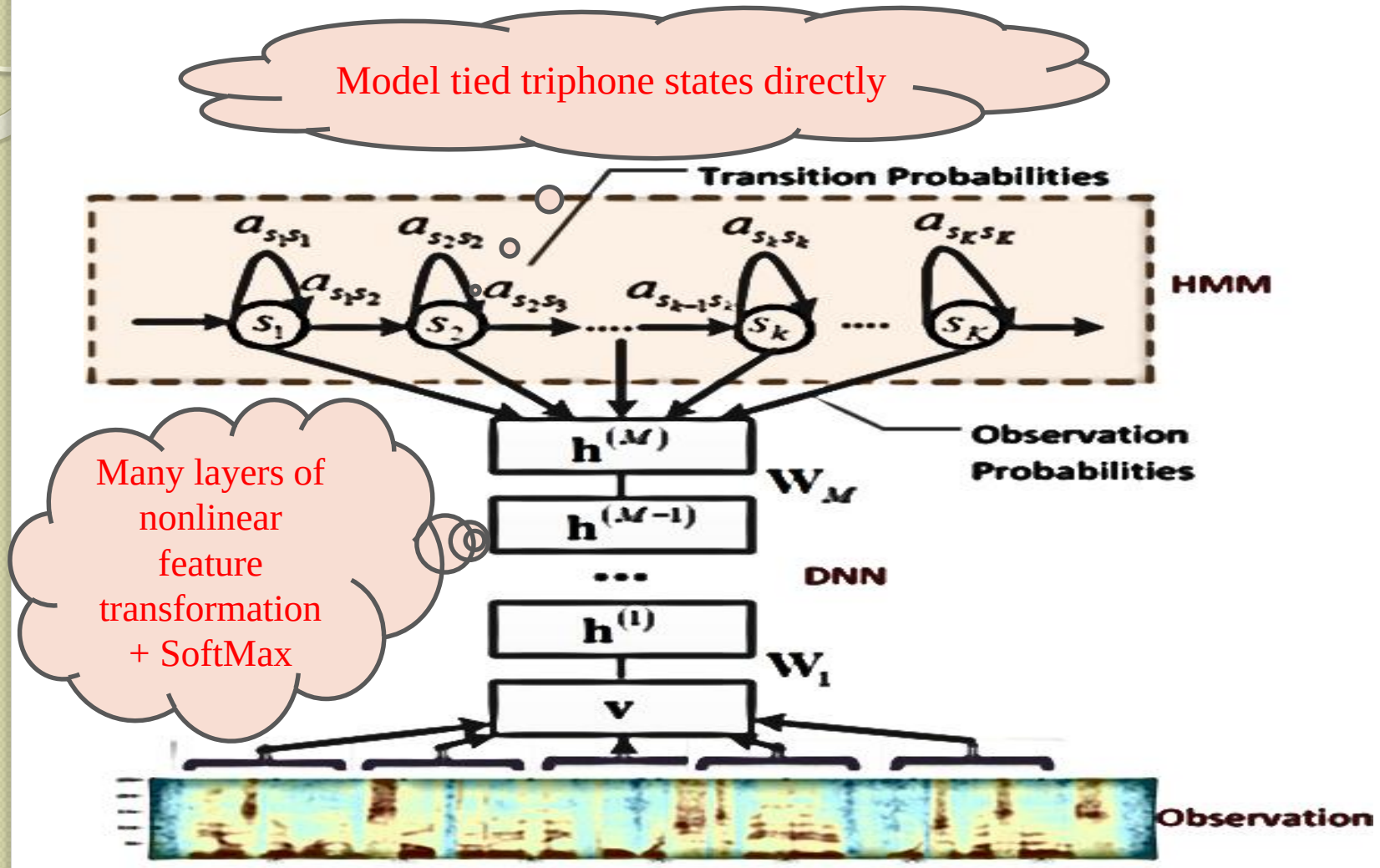
- Engineering for large speech systems:
  - Combined expertise in DNN (esp. with GPU implementation) **and** speech recognition
  - Collaborations among MSRR, MSRA, academic researchers

- Yu, Deng, Dahl, Roles of Pre-Training and Fine-Tuning in Context-Dependent DBN-HMMs for Real-World Speech Recognition, in *NIPS Workshop on Deep Learning*, 2010.
- Dahl, Yu, Deng, Acero, Large Vocabulary Continuous Speech Recognition With Context-Dependent DBN-HMMS, in *Proc. ICASSP, 2011.*
- Dahl, Yu, Deng, Acero, Context-Dependent Pre-trained Deep Neural Networks for Large Vocabulary Speech Recognition, in *IEEE Transactions on Audio, Speech, and Language Processing (2013 IEEE SPS Best Paper Award)* , vol. 20, no. 1, pp. 30-42, January 2012.
- Seide, Li, Yu, "Conversational Speech Transcription Using Context-Dependent Deep Neural Networks", Interspeech 2011, pp. 437-440.
- Hinton, Deng, Yu, Dahl, Mohamed, Jaitly, Senior, Vanhoucke, Nguyen, Sainath, Kingsbury, Deep Neural Networks for Acoustic Modeling in SpeechRecognition, in *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82-97, November 2012
- *Sainath, T., Kingsbury, B., Ramabhadran, B., Novak, P., and Mohamed, A. "Making deep belief networks effective for large vocabulary continuous speech recognition," Proc. ASRU, 2011.*
- *Sainath, T., Kingsbury, B., Soltau, H., and Ramabhadran, B. "Optimization Techniques to Improve Training Speed of Deep Neural Networks for Large Speech Tasks," IEEE Transactions on Audio, Speech, and Language Processing, vol.21, no.11, pp.2267-2276, Nov. 2013.*
- Jaitly, N., Nguyen, P., Senior, A., and Vanhoucke, V. "Application of Pretrained Deep Neural Networks to Large Vocabulary Speech Recognition," Proc. Interspeech, 2012.

Microsoft Research

DNN-HMM

(replacing GMM only; longer MFCC/filter-back windows w. no transformation)



DNN vs. Pre-DNN Prior-Art

- **Table:** TIMIT Phone recognition (3 hours of training)

Features	Setup	Error Rates
Pre-DNN	Deep Generative Model	24.8%
DNN	5 layers x 2048	23.4%

~10% relative improvement

- **Table:** Voice Search SER (24-48 hours of training)

Features	Setup	Error Rates
Pre-DNN	GMM-HMM with MPE	36.2%
DNN	5 layers x 2048	30.1%

~20% relative improvement

- **Table:** SwitchBoard WER (309 hours training)

Features	Setup	Error Rates
Pre-DNN	GMM-HMM with BMMI	23.6%
DNN	7 layers x 2048	15.8%

~30% relative Improvement

For DNN, the more data, the better!

Microsoft Research

Deep learning technology enabled speech-to-speech translation

The New York Times

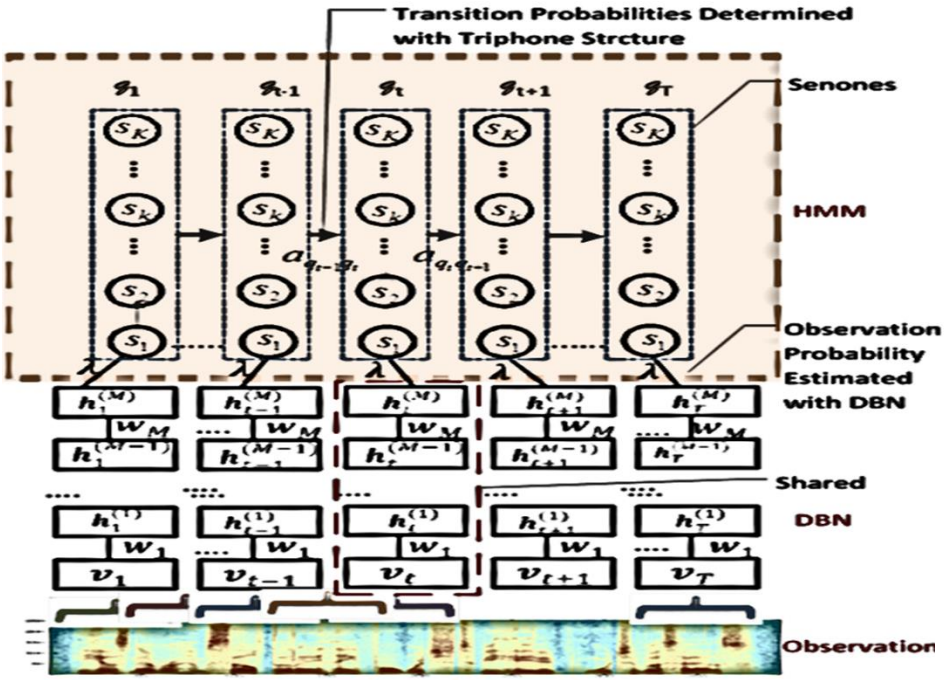
Scientists See Promise in Deep-Learning Programs

John Markoff
November 23, 2012

Rick Rashid in **Tianjin, China**, October, 25, 2012

A voice recognition program translated a speech given by Richard F. Rashid, Microsoft's top scientist, into Mandarin Chinese.

Microsoft Research

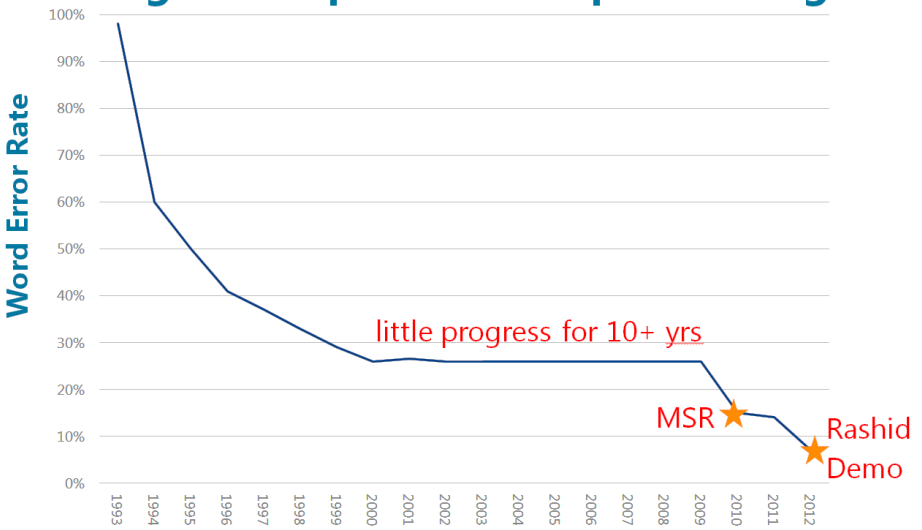


CD-DNN-HMM

Dahl, Yu, Deng, and Acero, "Context-Dependent Pre-trained Deep Neural Networks for Large Vocabulary Speech Recognition," *IEEE Trans. ASLP,* Jan. 2012 (also ICASSP 2011)

Seide et al, Interspeech, 2011.

Progress of spontaneous speech recognition



After no improvement for 10+ years by the research community…
…MSR reduced error from **~23%** to **<13%** (and under 7% for Rick Rashid's S2S demo in 2012)!

Microsoft Research

Impact of deep learning in speech technology

Skype to get 'real-time' translator

MIT Technology Review

10 BREAKTHROUGH TECHNOLOGIES 2013

Deep Learning

With massive amounts of computational power, machines can now recognize objects and translate speech in real time. Artificial intelligence is finally getting smart.

Temporary Social Media

Messages that quickly self-destruct could enhance the privacy of online communications and make people freer to be spontaneous.

Prenatal DNA Sequencing

Reading the DNA of fetuses will be the next frontier of the genomic revolution. But do you really want to know about the genetic problems or musical aptitude of your unborn child?

Additive Manufacturing

Skeptical about 3-D printing? GE, the world's largest manufacturer, is on the verge of using the technology to make jet parts.

Baxter: The Blue-Collar Robot

Rodney Brooks's newest creation is easy to interact with, but the complex innovations behind the robot show just how hard it is to get along with people.

Memory Implants

A maverick neuroscientist believes he has deciphered the code by which the brain forms long-term memories. Next: testing a prosthetic implant for people suffering from long-term memory loss.

Smart Watches

The designers of the Pebble watch realized that a mobile phone is more useful if you don't have to take it out of your pocket.

Ultra-Efficient Solar Power

Doubling the efficiency of a solar cell would completely change the economics of renewable energy. Nanotechnology just might make it possible.

Big Data from Cheap Phones

Collecting and analyzing information from simple cell phones can provide surprising insights into how people move about and behave – and even help us understand the spread of diseases.

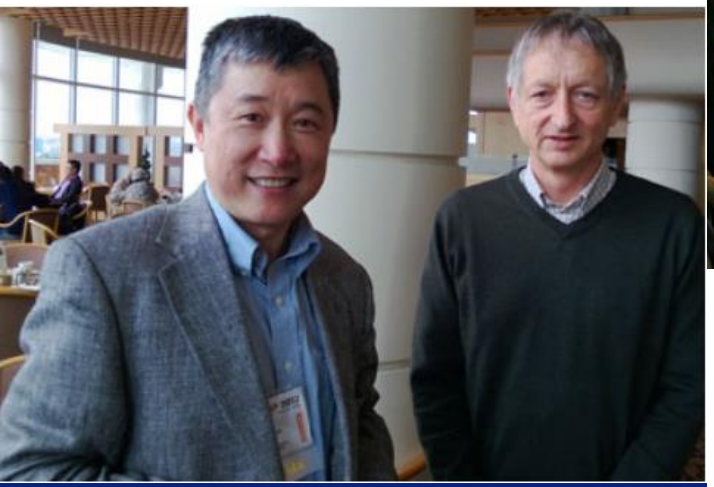
Supergrids

A new high-power circuit breaker could finally make highly efficient DC power grids practical.

Enabling Cross-Lingual Conversations in Real Time

Microsoft Research
May 27, 2014 5:58 PM PT

View milestones on the path to Skype Translator
#speech2speech

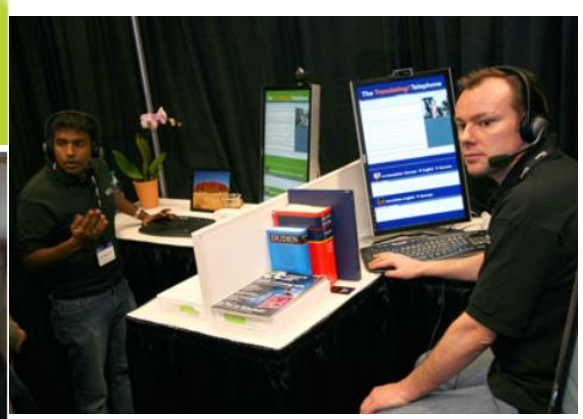


Microsoft Research

WIRED

ROBERT MCMILLAN BUSINESS 12.17.14 1:19 PM

HOW SKYPE USED AI TO BUILD ITS AMAZING NEW LANGUAGE TRANSLATOR



CNN

Taking a cue from science fiction, Microsoft demos 'universal translator'

By Jacopo Prisco, for CNN
Updated 12:35 PM ET, Thu October 16, 2014

Deep Learning in the News

The New York Tim

Monday, June 25, 2012 Last Update: 11:50 PM ET

EXCLUSIVE

Facebook, Google in 'Deep Learning' Arms Race

Yann LeCun, an NYU artificial intelligence researcher who now works for Facebook. *Photo: Josh Valcarcel/WIRED*

WIRED

NEWS BULLETIN

Google Beat Facebook for DeepMind

Google Acquires Artificial Intelligence Startup DeepMind For More Than $500M

Posted Jan 26, 2014 by Catherine Shu (@catherineshu)

(Slide from Bengio, 2015)

TECH 2/19/2015 @ 1:06PM | 6,586 views

Microsoft's Deep Learning Project Outperforms Humans In Image Recognition

+ Comment Now + Follow Comments

Anthony Wing Kosner
Contributor

FOLLOW

TECH 12/29/2014 @ 11:37AM | 75,350 views

Tech 2015: Deep Learning And Machine Intelligence Will Eat The World

+ Comment Now + Follow Comments

Despite what Stephen Hawking or Elon Musk say, hostile Artificial Intelligence is not going to destroy the world anytime soon. What is

In Academic World

nature International weekly journal of science

Deep learning

Yann LeCun, Yoshua Bengio & Geoffrey Hinton

Affiliations | Corresponding author

Nature **521**, 436–444 (28 May 2015) | doi:10.1038/nature14539

Received 25 February 2015 | Accepted 01 May 2015 | Published onli

IEEE Signal Processing MAGAZINE

[VOLUME 29 NUMBER 6 NOVEMBER 2012]

LOUD AND CLEAR

FUNDAMENTAL TECHNOLOGIES IN MODERN SPEECH RECOGNITION

Geoffrey Hinton, Li Deng, Dong Yu, George E. Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N. Sainath, and Brian Kingsbury

"This joint paper from the major speech recognition laboratories was the first major industrial application of deep learning."

Deep Neural Networks for Acoustic Modeling in Speech Recognition

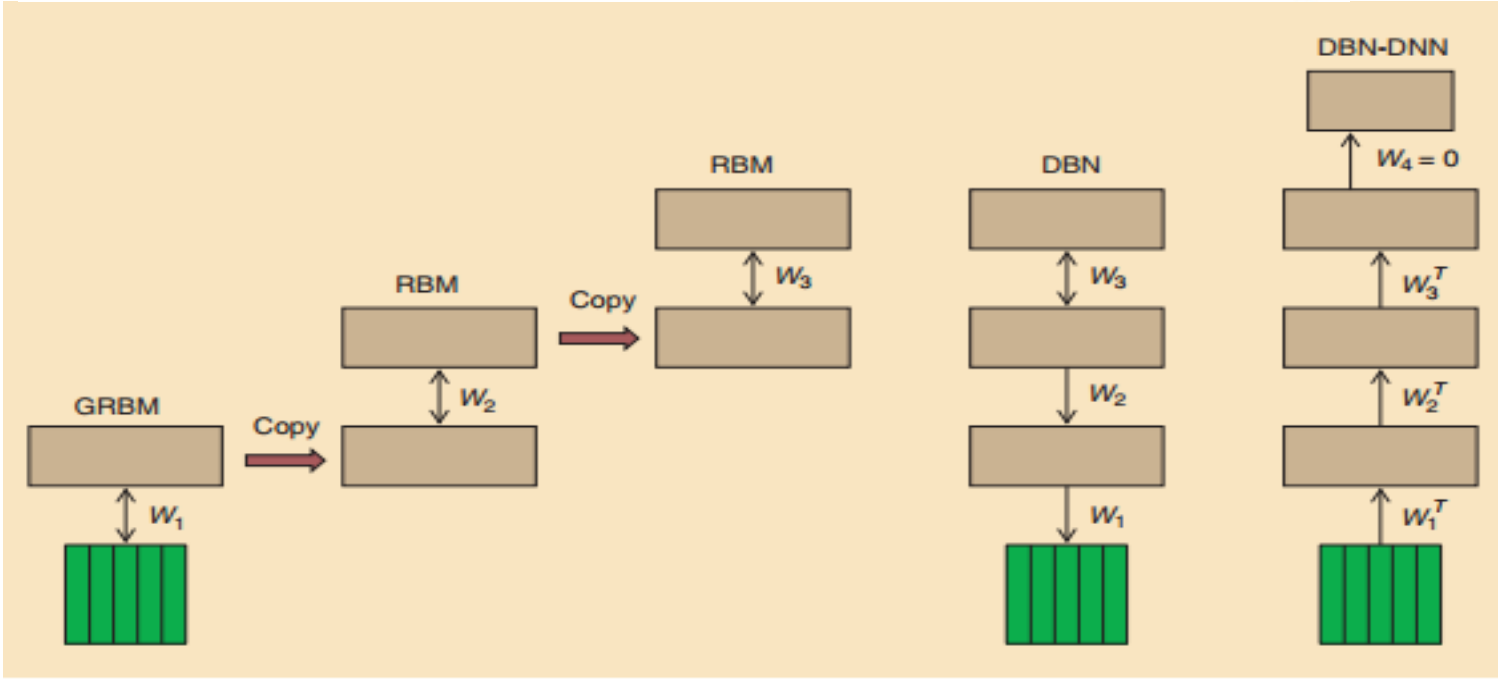
The shared views of four research groups

Microsoft Research

DNN: (Fully-Connected) Deep Neural Networks

"DNN for acoustic modeling in speech recognition," in *IEEE SPM*, Nov. 2012

Geoffrey Hinton, Li Deng, Dong Yu, George E. Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N. Sainath, and Brian Kingsbury



First train a stack of N models each of which has one hidden layer. Each model in the stack treats the hidden variables of the previous model as data.

Then compose them into a single Deep Belief Network (DBN).

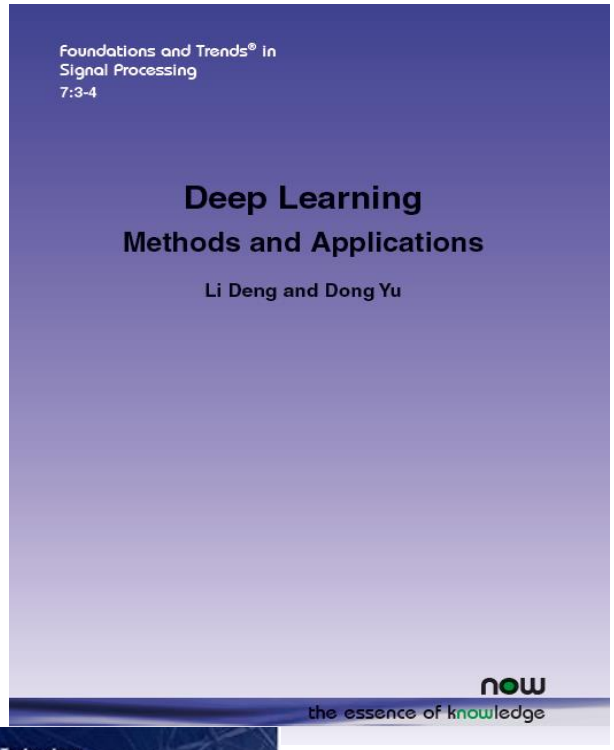
Then add outputs and train the DNN with backprop.

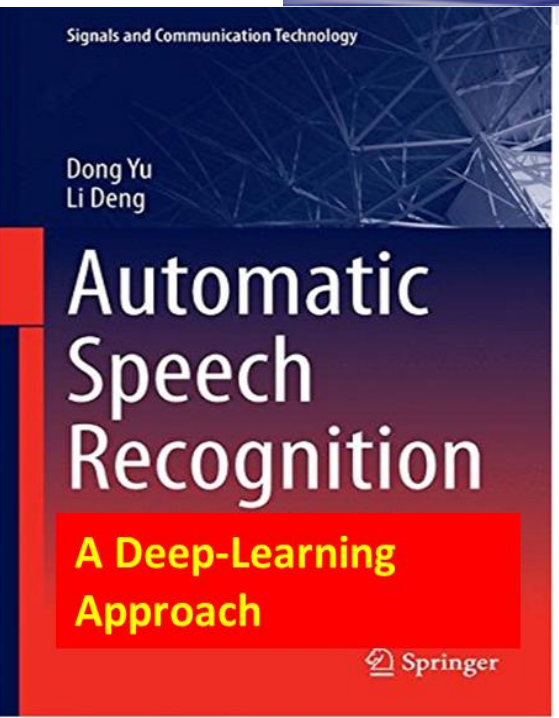
Practical guide to address some common deep learning ASR issues

ASR Issues	Solutions
How to reduce the runtime without accuracy loss?	SVD
How to do speaker adaptation with low footprints?	SVD-based adaptation
How to be robust to noise?	Variable component CNN
How to reduce accuracy gap between large and small DNN?	Teacher-student learning using output posterior
How to deal with large variety of data?	DNN factorization, mixed band training
How to enable languages with limited training data?	Multi-lingual DNN

(Slide from: Jinyu Li)

More Recent Development of Deep Learning for Speech





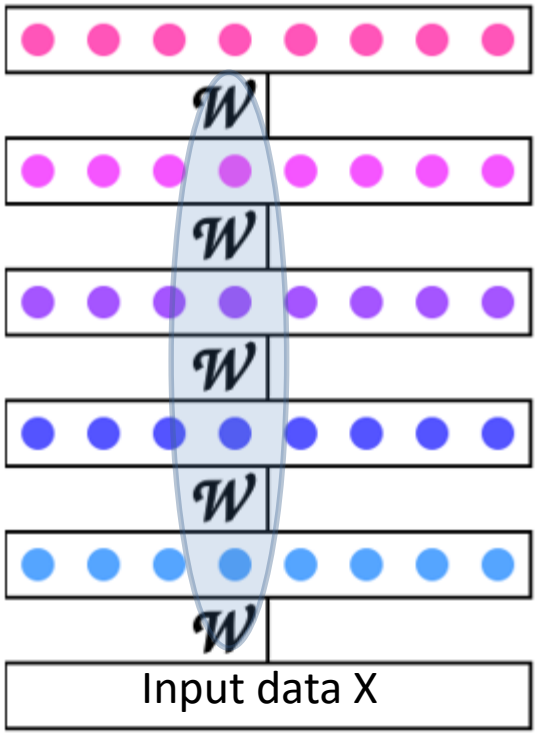
Chapter 7

Selected Applications in Speech and Audio Processing

7.1 Acoustic modeling for speech recognition

As discussed in Section 2, speech recognition is the very first successful application of deep learning methods at an industry scale. This success is a result of close academic-industrial collaboration, initiated at Microsoft Research, with the involved researchers identifying and acutely attending to the industrial need for large-scale deployment [68, 89, 109, 161, 323, 414]. It is also a result of carefully exploiting the strengths of the deep learning and the then-state-of-the-art speech recognition technology, including notably the highly efficient decoding techniques.

Innovation: Better Optimization



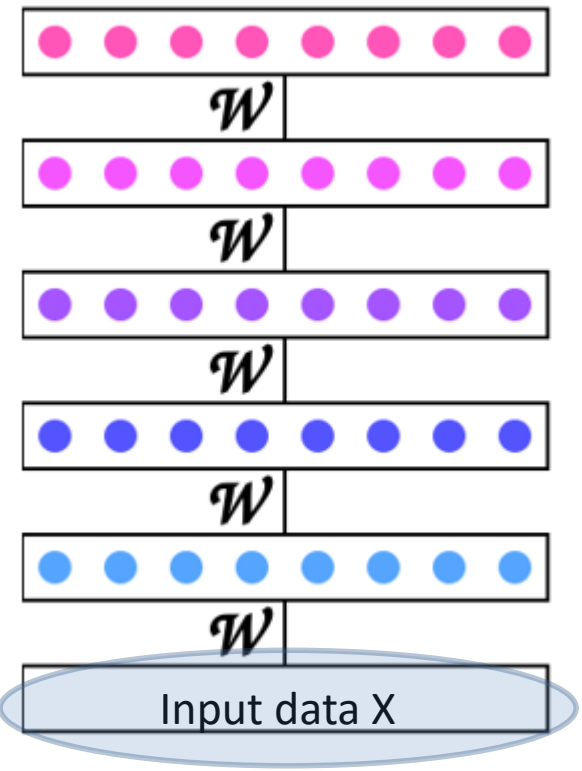
- **Sequence discriminative training for DNN:**

- Mohamed, Yu, Deng: "Investigation of full-sequence training of deep belief networks for speech recognition," Interspeech, 2010.
- Kingsbury, Sainath, Soltau. "Scalable minimum Bayes risk training of DNN acoustic models using distributed hessian-free optimization," Interspeech, 2012.
- Su, Li, Yu, Seide. "Error back propagation for sequence training of CD deep networks for conversational speech transcription," ICASSP, 2013.
- Vesely, Ghoshal, Burget, Povey. "Sequence-discriminative training of deep neural networks, Interspeech, 2013.

- **Distributed asynchronous SGD**

- Dean, Corrado,…Senior, Ng. "Large Scale Distributed Deep Networks," NIPS, 2012.
- Sak, Vinyals, Heigold, Senior, McDermott, Monga, Mao. "Sequence Discriminative Distributed Training of Long Short-Term Memory Recurrent Neural Networks," Interspeech,2014.

Innovation: Towards Raw Inputs



- **Bye-Bye MFCCs (no more cosine transform, Mel-scaling?)**

- Deng, Seltzer, Yu, Acero, Mohamed, Hinton. "Binary coding of speech spectrograms using a deep auto-encoder," **Interspeech, 2010**.
- Mohamed, Hinton, Penn. "Understanding how deep belief networks perform acoustic modeling," ICASSP, 2012.
- Li, Yu, Huang, Gong, "Improving wideband speech recognition using mixed-bandwidth training data in CD-DNN-HMM" SLT, 2012
- Deng, J. Li, Huang, Yao, Yu, Seide, Seltzer, Zweig, He, Williams, Gong, Acero. "Recent advances in deep learning for speech research at Microsoft," ICASSP, 2013.
- Sainath, Kingsbury, Mohamed, Ramabhadran. "Learning filter banks within a deep neural network framework," ASRU, 2013.

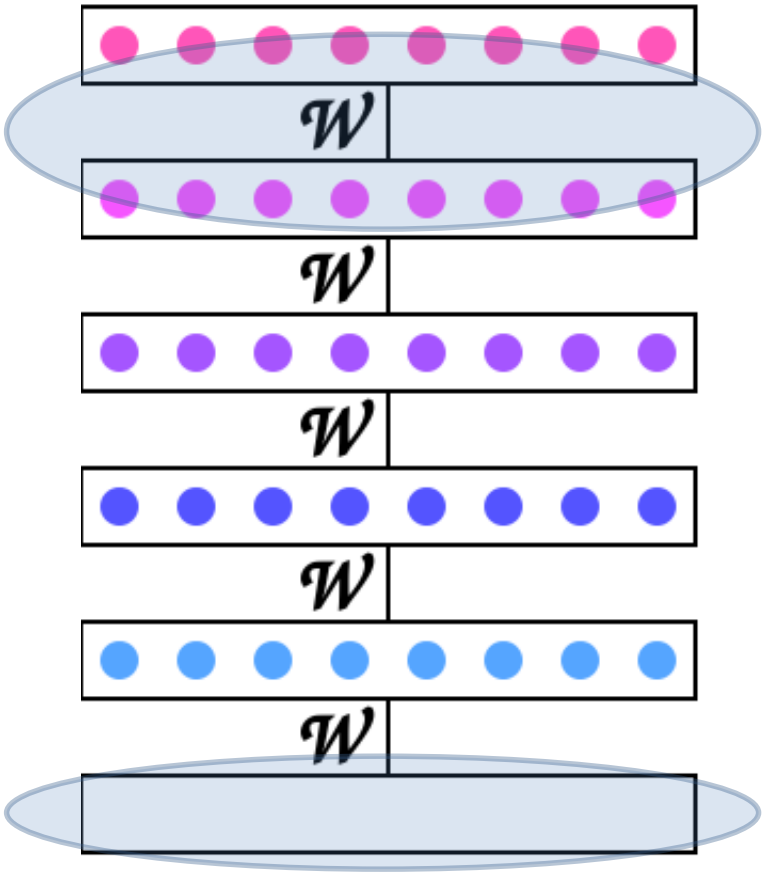
- **Bye-Bye Fourier transforms?**

- Jaitly and Hinton. "Learning a better representation of speech sound waves using RBMs," ICASSP, 2011.
- Tuske, Golik, Schluter, Ney. "Acoustic modeling with deep neural networks using raw time signal for LVCSR," Interspeech, 2014.
- Golik et al, "Convolutional NNs for acoustic modeling of raw time signals in LVCSR," Interspeech, 2015.
- Sainath et al. "Learning the Speech Front-End with Raw Waveform CLDNNs," Interspeech, 2015

- **DNN as hierarchical nonlinear feature extractors:**

- Seide, Li, Chen, Yu. "Feature engineering in context-dependent deep neural networks for conversational speech transcription, ASRU, 2011.
- Yu, Seltzer, Li, Huang, Seide. "Feature learning in deep neural networks - Studies on speech recognition tasks," ICLR, 2013.
- Yan, Huo, Xu. "A scalable approach to using DNN-derived in GMM-HMM based acoustic modeling in LVCSR," Interspeech, 2013.
- Deng, Chen. "Sequence classification using high-level features extracted from deep neural networks," ICASSP, 2014.

Innovation: Transfer/Multitask Learning & Adaptation



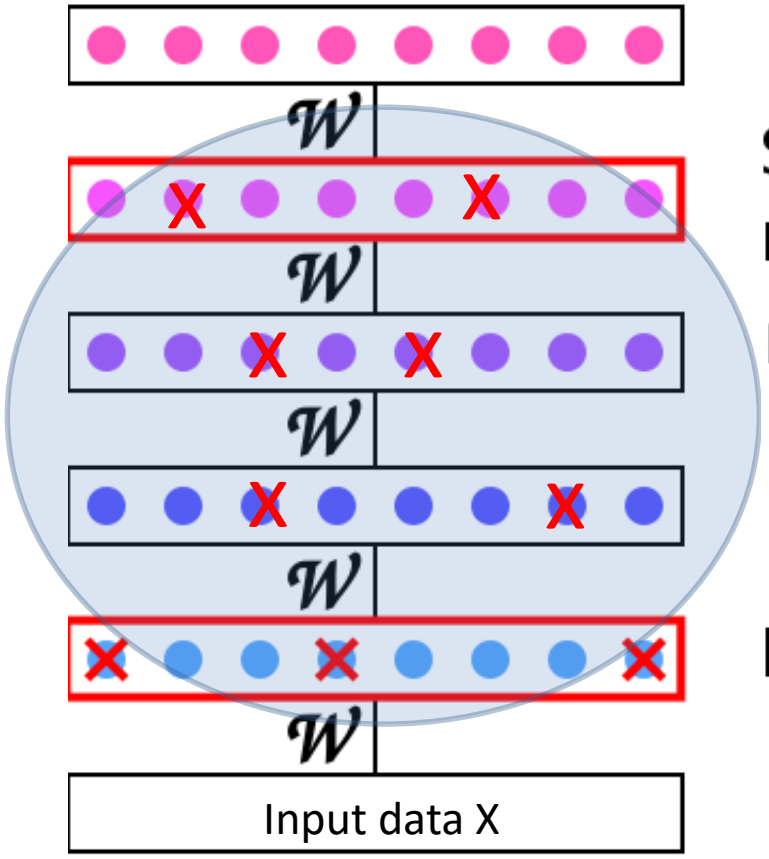
Multi-lingual acoustic modeling

Adaptation to speakers & environments (i-vectors)

Mixed-bandwidth acoustic modeling

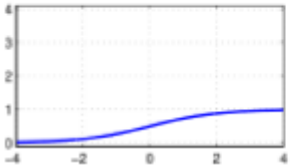
- Too many references to list & organize

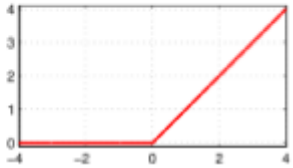
Innovation: Better regularization & **nonlinearity**



Sparsity in hidden representations

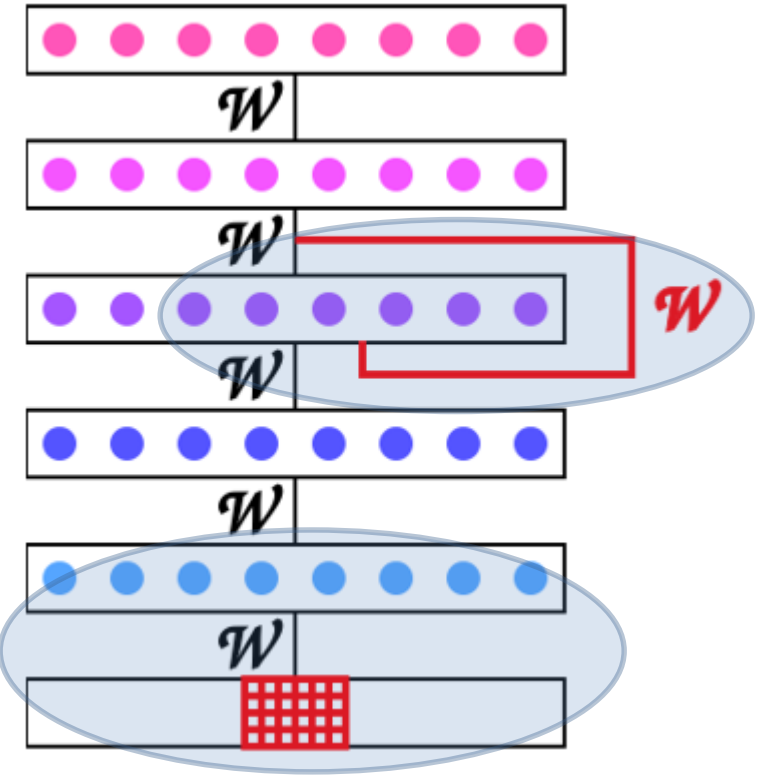
logistic ⟶ ReLU , MaxOut,





Dropout

Innovation: Better architectures



- **Recurrent Nets (bi-directional RNN/LSTM) and Conv Nets (CNN) are superior to fully-connected DNNs**
- Sak, Senior, Beaufays. "LSTM Recurrent Neural Network architectures for large scale acoustic modeling," Interspeech,2014.
- Soltau, Saon, Sainath. "Joint Training of Convolutional and Non-Convolutional Neural Networks," ICASSP, 2014.

Innovation: Ensemble Deep Learning

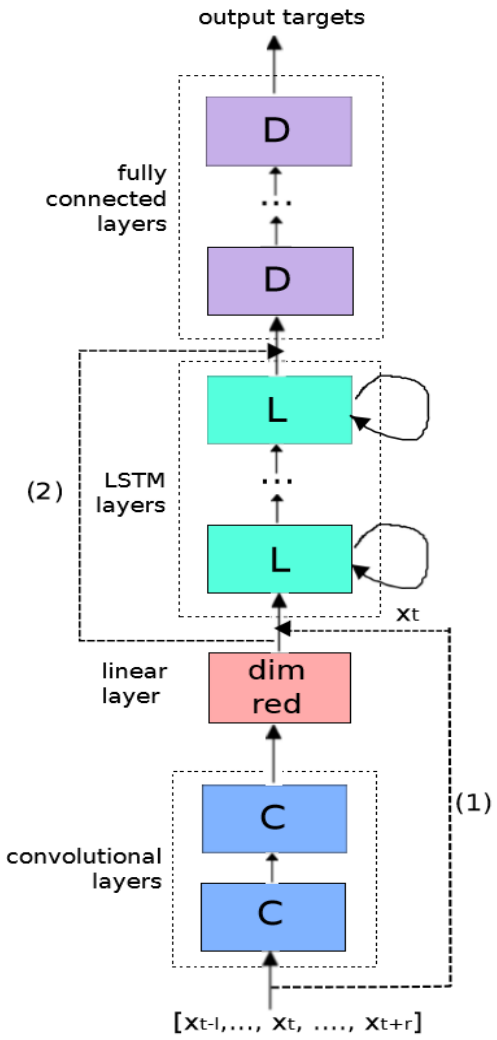
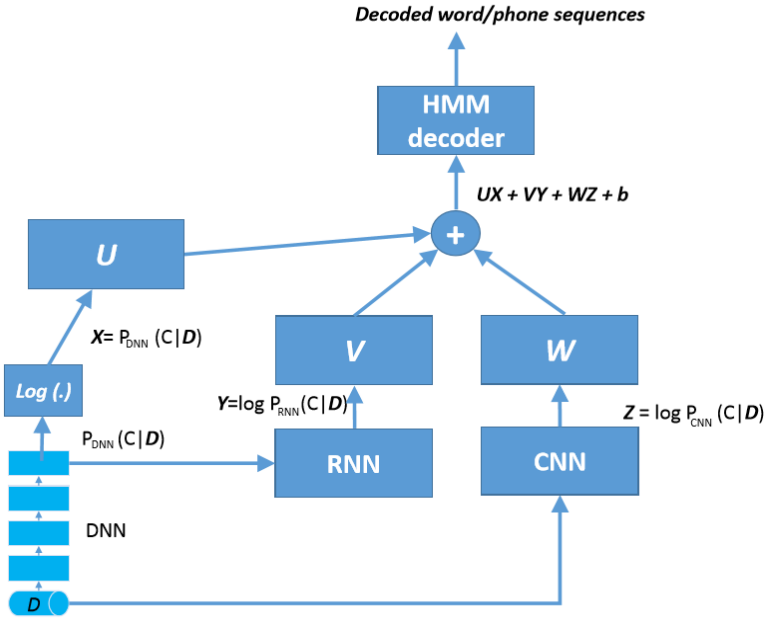
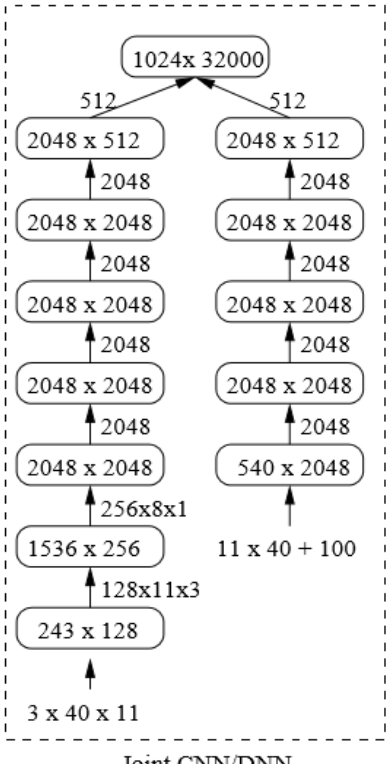


Fig. 1. CLDNN Architecture

- **Ensembles of RNN/LSTM, DNN, & Conv Nets (CNN) give huge gains:**
- T. Sainath, O. Vinyals, A. Senior, H. Sak. "Convolutional, Long Short-Term Memory, Fully Connected Deep Neural Networks," ICASSP 2015.
- L. Deng and John Platt, Ensemble Deep Learning for Speech Recognition, Interspeech, 2014.
- G. Saon, H. Kuo, S. Rennie, M. Picheny. "The IBM 2015 English conversational telephone speech recognition system," arXiv, May 2015. (8% WER on SWB-309h)





Joint CNN/DNN

Innovation: Better learning objectives/methods

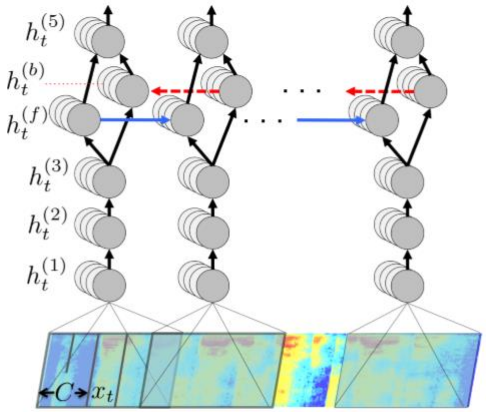
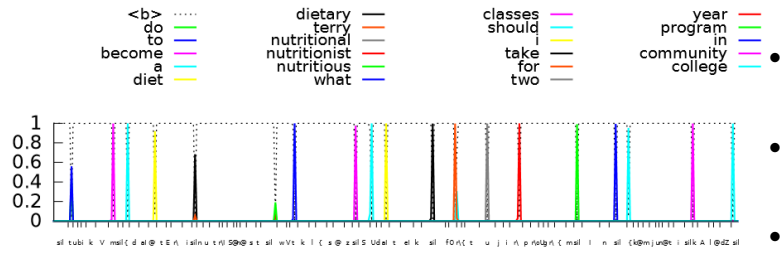


Figure 1: Structure of our RNN model and notation.



(b) 90k vocabulary

Figure 5: *'To become a dietary nutritionist what classes should I take for a two year program in a community college'*

- **Use of CTC as a new objective in RNN/LSTM with end2end learning drastically simplifies ASR systems**
- **Predict graphemes or words directly; no pron. dictionaries; no CD; no decision trees**
- **Use of "Blank" symbols may be equivalent to a special HMM state tying scheme**

➔ **CTC/RNN has NOT replaced HMM (left-to-right)**

- **Relative 8% gain by CTC has been shown by a very limited number of labs**

- **A. Graves** and N. Jaitly. "Towards End-to-End Speech Recognition with Recurrent Neural Networks," ICML, 2014.
- A. Hannun, A. Ng et al. "DeepSpeech: Scaling up End-to-End Speech Recognition," arXiv Nov. 2014.
- A. Maas et al. "Lexicon-Free Conversational ASR with NN," NAACL, 2015
- H. Sak et al. "Learning Acoustic Frame Labeling for ASR with RNN," ICASSP, 2015
- H. Sak, A. Senior, K. Rao, F. Beaufays. "Fast and Accurate Recurrent Neural Network Acoustic Models for Speech Recognition," Interspeech, 2015

Innovation: A new paradigm for speech recognition

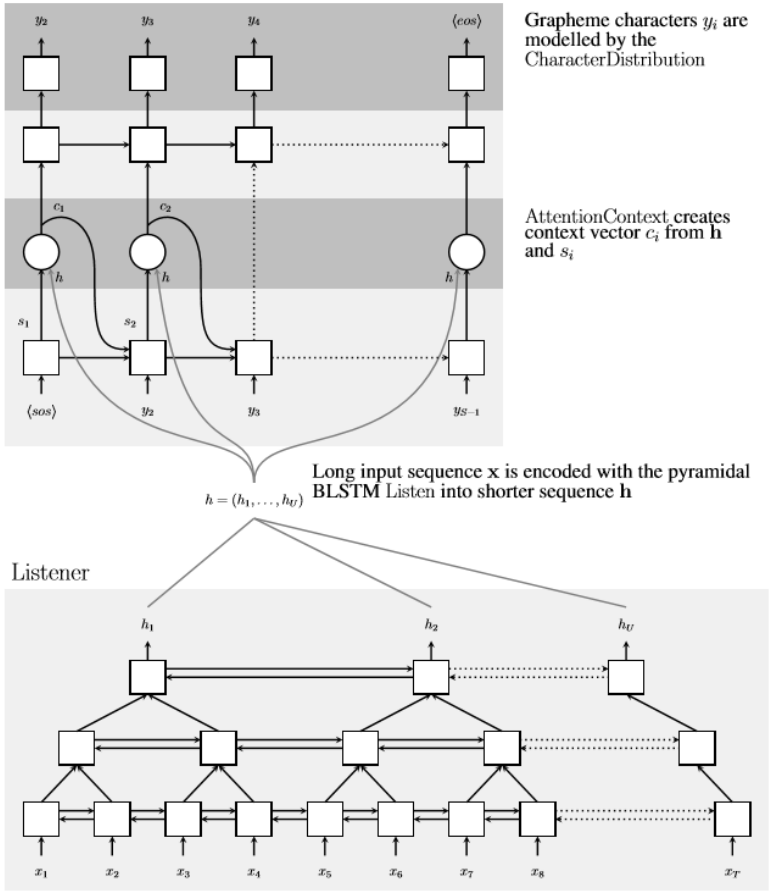
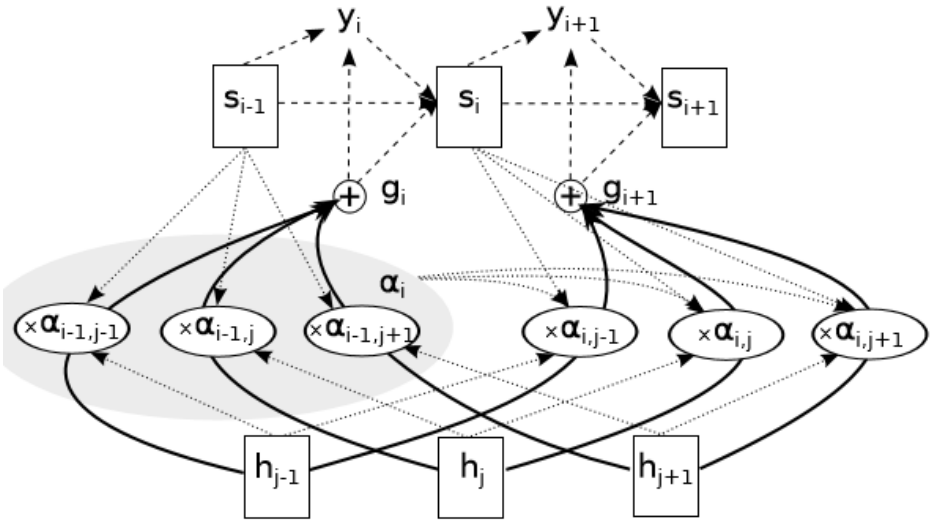


Figure 1: Listen, Attend and Spell (LAS) model: the listener is a pyramidal BLSTM encoding our input sequence x into high level features h, the speller is an attention-based decoder generating the y characters from h.

- **Seq2seq learning with attention mechanism (borrowed from NLP-MT)**

- W. Chan, N. Jaitly, Q. Le, O. Vinyals. "Listen, attend, and spell," arXiv, 2015.
- J. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, Y. Bengio. "Attention-Based Models for Speech Recognition," arXiv, 2015.



A Perspective on Recent Innovations of ASR

- All above deep learning innovations are based on supervised, discriminative learning of DNN and recurrent variants
- Capitalizing on big, **labeled** data
- Incorporating **monotonic-sequential** structure of speech (non-monotonic for language, later)
- Hard to incorporate many other aspects of speech knowledge with (e.g. speech distortion model)
- Hard to do semi- and **unsupervised** learning

➔Deep generative modeling may overcome such difficulties

Li Deng and Roberto Togneri, Chapter 6: Deep Dynamic Models for Learning Hidden Representations of Speech Features, pp. 153-196, Springer, December 2014.

Li Deng and Navdeep Jaitly, Chapter 2: Deep discriminative and generative models for pattern recognition, ~30 pages, in Handbook of Pattern Recognition and Computer Vision: 5th Edition, World Scientific Publishing, Jan 2016.

	Deep Neural Nets	Deep Generative Models
Structure	Graphical; info flow: **bottom-up**	Graphical; info flow: **top-down**
Incorp constraints & domain knowledge	Hard	Easy
Unsupervised	Harder or impossible	*Easier, at least possible*
Interpretation	Harder	Easier (generative "story" on data and hidden variables)
Representation	**Distributed**	Localist (mostly); can be distributed also
Inference/decode	Easy	Harder (but note recent progress)
Scalability/compute	**Easier (regular computes/GPU)**	Harder (but note **recent progress**)
Incorp. uncertainty	Hard	Easy
Empirical goal	Classification, feature learning, …	Classification (via Bayes rule), latent variable inference…
Terminology	Neurons, activation/gate functions, weights …	Random vars, stochastic "neurons", potential function, parameters …
Learning algorithm	A single, unchallenged, algorithm -- BackProp	A major focus of open research, many algorithms, & more to come
Evaluation	On a black-box score – end performance	On almost every intermediate quantity
Implementation	Hard (but increasingly easier)	Standardized but insights needed
Experiments	Massive, real data	Modest, often simulated data
Parameterization	Dense matrices	Sparse (often PDFs); can be dense

Example 1: Interpretable deep learning using deep topic models (NIPS-2015)

End-to-end Learning of Latent Dirichlet Allocation by Mirror-Descent Back Propagation

Jianshu Chen*, Ji He†, Yelong Shen*, Lin Xiao*, Xiaodong He*, Jianfeng Gao*, Xinying Song* and Li Deng*

*Microsoft Research, Redmond, WA 98052, USA,
{jianshuc,yeshen,lin.xiao,xiaohe,jfgao,xinson,deng}@microsoft.com
†Department of Electrical Engineering, University of Washington, Seattle, WA 98195, USA,
jvking@uw.edu

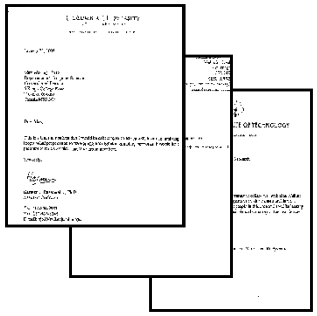
Abstract

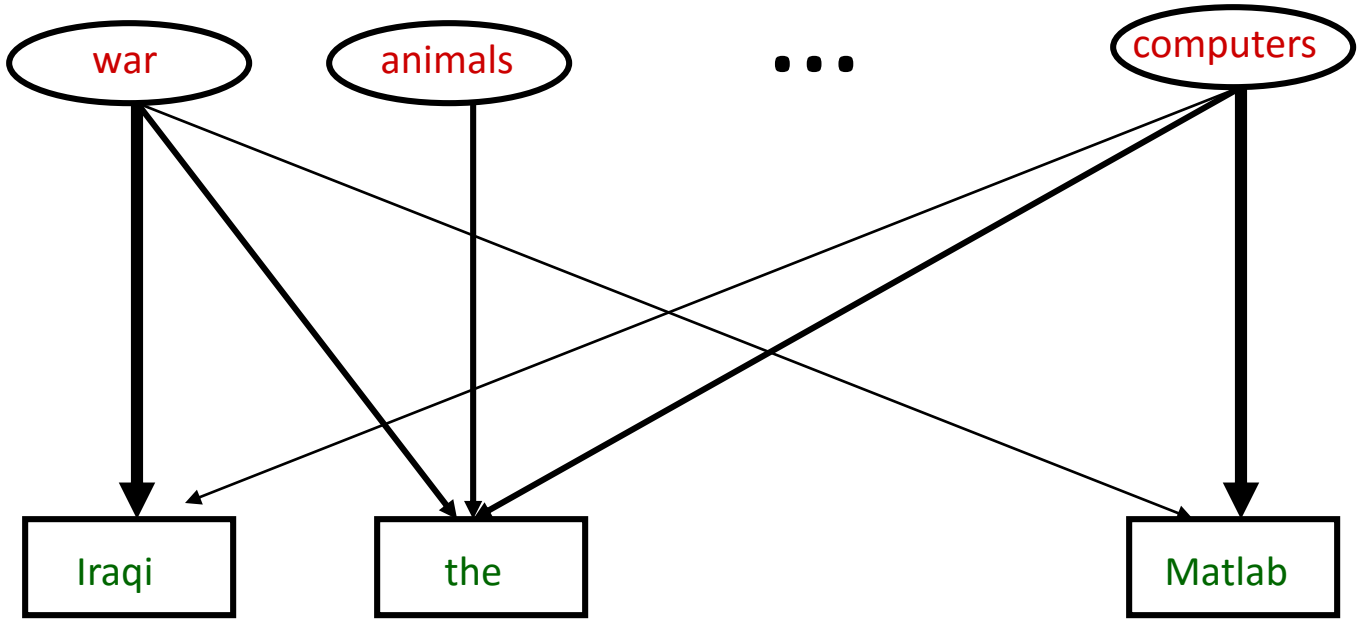
We develop a fully discriminative learning approach for supervised Latent Dirichlet Allocation (LDA) model, which maximizes the posterior probability of the prediction variable given the input document. Different from traditional variational learning or Gibbs sampling approaches, the proposed learning method applies (i) the mirror descent algorithm for exact maximum a posterior inference and (ii)

Recall: (Shallow) Generative Model



"TOPICS"
as hidden layer





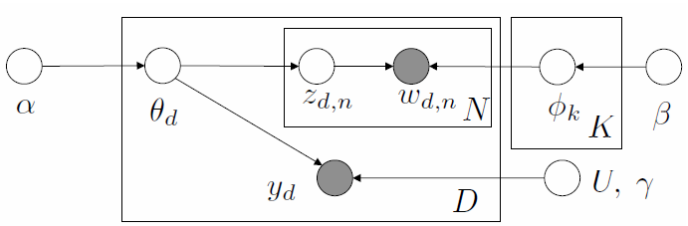
slide revised from: Max Welling

Interpretable deep learning: Deep generative model

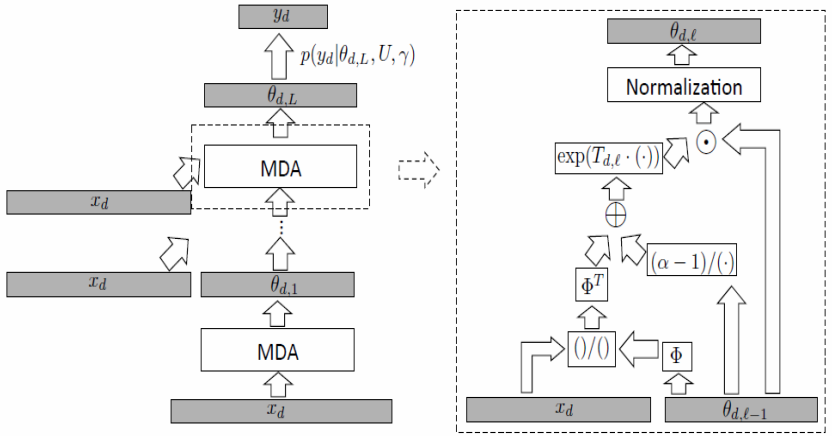
- Constructing interpretable DNNs based on generative topic models
- End-to-end learning by mirror-descent backpropagation

→ to maximize posterior probability p(y|x)

- y: output (win/loss), and x: input feature vector



Mirror Descent Algorithm (MDA)



$$\theta_{d,\ell} = \frac{1}{C_\theta} \cdot \theta_{d,\ell-1} \odot \exp\left(T_{d,\ell}\left[\Phi^T \frac{x_d}{\Phi\theta_{d,\ell-1}} + \frac{\alpha - \mathbb{1}}{\theta_{d,\ell-1}}\right]\right),\ \ell = 1, \ldots, L, \quad \theta_{d,0} = \frac{1}{K}\mathbb{1}$$

J. Chen, J. He, Y. Shen, L. Xiao, X. He, J. Gao, X. Song, and L. Deng, "End-to-end Learning of Latent Dirichlet Allocation by Mirror-Descent Back Propagation", submitted to NIPS2015.

Example 2: Unsupervised learning using deep generative model (ACL, 2013)

- Distorted character string Images→ Text
- Easier than unsupervised Speech→ Text
- 47% error reduction over Google's open-source OCR system

Unsupervised Transcription of Historical Documents

Taylor Berg-Kirkpatrick Greg Durrett Dan Klein
Computer Science Division
University of California at Berkeley
{tberg,gdurrett,klein}@cs.berkeley.edu

Abstract

We present a generative probabilistic model, inspired by historical printing processes, for transcribing images of documents from the printing press era. By jointly modeling the text of the document and the noisy (but regular) process of rendering glyphs, our unsupervised sys-

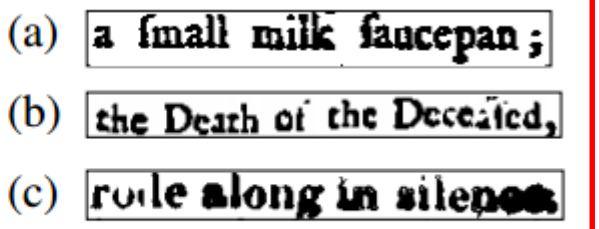


Figure 1: Portions of historical documents with (a) unknown font, (b) uneven baseline, and (c) over-inking.

Motivated me to think about **unsupervised ASR and NLP**

Power: **Character-level LM** & **generative modeling** for unsupervised learning

- "Image" data are naturally "generated" by the model quite accurately (like "computer graphics")
- I had the same idea for unsupervised generative Speech-to-Text in 90's
- Not successful because 1) Deep generative models were too simple for generating speech waves
  2) Inference/learning methods for deep generative models not mature then
  3) Computers were too slow

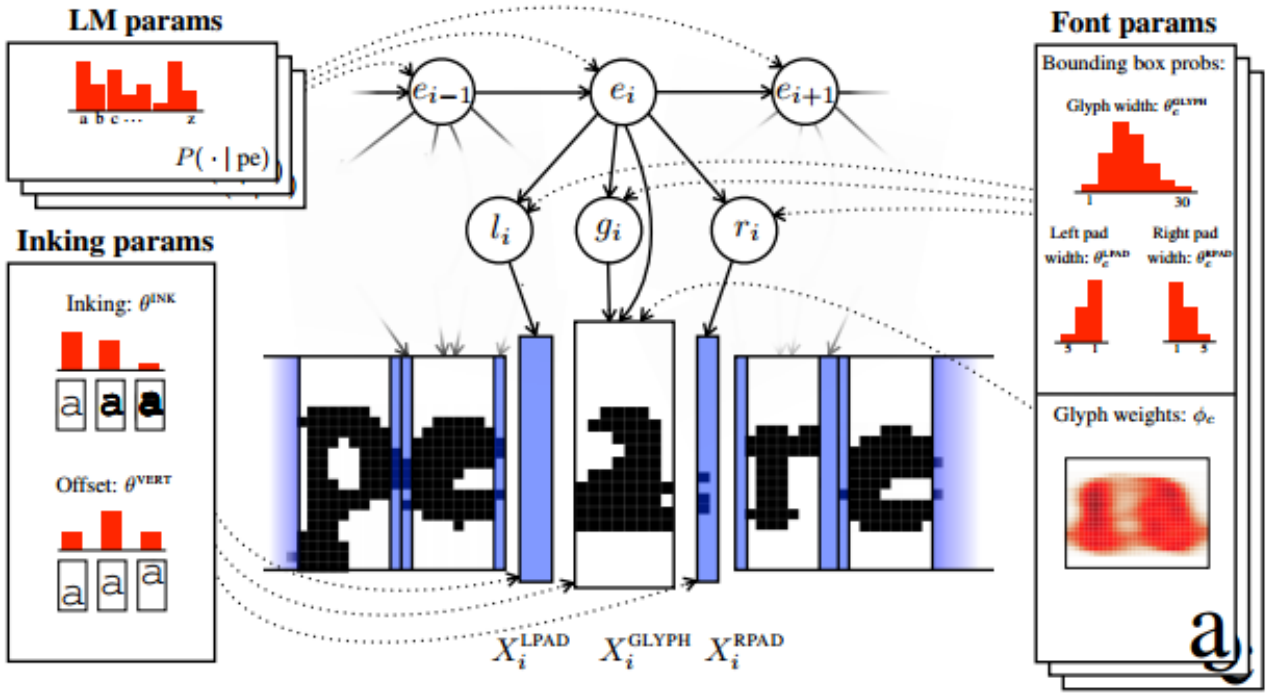
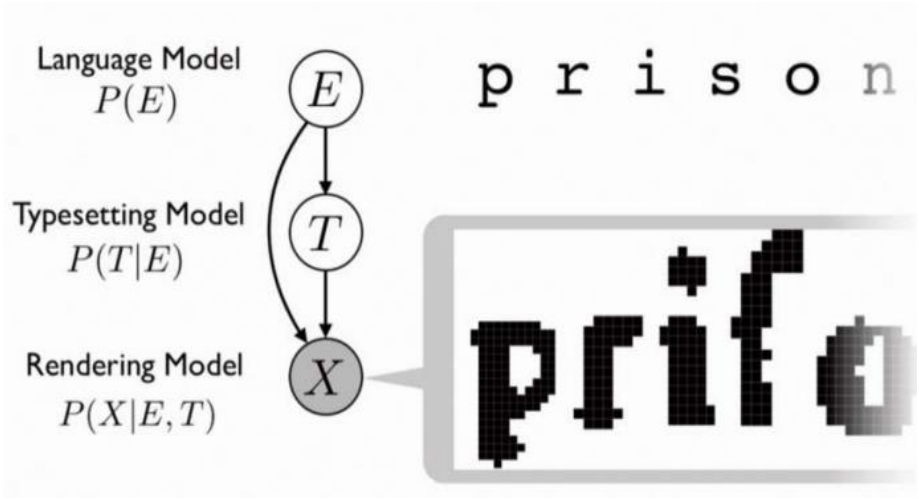


Figure 3: Character tokens e_i are generated by the language model. For each token index i, a glyph bounding box width g_i, left padding width l_i, and a right padding width r_i, are generated. Finally, the pixels in each glyph bounding box X_i^{GLYPH} are generated conditioned on the corresponding character, while the pixels in left and right padding bounding boxes, X_i^{LPAD} and X_i^{RPAD}, are generated from a background distribution.

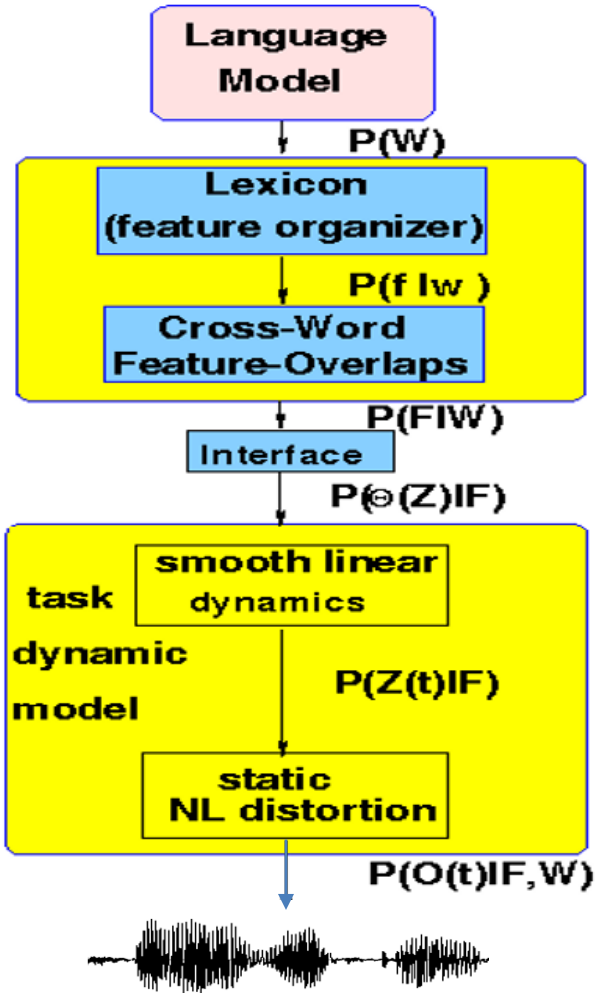
Deep Generative Model for Image-Text

(Berg-Kirkpatrick et al., 2013, 2015)



Deep Generative Model for Speech-Text

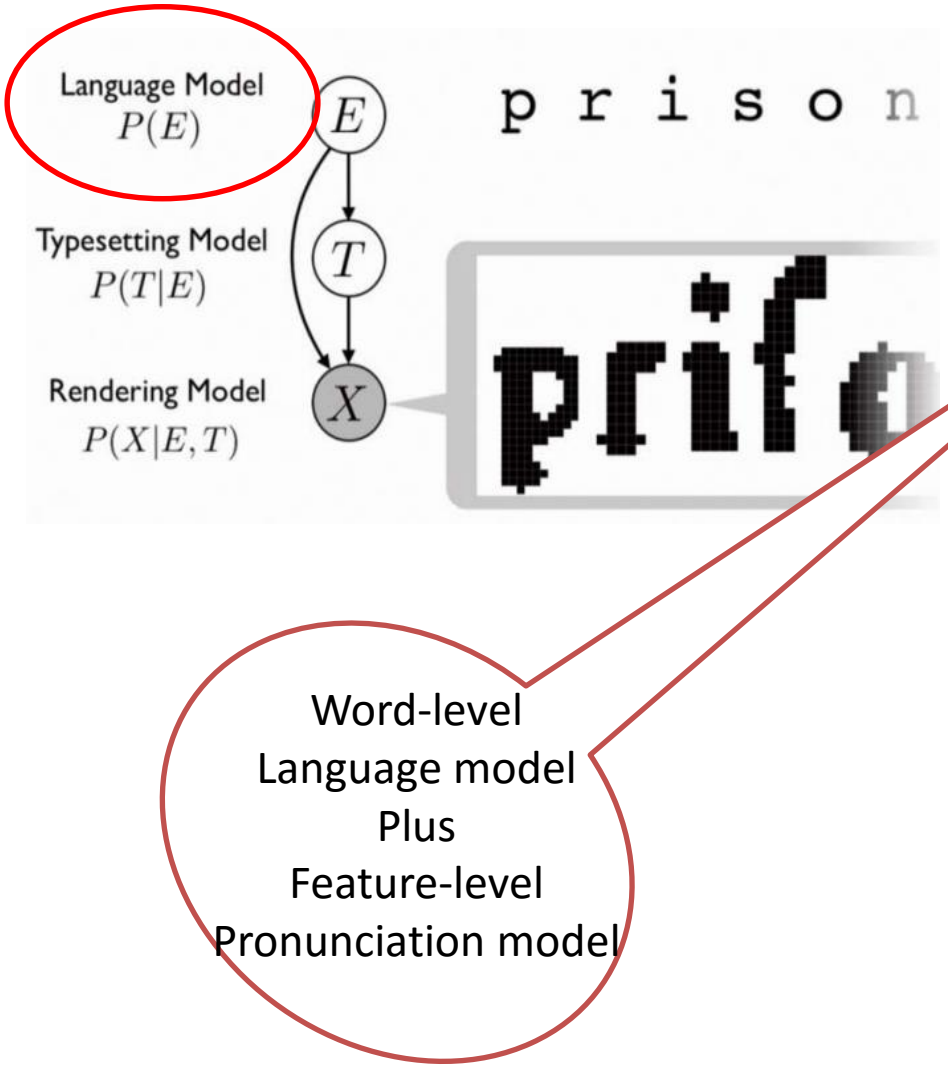
(**Deng, 1998**; Deng et al, 1997, 2000, 2003, 2006)



L.Deng, A dynamic, feature-based approach to the interface between phonology & phonetics for speech modeling and recognition, *Speech Communication*, vol. 24, no. 4, pp. 299-323, 1998.

Deep Generative Model for Image-Text

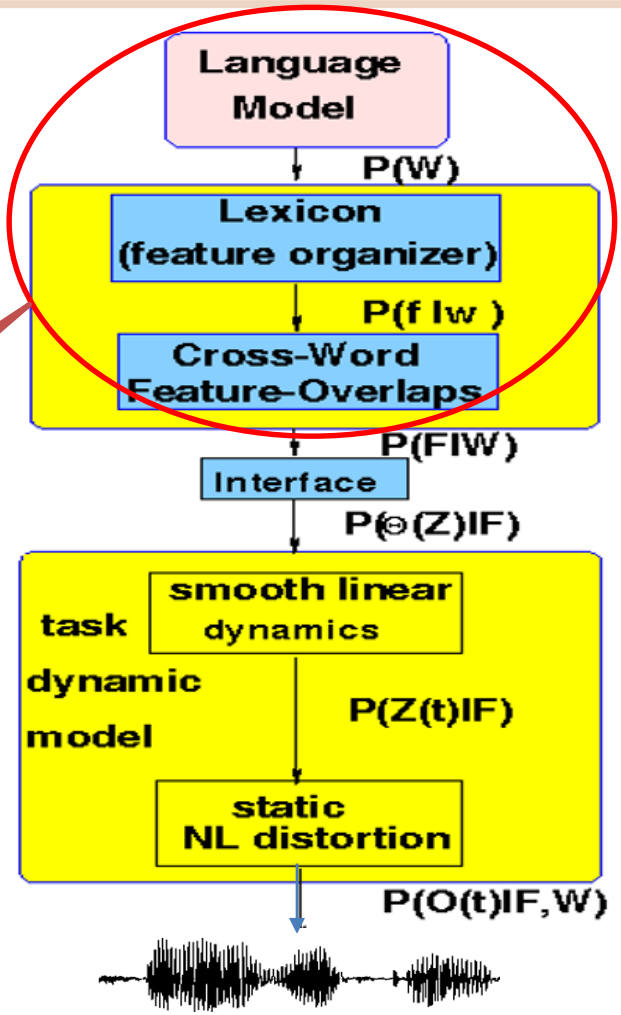
(Berg-Kirkpatrick et al., 2013, 2015)



Word-level
Language model
Plus
Feature-level
Pronunciation model

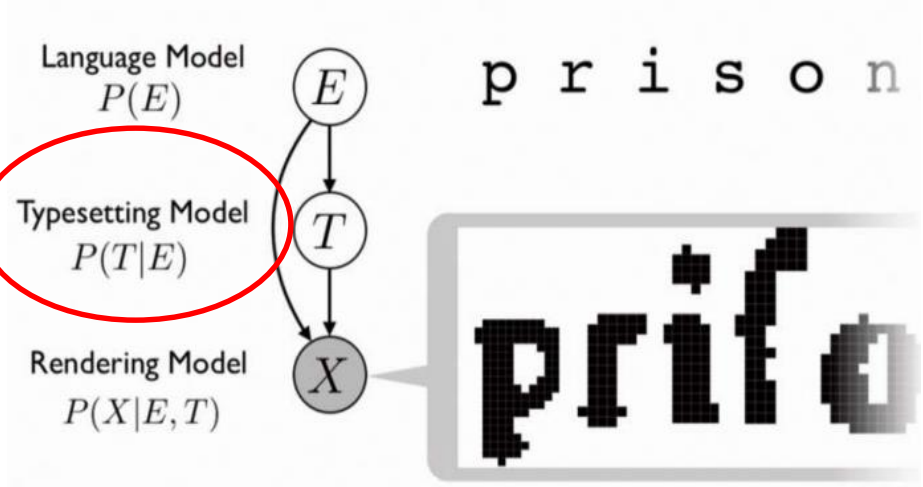
Deep Generative Model for Speech-Text

(Deng, 1998; Deng et al, 2000, 2003, 2006)



Deep Generative Model for Image-Text

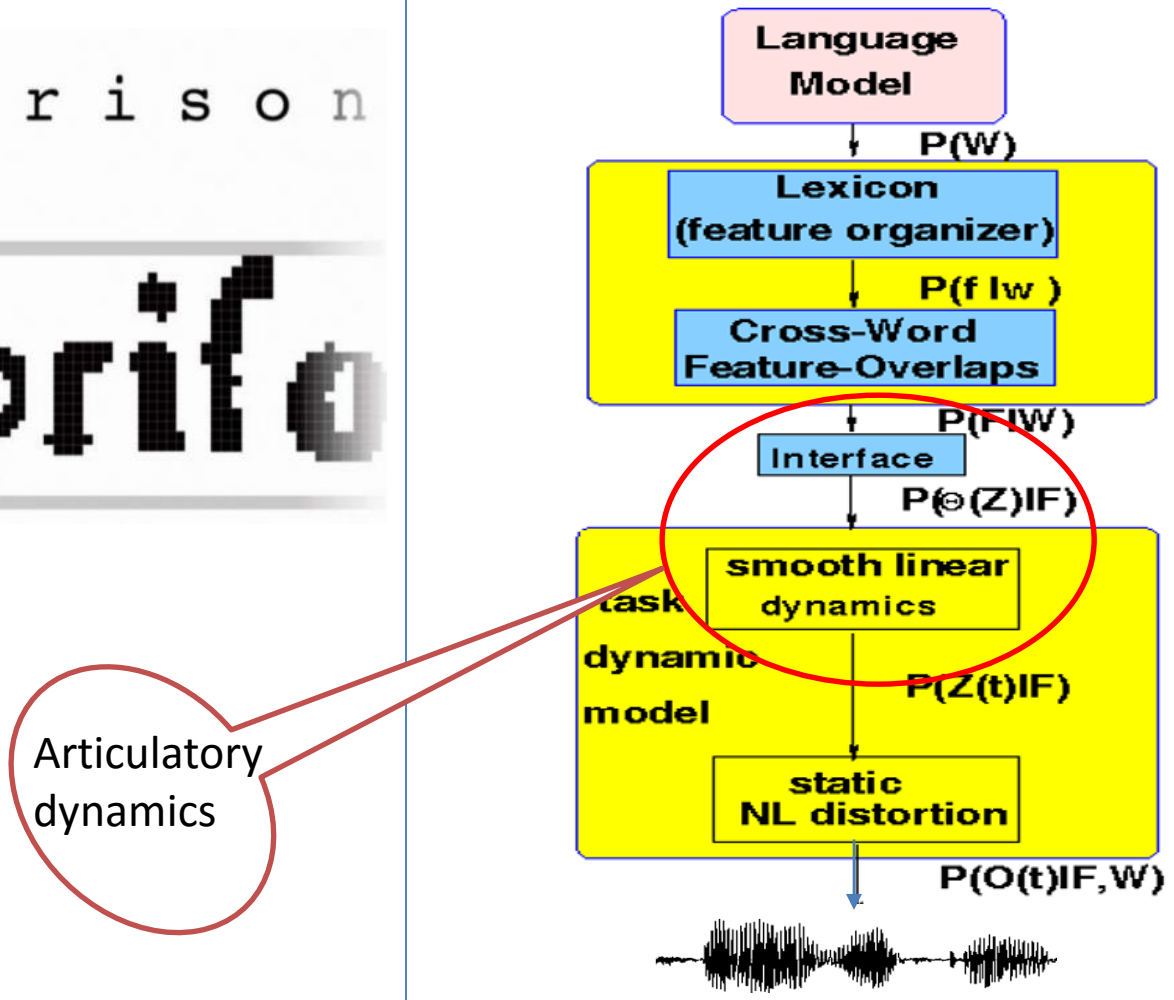
(Berg-Kirkpatrick et al., 2013, 2015)



Easy: likely no "explaining away" problem in inference and learning

Deep Generative Model for Speech-Text

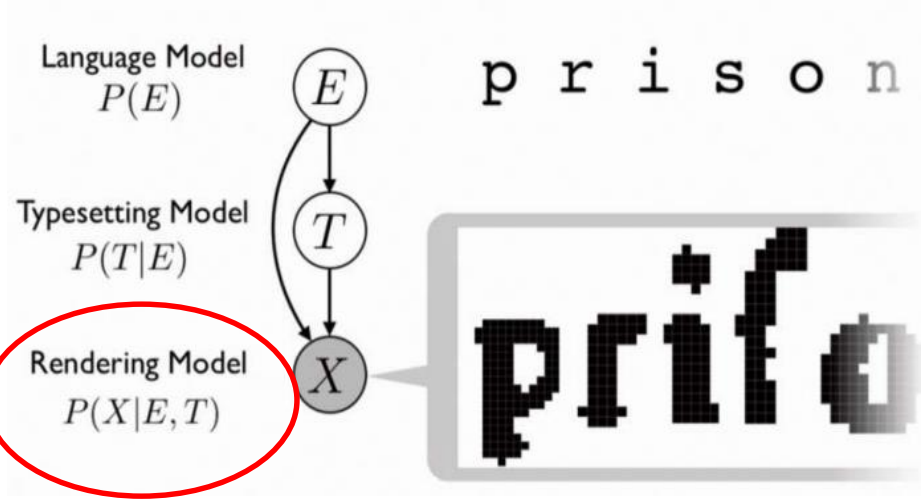
(Deng, 1998; Deng et al, 2000, 2003, 2006)



Hard: pervasive "explaining away" problem due to speech dynamics

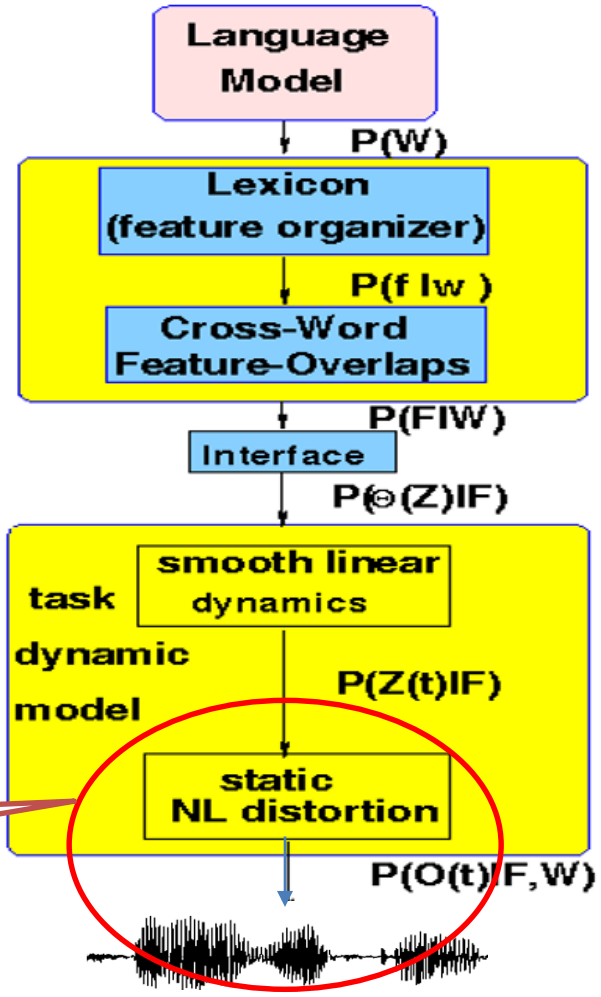
Deep Generative Model for Image-Text

(Berg-Kirkpatrick et al., 2013, 2015)



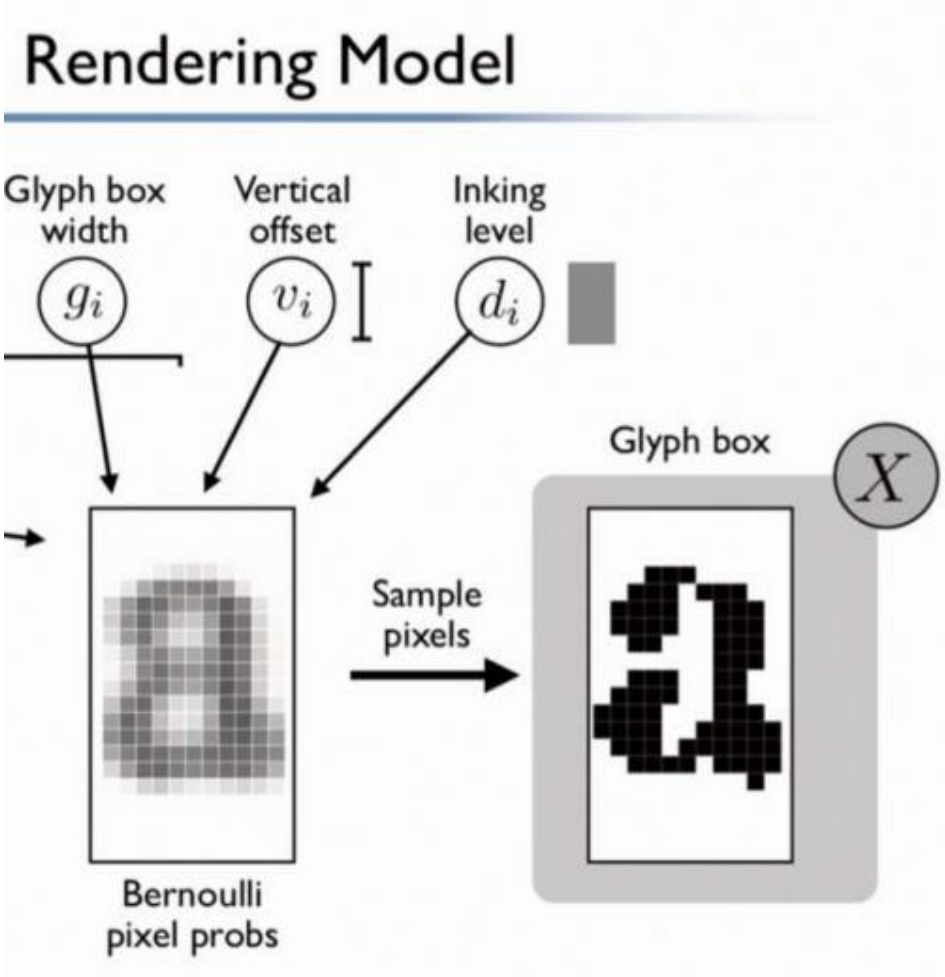
Deep Generative Model for Speech-Text

(Deng, 1998; Deng et al, 2000, 2003, 2006)



Articulatory To Acoustics mapping

Deep Generative Model for Image-Text



Very simple, & easy to model accurately

Deep Generative Model for Speech-Text

- In contrast, articulatory-to-acoustics mapping in ASR is much more complex
- During 1997-2000, shallow NNs were used for this as "universal approximator"
- Not successful
- Now we have better DNN tool
- Even RNN/LSTM/CTC tool for dynamic modeling

- Essence: exploit the strong prior of LM: trained with billions of words/text
- No need to pair the text with acoustics; hence unsupervised learning
- Think of it as using a new objective function based on distribution matching between LM prior and the distribution of words predicted from acoustic models

Further thoughts on unsupervised ASR

- Deep generative modeling experiments not successful in 90's
  - ~~Computers were too slow~~
  - Models were too simple from text to speech waves
    - Still true → need speech scientists to work harder with technologists
    - And when generative models are not good enough, discriminative models and learning (e.g., RNN) can help a lot
    - Further, can iterate between the two, like wake-sleep (algorithm)
  - Inference/learning methods for deep generative models not mature at that time
    - Only partially true today
    - ← due to recent big advances in machine learning
    - Based on new ways of thinking about generative graphical modeling motivated by the availability of deep learning tools (e.g. DNN)
    - A brief review next

Advances in Inference Algms for Deep Generative Models

Kingma & Welling 2014, Salakhutdinov et al, 2015

ICML-2014 Talk Monday June 23, 15:20

In Track F (Deep Learning II)

"Efficient Gradient Based Inference through Transformations between Bayes Nets and Neural Nets"

Other solutions to solve the "large variance problem" in variational inference:

-Variational Bayesian Inference with Stochastic Search [D.M. Blei, M.I. Jordan and J.W. Paisley, 2012]
-Fixed-Form Variational Posterior Approximation through Stochastic Linear Regression [T. Salimans and A. Knowles, 2013].
-Black Box Variational Inference. [R. Ranganath, S. Gerrish and D.M. Blei. 2013]
-Stochastic Variational Inference [M.D. Hoffman, D. Blei, C. Wang and J. Paisley, 2013]
-Estimating or propagating gradients through stochastic neurons. [Y. Bengio, 2013].
-Neural Variational Inference and Learning in Belief Networks. [A. Mnih and K. Gregor, 2014, ICML]
-Stochastic backprop & approximation inference in deep generative models [D. Rezende, S. Mohamed, D. Wierstra, 2014]
-Semi-supervised learning with deep generative models [K. Kingma, D. Rezende, S. Mohamed, M. Welling, 2014, NIPS]
-auto-encoding variational Bayes [K. Kingma, M. Welling, 2014, ICML]
-Learning stochastic recurrent networks [Bayer and Osendorfer, 2015 ICLR]
-DRAW: A recurrent neural network for image generation. [K. Gregor, Danihelka, Rezende, Wierstra, 2015]
- Plus a number of NIPS-2015 papers, to appear.

Slide provided by Max Welling (ICML-2014 tutorial) w. my updates on references of 2015 and late 2014

Further thoughts on unsupervised ASR

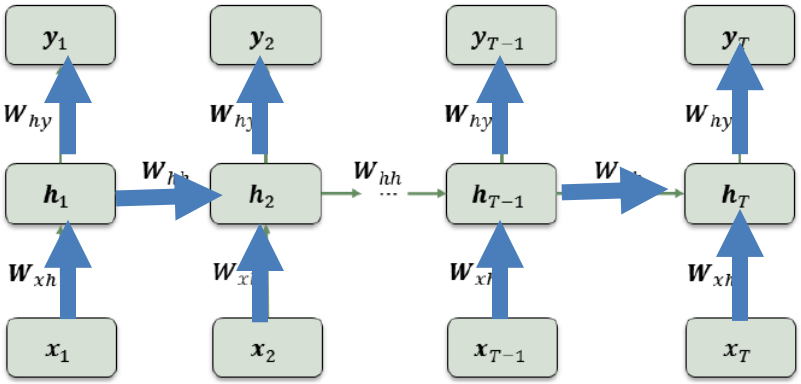
- Deep generative modeling experiments not successful in 90's
  - ~~Computers were too slow~~
  - Models were too simple from text to speech waves
    - Still true → need speech scientists to work harder with technologists
    - **And when generative models are not good enough, discriminative models and learning (e.g., RNN) can help a lot; but to do it? A hint next**
    - Further, can iterate between the two, like wake-sleep (algorithm)
  - Inference/learning methods for deep generative models not mature at that time
    - Only partially true today
    - ← due to recent big advances in machine learning
    - Based on new ways of thinking about generative graphical modeling motivated by the availability of deep learning tools (e.g. DNN)
    - A brief review next

RNN vs. Generative HDM

$$\mathbf{h}_t = f(\mathbf{h}_{t-1}; \mathbf{W}_{hh}, \mathbf{W}_{xh}, \mathbf{x}_t)$$
$$\mathbf{y}_t = g(\mathbf{h}_t; \mathbf{W}_{hy})$$

Parameterization:

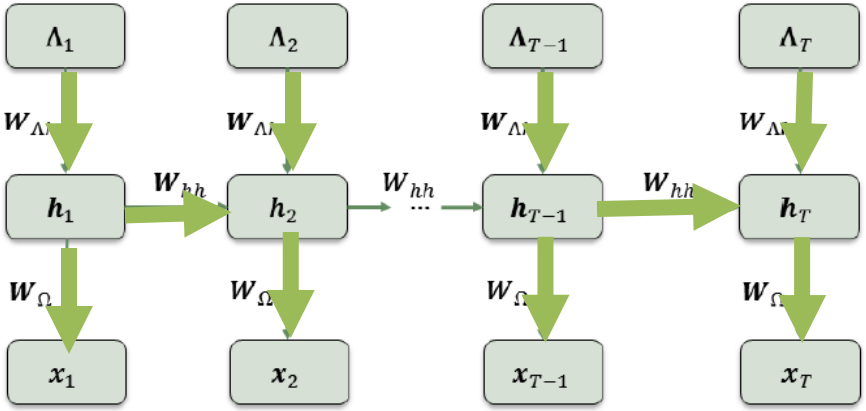
- W_{hh}, W_{hy}, W_{xh}: all unstructured regular matrices



$$\mathbf{h}_t = q(\mathbf{h}_{t-1}; \mathbf{W}_{l_t}, \mathbf{t}_{l_t})$$
$$\mathbf{x}_t = r(\mathbf{h}_t, \Omega_{l_t})$$

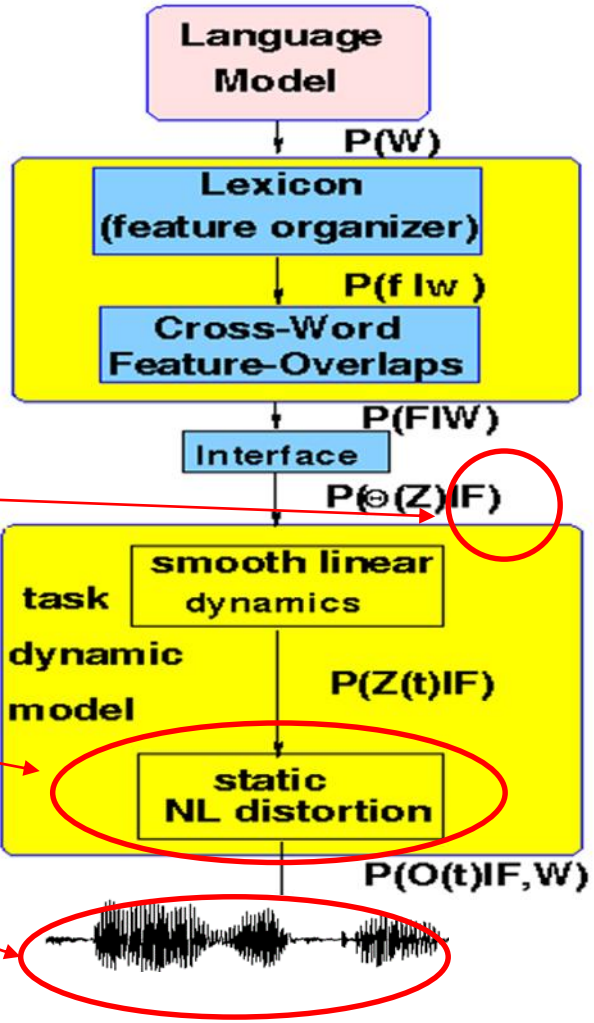
Parameterization:

- W_{hh} =M(γ_l); sparse system matrix
- W_Ω =(Ω_l); Gaussian-mix params; MLP
- $\Lambda = \boldsymbol{t}_l$



Generative HDM

$$\mathbf{h}_t = q(\mathbf{h}_{t-1}; \mathbf{W}_{l_t}, \mathbf{t}_{l_t})$$
$$\mathbf{x}_t = r(\mathbf{h}_t, \Omega_{l_t})$$



RNN vs. Generative HDM

$$\mathbf{h}_t = f(\mathbf{h}_{t-1}; \mathbf{W}_{hh}, \mathbf{W}_{xh,}, \mathbf{x}_t)$$
$$\mathbf{y}_t = g(\mathbf{h}_t; \mathbf{W}_{hy})$$

$$\mathbf{h}_t = q(\mathbf{h}_{t-1}; \mathbf{W}_{l_t}, \mathbf{t}_{l_t})$$
$$\mathbf{x}_t = r(\mathbf{h}_t, \Omega_{l_t})$$

~DNN

~DBN

e.g. Generative pre-training

(analogous to generative DBN pretraining for DNN)

NIPS-2015 paper to appear on simpler dynamic models for a non-ASR application

Better ways of integrating deep generative/discriminative models are possible

- Hint: Example 1 where generative models are used to define the DNN architecture

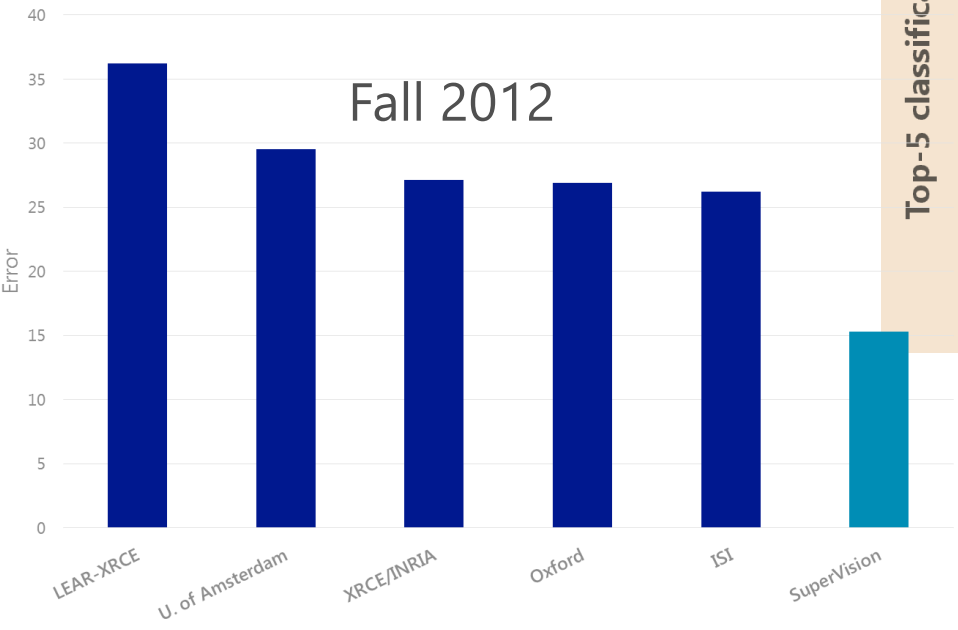
end of
Part II: Speech

-Deep supervised learning shattered ASR via DNN/LSTM
-Deep unsupervised learning may impact more in the future
-No more low-hanging fruit

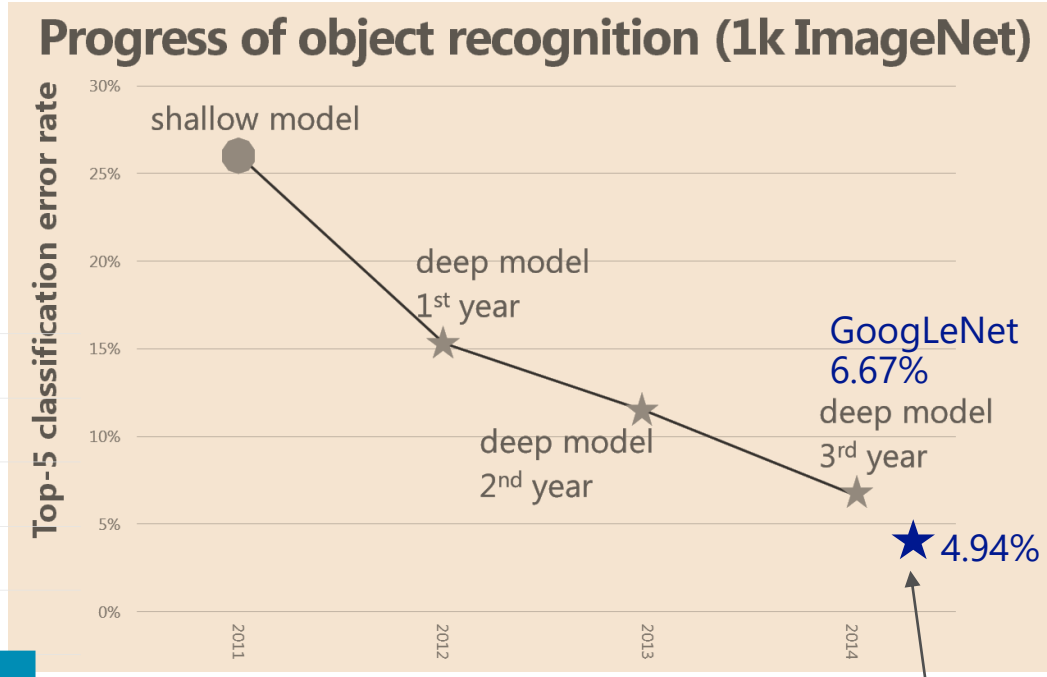
Deep Learning also Shattered the Entire Field of Image Recognition and Computer Vision (since 2012)

ImageNet-1K Competition

Krizhevsky, Sutskever, Hinton, "ImageNet Classification with Deep Convolutional Neural Networks." *NIPS*, Dec. 2012



Top-5 classification error rate



2012 - 2015

Deep CNN
Univ. Toronto team

Microsoft Research

Google

Baidu 百度 ?

Part III: Language

Moving **from perception**: phonetic/word recognition, image/gesture recognition, etc

to

Cognition: **memory, attention**, **reasoning, Q/A, & decision making**, etc.

Embedding enables exploring models for these human cognitive functions

Embedding Linguistic entities
in low-dimensional continuous space
and
Examples of NLP Applications

Semantic embedding

Project raw text into a continuous semantic space

e.g., word embedding

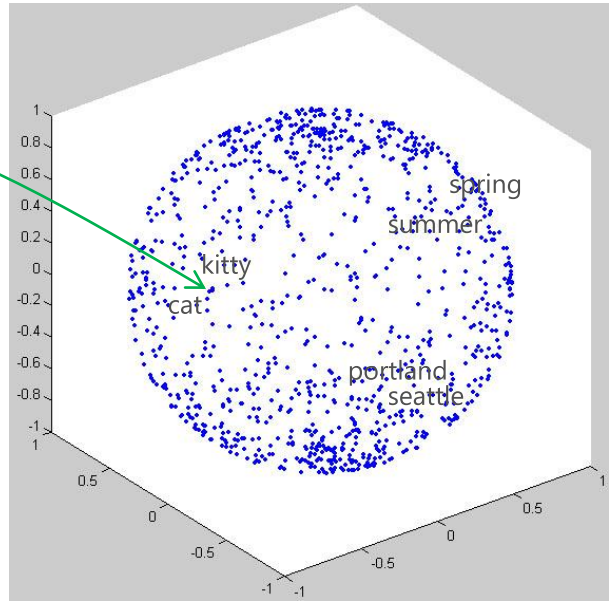
Captures the word meaning in a semantic space

$f(\boldsymbol{cat}) =$

The index of "cat" in the vocabulary

a.k.a the 1-hot word vector

word embedding vector in the semantic space



Dim=100~1000

Dim=|V|=100K~100M

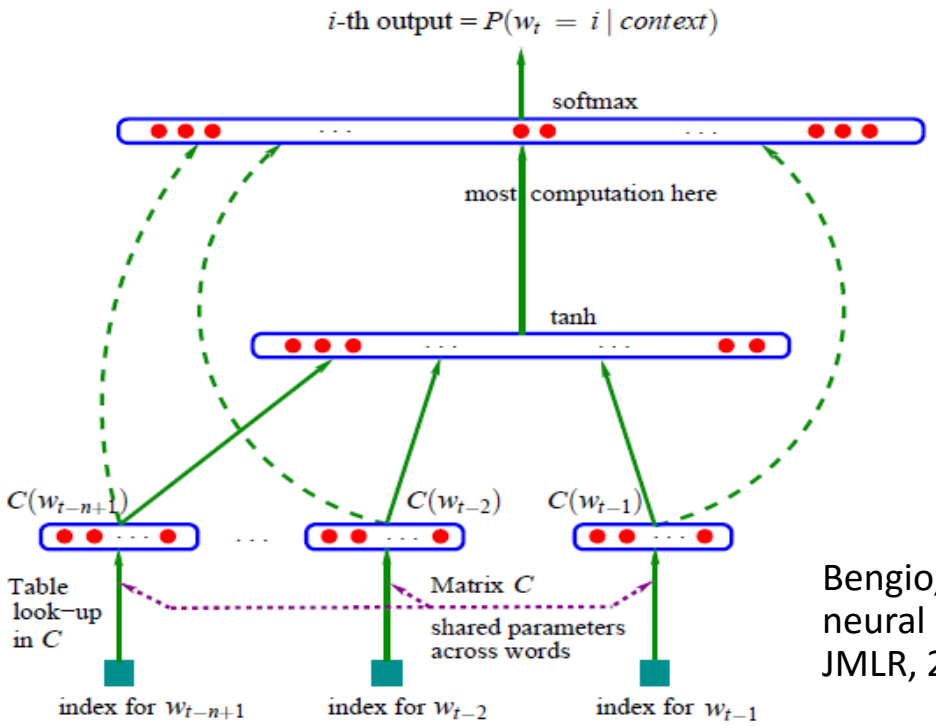
Deerwester, Dumais, Furnas, Landauer, Harshman, "Indexing by latent semantic analysis," JASIS 1990

Microsoft Research

NN Word Embedding

LM: predict the next word given the past:

e.g., $p(chases|the\ cat) = ?$, $p(says|the\ cat) = ?$



Bengio, Ducharme, Vincent, Jauvin, "A neural probabilistic language model. " JMLR, 2003

SENNA word embedding

Scoring:

$$Score(w_1, w_2, w_3, w_4, w_5) = U^T \sigma(W[f_1, f_2, f_3, f_4, f_5] + b)$$

Training:

$$J = \max(0, 1 + S^- - S^+)$$

Update the model until $S^+ > 1 + S^-$

Where

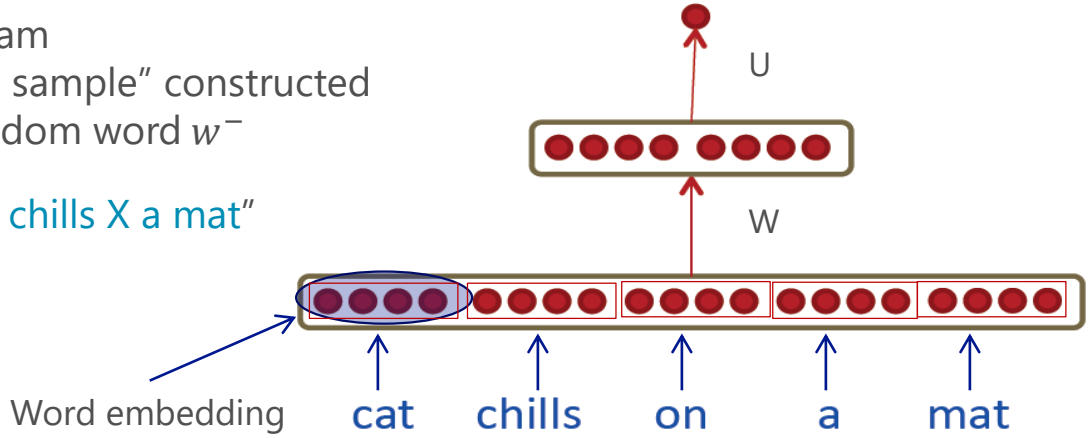
$$S^+ = Score(w_1, w_2, w_3, w_4, w_5)$$
$$S^- = Score(w_1, w_2, w^-, w_4, w_5)$$

And

$< w_1, w_2, w_3, w_4, w_5 >$ is a valid 5-gram

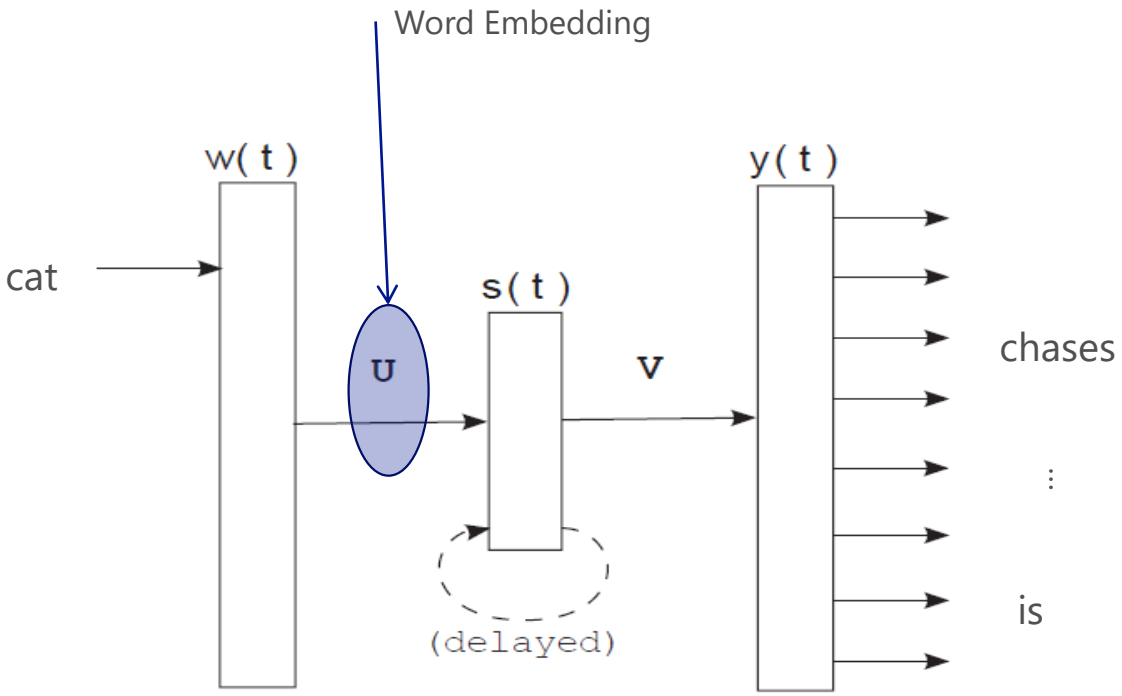
$< w_1, w_2, w^-, w_4, w_5 >$ is a "negative sample" constructed by replacing the word w_3 with a random word w^-

e.g., a negative example: "cat chills X a mat"



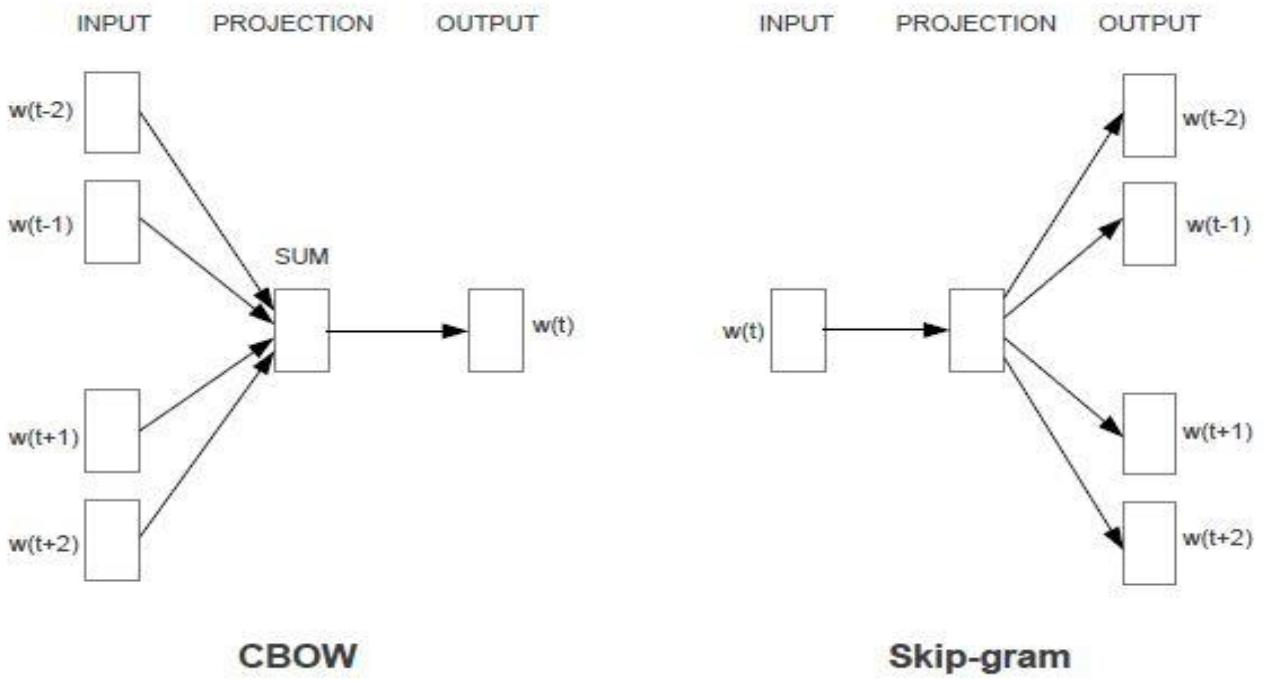
Collobert, Weston, Bottou, Karlen, Kavukcuoglu, Kuksa, "Natural Language Processing (Almost) from Scratch," JMLR 2011

RNN-LM base word embedding



Mikolov, Yih, Zweig, "Linguistic Regularities in Continuous Space Word Representations," NAACL 2013

CBOW/Skip-gram Word Embeddings

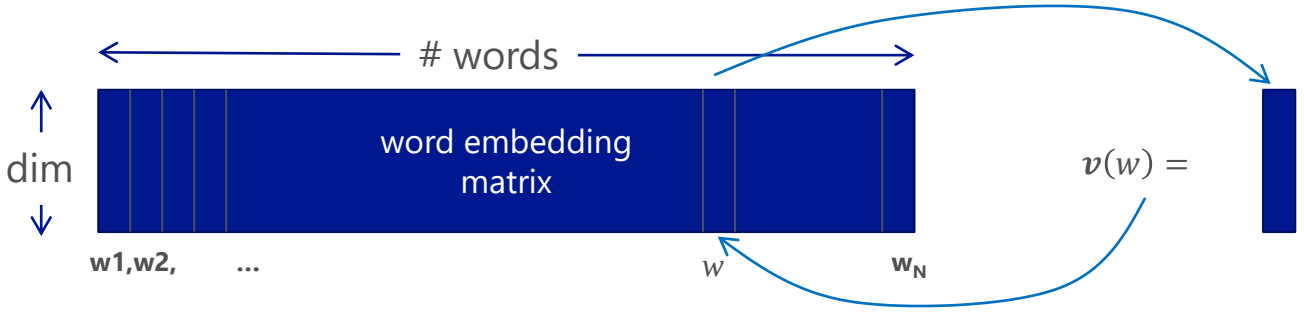


Continuous Bag-of-Words

The CBOW architecture (a) on the left, and the Skip-gram architecture (b) on the right. [Mikolov et al., 2013 ICLR].

Word embedding: rethinking

- Word embedding is a neat and effective representation:



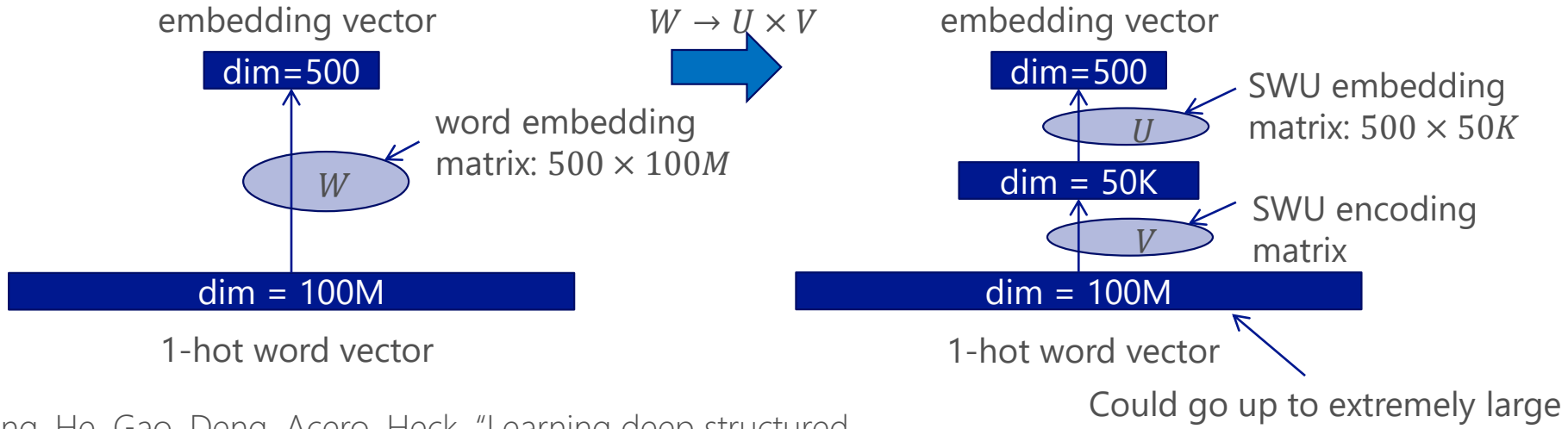
- However, for large scale NL tasks a decomposable, robust representation is preferable
  - Vocabulary of real-world big data tasks could be huge (*scalability*)

    >100M unique words in a modern commercial search engine log, and keeps growing
  - New words, misspellings, and word fragments frequently occur (*generalizability*)

Build semantic embedding on top of sub-word units

Learn semantic embedding on top of sub-word units (SWU)

- Decompose *any* word into sub-word units
- *Scale* the capacity to handle almost unbounded variability (word) based on bounded variability (sub-word)



Huang, He, Gao, Deng, Acero, Heck, "Learning deep structured semantic models for web search using clickthrough data," CIKM, 2013

Sub-word unit

- Letters, context-dept letters, positioned-phones, context-dept phones, positioned-roots/morphs, context-dept morphs
- Multi-hashing approach to word input representation

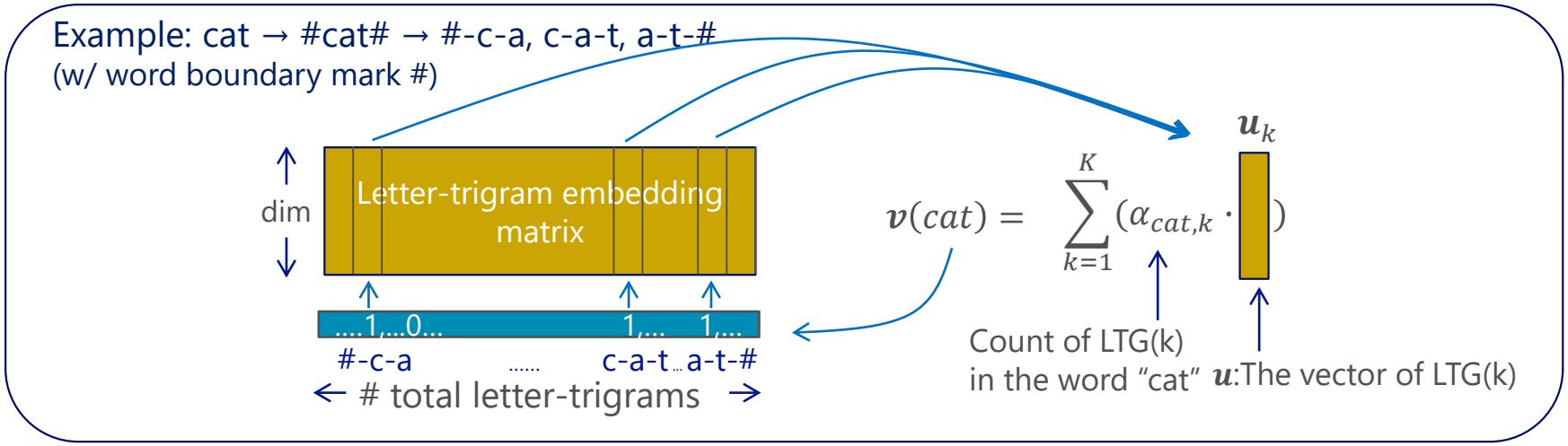
Or random projection (random basis)

From sub-word unit embedding vectors to word vectors

SWU uses context-dependent letter, e.g., letter-trigram.

Learn one vector per letter-trigram (LTG), the encoding matrix is a fixed matrix

- Use the count of each LTG in the word for encoding



Two words has the same LTG: collision rate ≈ 0.004%

Supervised Embeddings for Semantic Modeling with Applications

--- embedding linguistic symbols by backprop
--- mining distant supervision signals

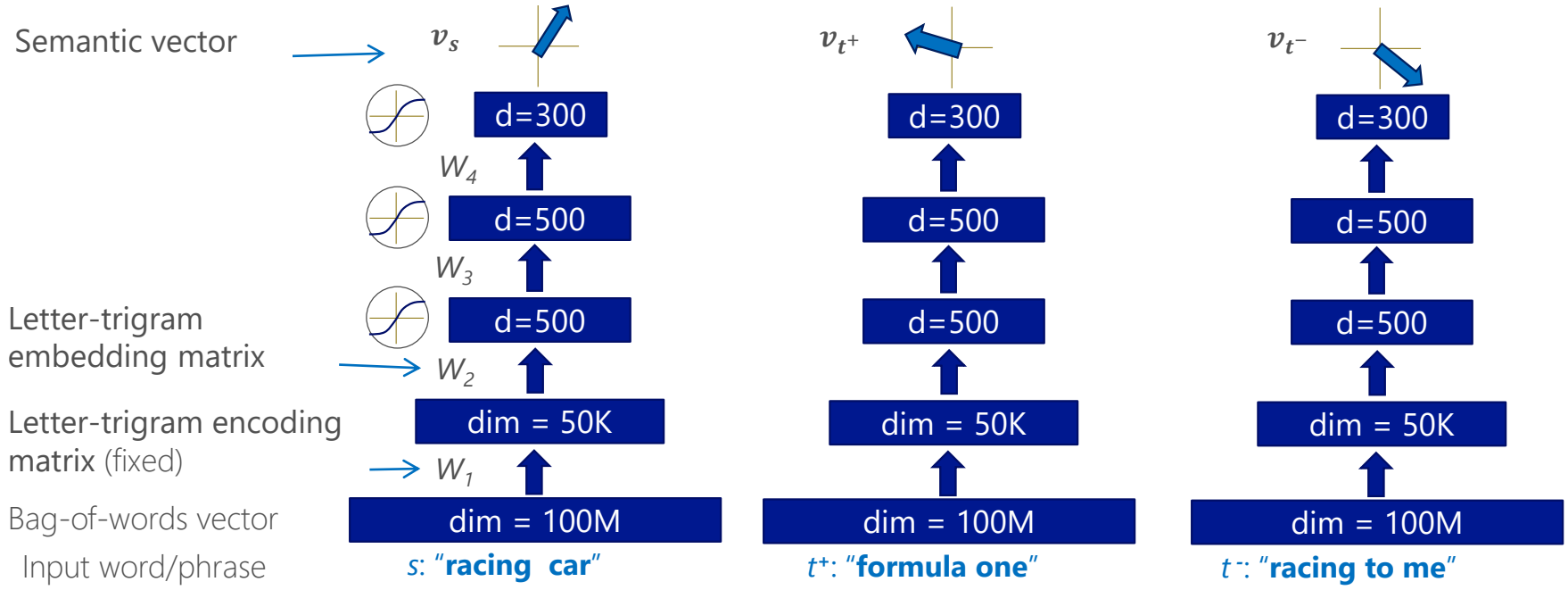
Deep Structured Semantic Model (DSSM)

- Build word/phrase, or sentence-level semantic vector representation
- Trained by a similarity-driven objective
  - projecting semantically similar phrases to vectors close to each other
  - projecting semantically different phrases to vectors far apart

DSSM for learning semantic embedding

Initialization:

Neural networks are initialized with random weights

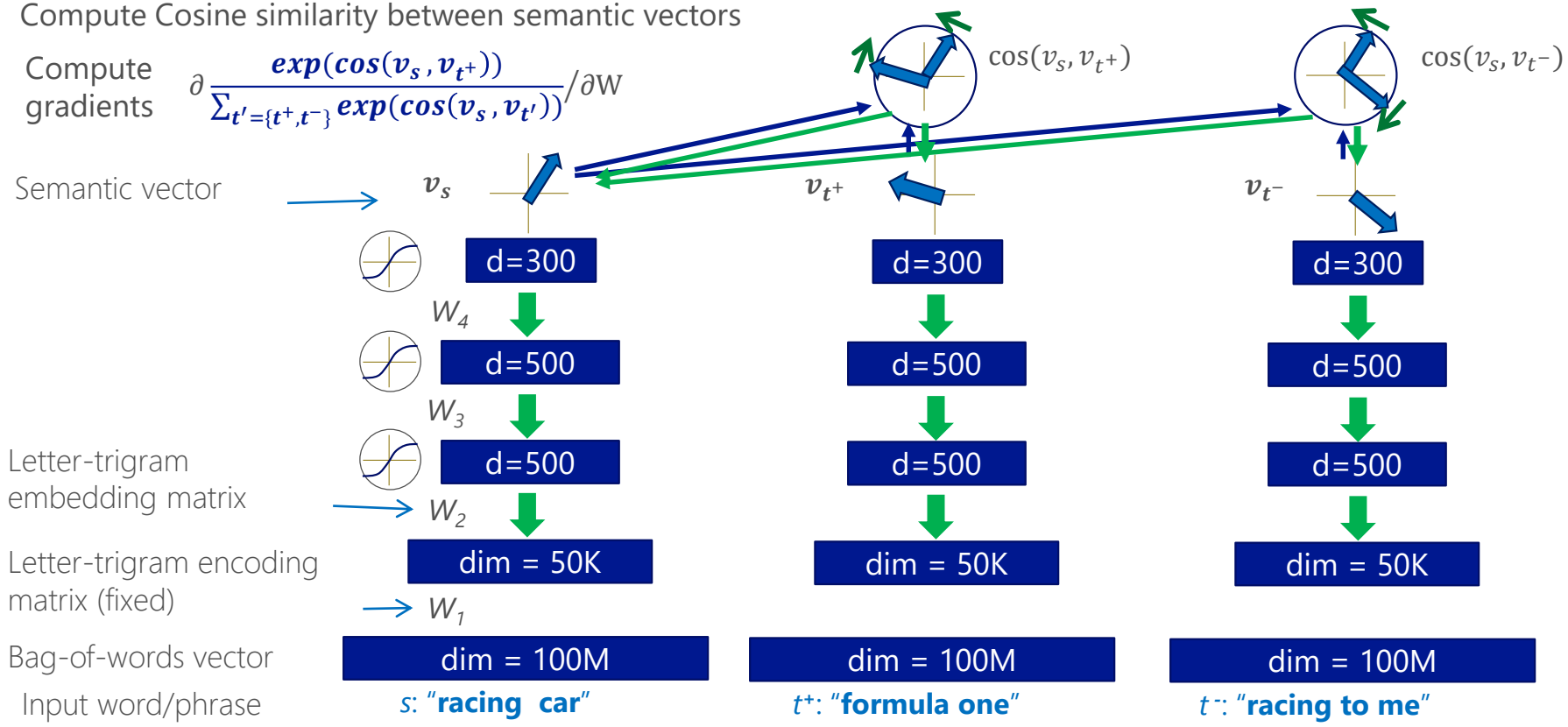


DSSM for learning semantic embedding

Training:

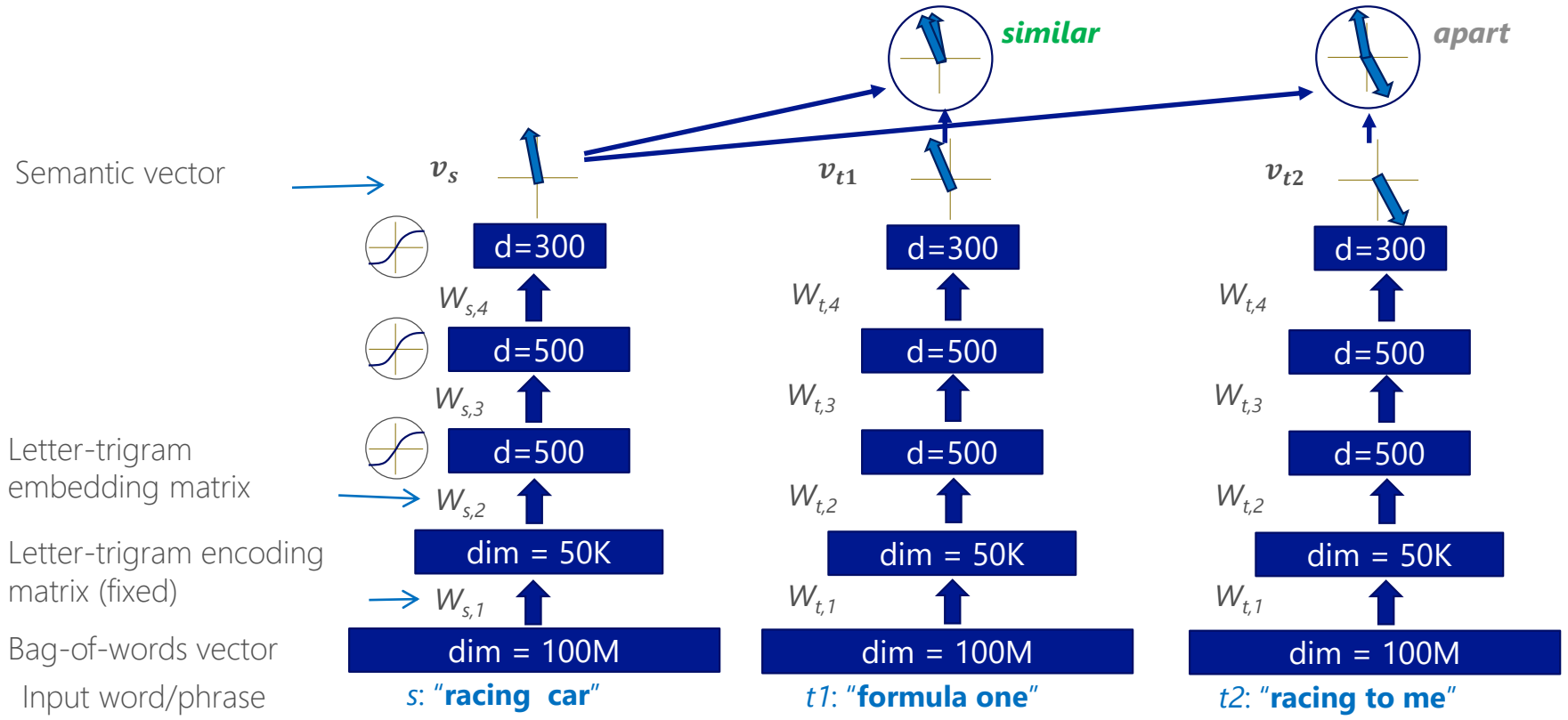
Compute Cosine similarity between semantic vectors

Compute gradients $\partial \frac{\exp(\cos(v_s, v_{t^+}))}{\sum_{t'=\{t^+,t^-\}} \exp(\cos(v_s, v_{t'}))} / \partial W$



DSSM for learning semantic embedding

Runtime:



Training of the DSSM

Data: semantically-similar text pairs

e.g., context <-> word in word embedding vector learning

query <-> clicked-doc in Web Search

pattern<-> relationship in Question Answering

Objective: cosine similarity based loss

- Web search as an example: a query $\boldsymbol{q}$ and a list of docs $\boldsymbol{D} = \{\boldsymbol{d}^+, \boldsymbol{d}_1^-, \ldots \boldsymbol{d}_K^-\}$
  - $\boldsymbol{d}^+$ positive doc; $\boldsymbol{d}_1^-, \ldots \boldsymbol{d}_K^-$ are negative docs to $\boldsymbol{q}$ (e.g., sampled from not clicked docs)

- Objective: the posterior probability of clicked document given query

$$P(d^+|q) = \frac{\exp\left(\gamma\, cos(q, d^+)\right)}{\sum_{d \in \boldsymbol{D}} \exp(\gamma\, cos(q, d))}$$

- Optimize $\boldsymbol{\theta}$ to maximize $\boldsymbol{P}(\boldsymbol{d}^+|\boldsymbol{q})$. SGD training on GPU (NVidia K20x)

Many applications of DSSM (many low-hanging fruits):
Learning semantic similarity between *X* and *Y*

Tasks	*Source X*	*Target Y*
Word semantic embedding	*context*	*word*
Web search	*search query*	*web documents*
Query intent detection	*Search query*	*Use intent*
Question answering	*pattern / mention* (in NL)	*relation / entity* (in KB)
Machine translation	*sentence in language a*	*translated sentences in language b*
Query auto-suggestion	*Search query*	*Suggested query*
Query auto-completion	*Partial search query*	*Completed query*
Apps recommendation	*User profile*	*recommended Apps*
Distillation of survey feedbacks	*Feedbacks in text*	*Relevant feedbacks*
Automatic image captioning	***image***	***text caption***
Image retrieval	*text query*	*images*
Natural user interface	*command (text / speech / gesture)*	*actions*
Ads selection	*search query*	*ad keywords*
Ads click prediction	*search query*	*ad documents*
Email analysis: people prediction	*Email content*	*Recipients, senders*
Email search	*Search query*	*Email content*
Email declutering	*Email contents*	*Email contents in similar threads*
Knowledge-base construction	*entity from source*	*entity fitting desired relationship*
Contextual entity search	*key phrase / context*	*entity / its corresponding page*
Automatic highlighting	*documents in reading*	*key phrases to be highlighted*
Text summarization	*long text*	*summarized short text*

Automatic image captioning (MSR system)

Detector Models, Deep Neural Net Features

Computer Vision System

street
signs
light
under
on
pole
stop
red
building
sign
bus
city
traffic

Language Model

Caption Generation System

a red stop sign sitting under a traffic light on a city street
a stop sign at an intersection on a street
a stop sign with two street signs on a pole on a sidewalk
a stop sign at an intersection on a city street
…
a stop sign
a red traffic light

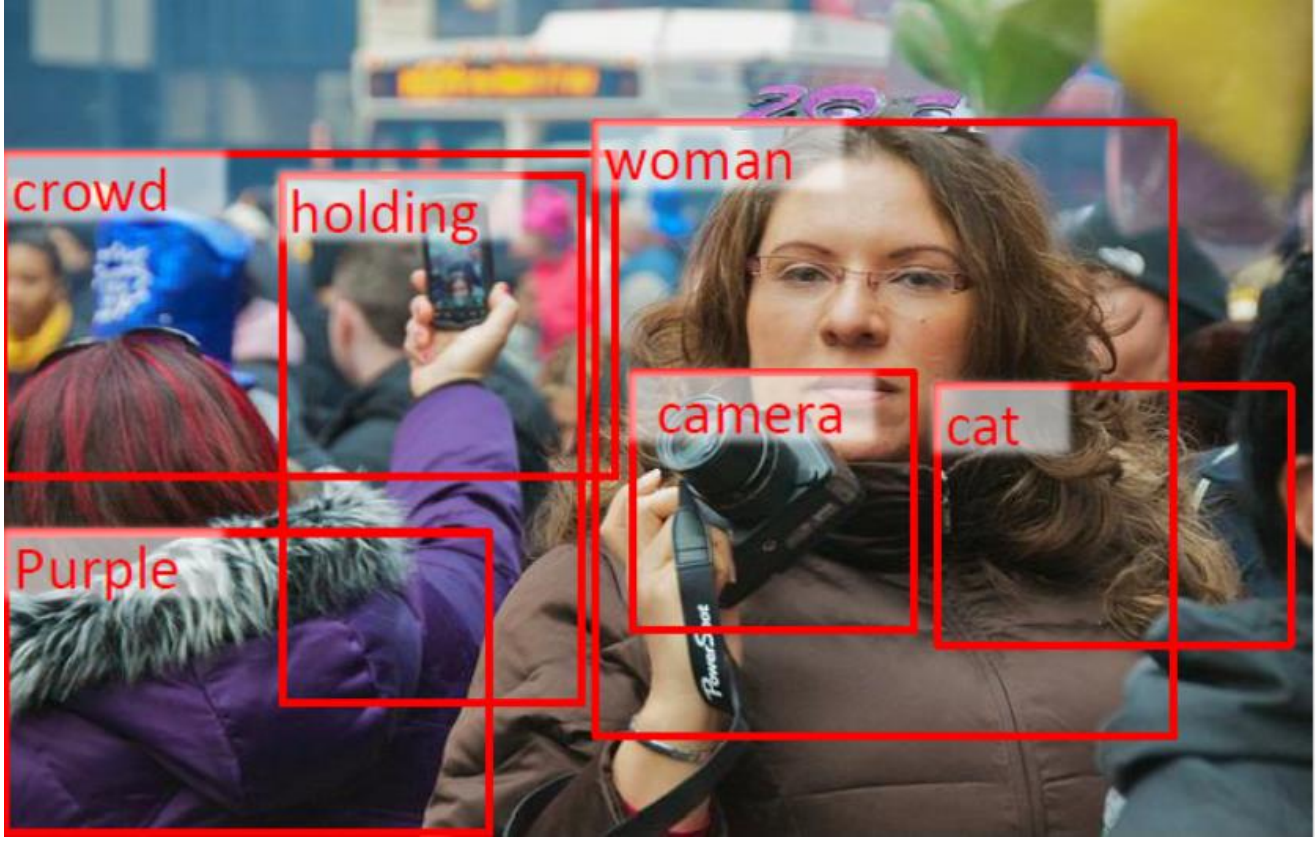
DSSM Model

Semantic Ranking System

a stop sign at an intersection on a city street

Fang, Gupta, Iandola, Srivastava, Deng, Dollar, Gao, He, Mitchell, Platt, Zitnick, Zweig, "**From captions to visual concepts and back**," accepted to appear in CVPR, 2015; in arXiv 2014

Microsoft System (MSR): Use of DSSM for Global Semantic Matching



1. Word Detection → 2. Sentence Generation → 3. Sentence Re-Ranking

woman, crowd, cat, camera, holding, purple

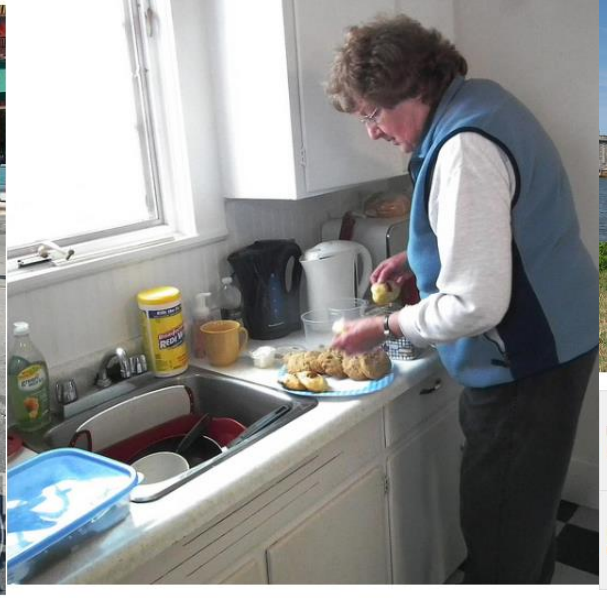
A purple camera with a woman.
A woman holding a camera in a crowd.
...
A woman holding a cat.

#1 A woman holding a camera in a crowd.

A	a woman in a kitchen preparing food
B	woman working on counter near kitchen sink preparing a meal

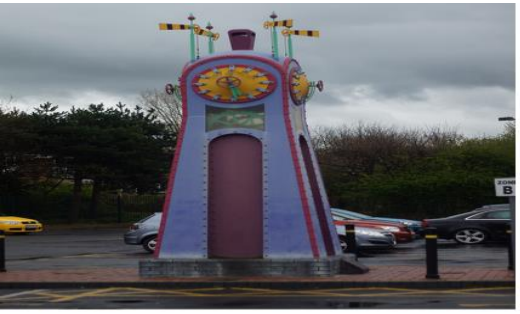
Machine:	a woman in a kitchen preparing food
Human:	woman working on counter near kitchen sink preparing a meal

Machine-generated (but turker prefered)	a group of motorcycles parked next to a motorcycle
Human-annotated (but turker not prefered)	two girls wearing are wearing short skirts and one of them sits on a motorcycle while the other stands nearby



Machine-generated (but turker prefered)	a woman in a kitchen preparing food
Human-annotated (but turker not prefered)	woman working on counter near kitchen sink preparing a meal

Machine-generated (but turker prefered)	a bicycle is parked next to a river
Human-annotated (but turker not prefered)	a bike sits parked next to a body of water



Machine-generated (but turker prefered)	a clock tower in the middle of the street
Human-annotated (but turker not prefered)	a statue with a clock on it near a parking lot

Machine-generated (but turker prefered)	a kitchen with wooden cabinets and a sink
Human-annotated (but turker not prefered)	an ornate kitchen is designed with rustic wooden parts

Machine-generated (but turker prefered)	a man holding a tennis racquet on a tennis court
Human-annotated (but turker not prefered)	the man is on the tennis court playing a game

Potential Scenarios

take a snapshot, generate a caption,
and share the fun ☺

a dining room table and chairs next to a window

a young boy standing on a beach

a group of people sitting at a table with laptops

a man that is standing in front of a store

Field test of our prototype App: on cell phone-quality photos

Slide from Xiaodong He (MSR)

Competition results (CVPR-2015, June, Boston)

Won 1st Prize at MS COCO Captioning Challenge 2015!

Measure the quality of the captions by human judge (e.g., *Turing Test*).

The top teams and the state-of-the-art

	% of captions that pass the Turing Test	Official Rank
MSR	32.2%	1st(tie)
Google	31.7%	1st(tie)
MSR Captivator	30.1%	3rd(tie)
Montreal/Toronto	27.2%	3rd(tie)
Berkeley LRCN	26.8%	5th
Other groups: Baidu/UCLA, Stanford, Tsinghua, etc.		

Note: even a *Human* cannot guarantee to pass Turing Test by 100%

Human	67.5%	--

Many applications of DSSM (many low-hanging fruits):
Learning semantic similarity between *X* and *Y*

Tasks	*Source X*	*Target Y*
Word semantic embedding	*context*	*word*
Web search	*search query*	*web documents*
Query intent detection	*Search query*	*Use intent*
Question answering	*pattern / mention* (in NL)	*relation / entity* (in KB)
Machine translation	*sentence in language a*	*translated sentences in language b*
Query auto-suggestion	*Search query*	*Suggested query*
Query auto-completion	*Partial search query*	*Completed query*
Apps recommendation	*User profile*	*recommended Apps*
Distillation of survey feedbacks	*Feedbacks in text*	*Relevant feedbacks*
Automatic image captioning	*image*	*text caption*
Image retrieval	*text query*	*images*
Natural user interface	*command (text / speech / gesture)*	*actions*
Ads selection	*search query*	*ad keywords*
Ads click prediction	*search query*	*ad documents*
Email analysis: people prediction	*Email content*	*Recipients, senders*
Email search	*Search query*	*Email content*
Email declutering	*Email contents*	*Email contents in similar threads*
Knowledge-base construction	*entity from source*	*entity fitting desired relationship*
Contextual entity search	***key phrase / context***	***entity / its corresponding page***
Automatic highlighting	*documents in reading*	*key phrases to be highlighted*
Text summarization	*long text*	*summarized short text*

DECEMBER 10 2014

Bing brings the world's knowledge straight to you with Insights for Office

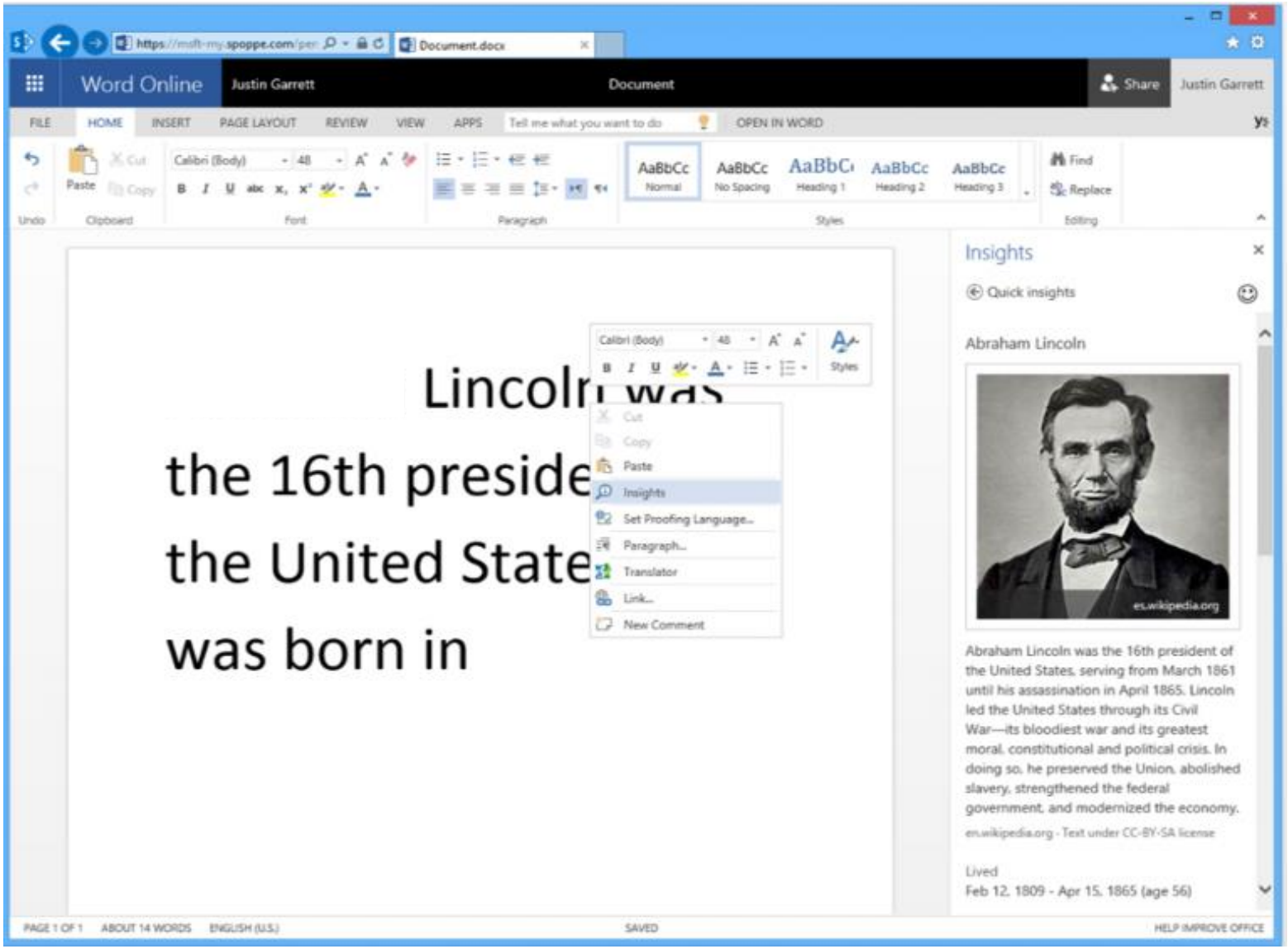
Today Bing and Office are introducing Insights for Office, a new, more powerful way to search for the information you need **while in Office Word Online** – available in English to all markets in the next few days. We encourage you to try it here, always free.

How Bing's intelligence powers Insights for Office

Bing indexes and stores entity data from around the web representing real world people, places and things. Insights for Office utilizes Bing's ability to index the world's knowledge and our machine learned relevance models to semantically understand the most important content in a user's document and then return the most relevant results. This capability is derived largely from patterns of text analysis developed in collaboration with Microsoft Research. The results deliver the

http://blogs.bing.com/search/2014/12/10/bing-brings-the-worlds-knowledge-straight-to-you-with-insights-for-office/

Scenario: Contextual search in Microsoft Office/Word



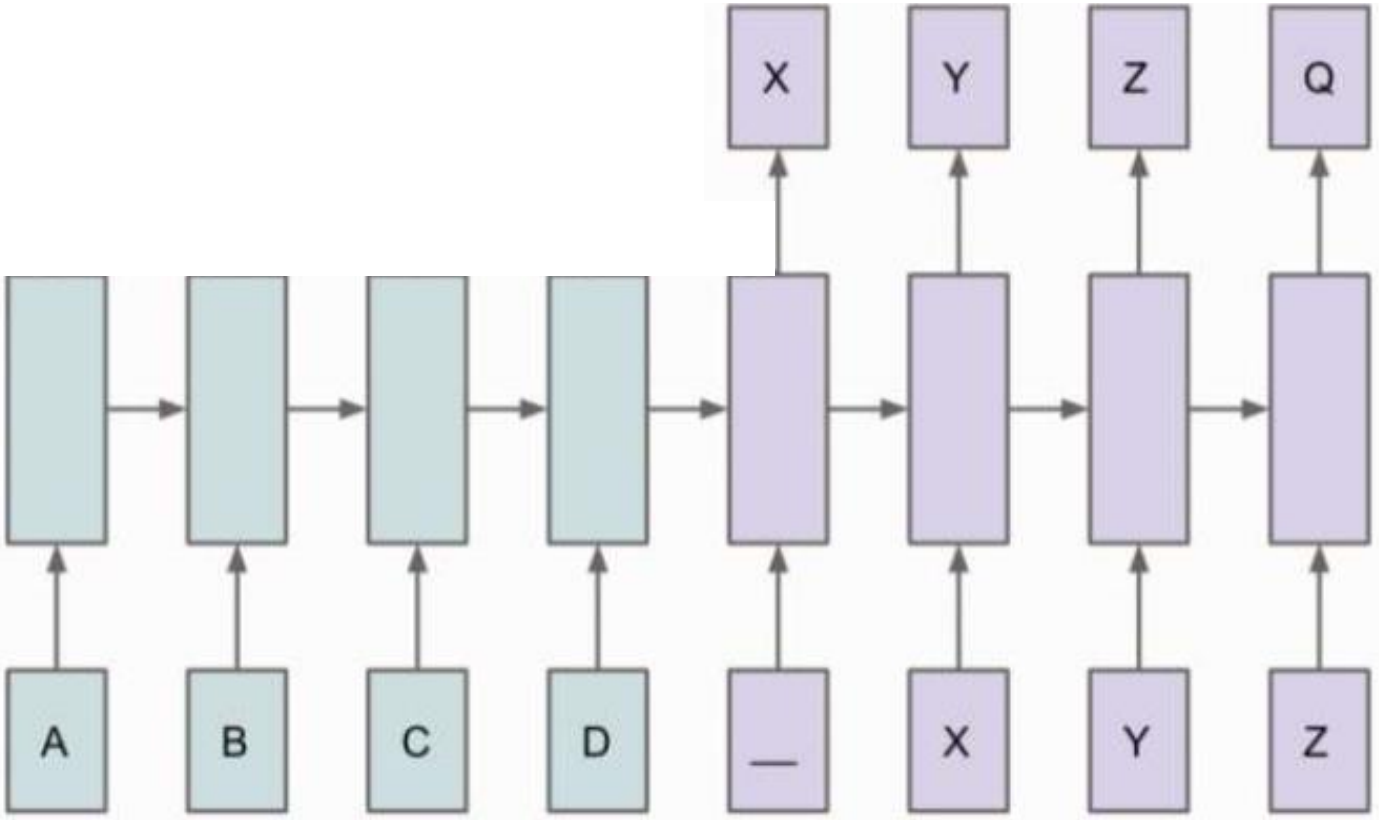
When “Lincoln” is selected, pages of a car company, movie, or the town in Nebraska will not appear[183]

Towards Modeling Cognitive Functions: Memory and Attention

-seq-to-seq learning via LSTM with attention mechanism
-memory nets and neural Turing machines
-dynamic memory nets
-from seq2seq to seq2struct and to struct2struct

Seq-2-Seq Learning for Machine Translation

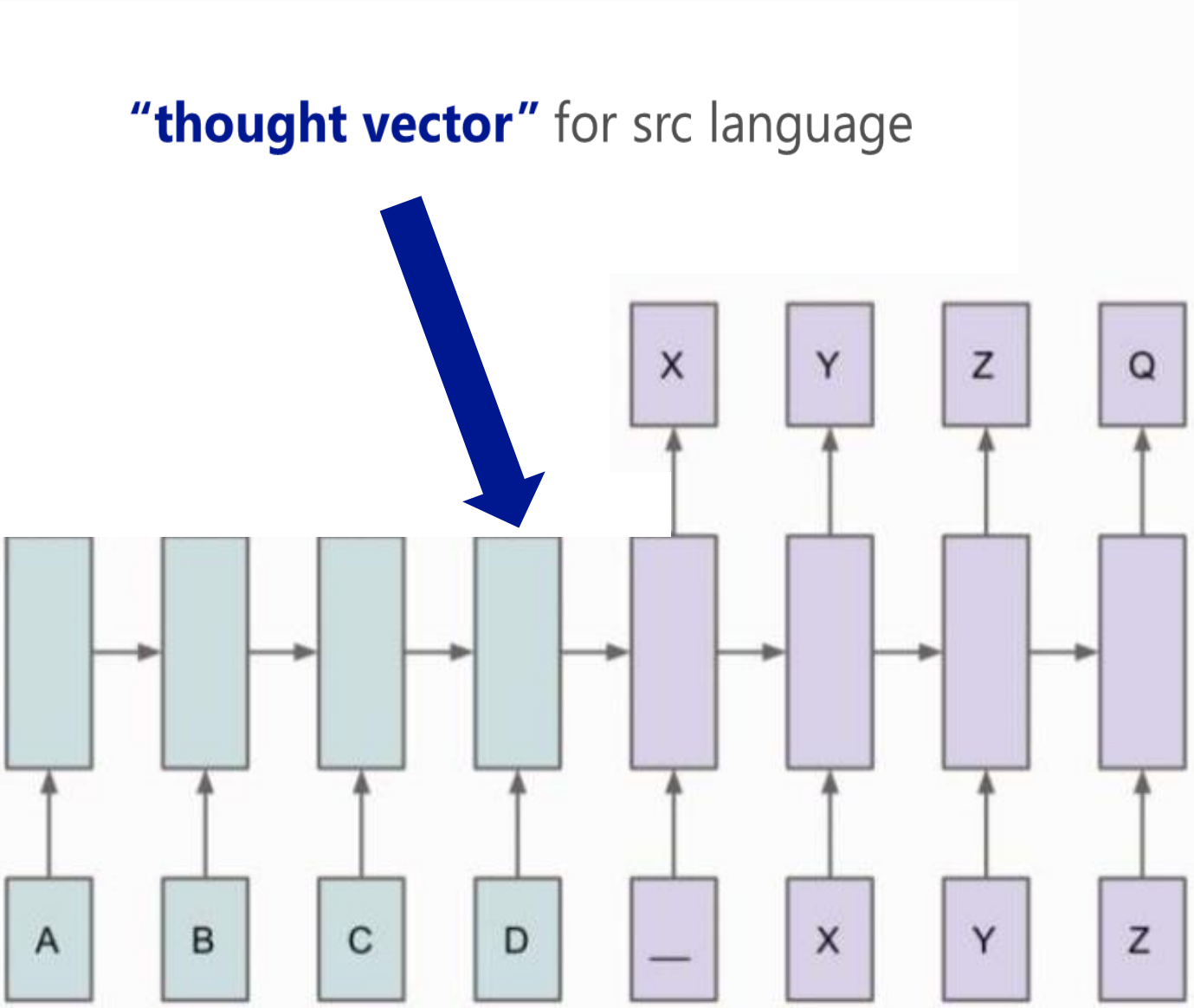
[Sutskever, Vinyals, Le, NIPS, 2014]



(Slides from Vinyals)

Seq-2-Seq Learning for Machine Translation

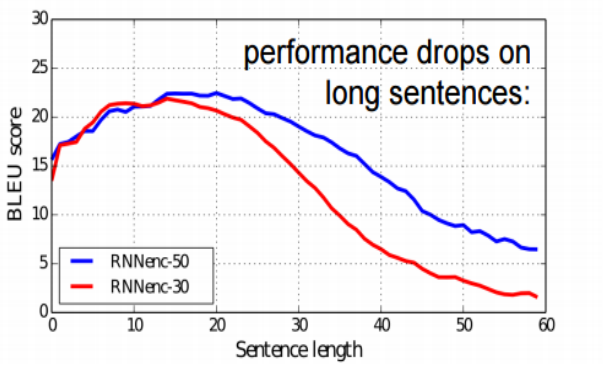
[Sutskever, Vinyals, Le, NIPS, 2014]

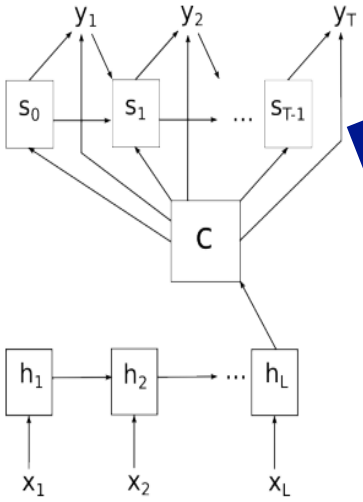


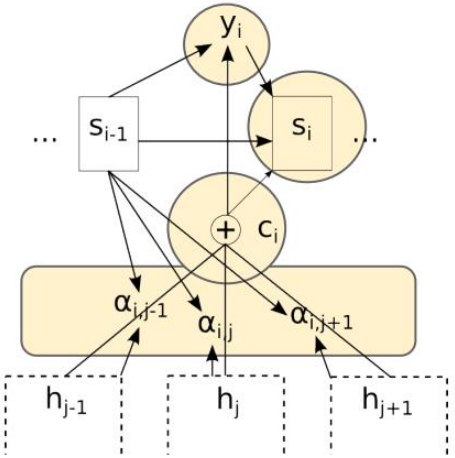
Bahdanau et al., ICLR, 2015

RNN Encoder-Decoder: Issues

- has to remember the whole sentence
- fixed size representation can be the bottleneck
- humans do it differently







"attention" mechanism
used for "softly" selecting relevant input portion from memory

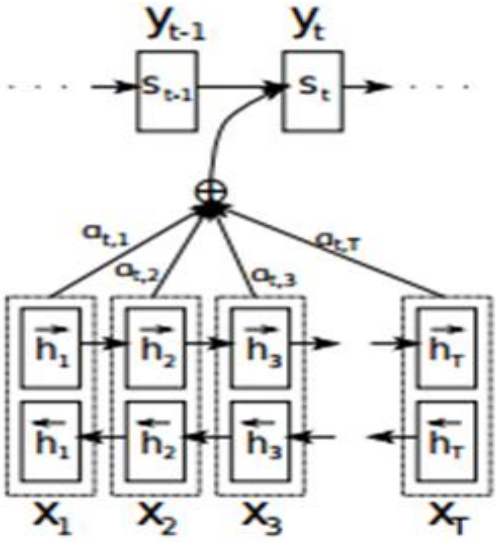
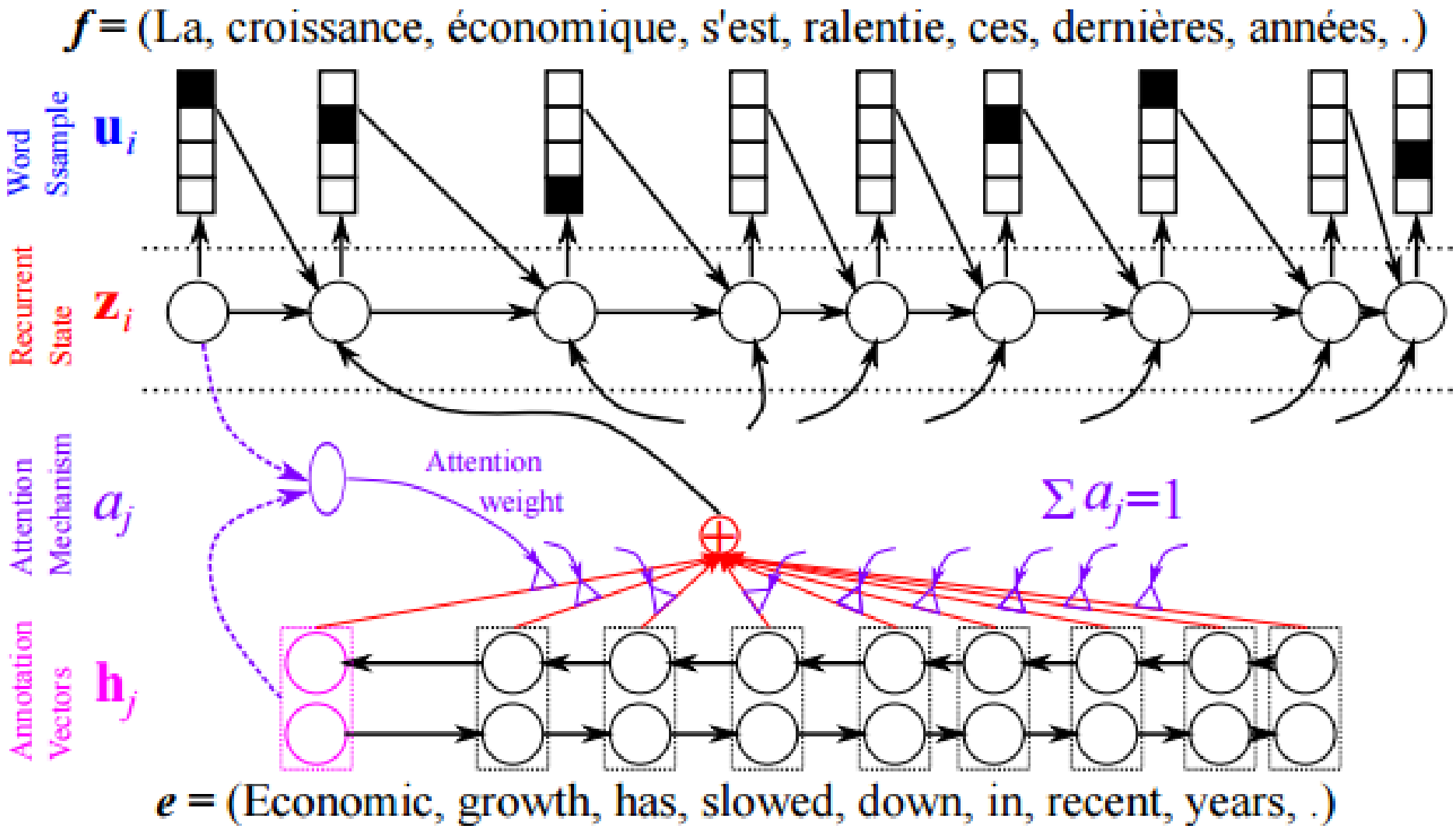


Figure 1: The graphical illustration of the proposed model trying to generate the t-th target word y_t given a source sentence $(x_1, x_2, \ldots, x_T)$.

The full picture where "attention" is situated

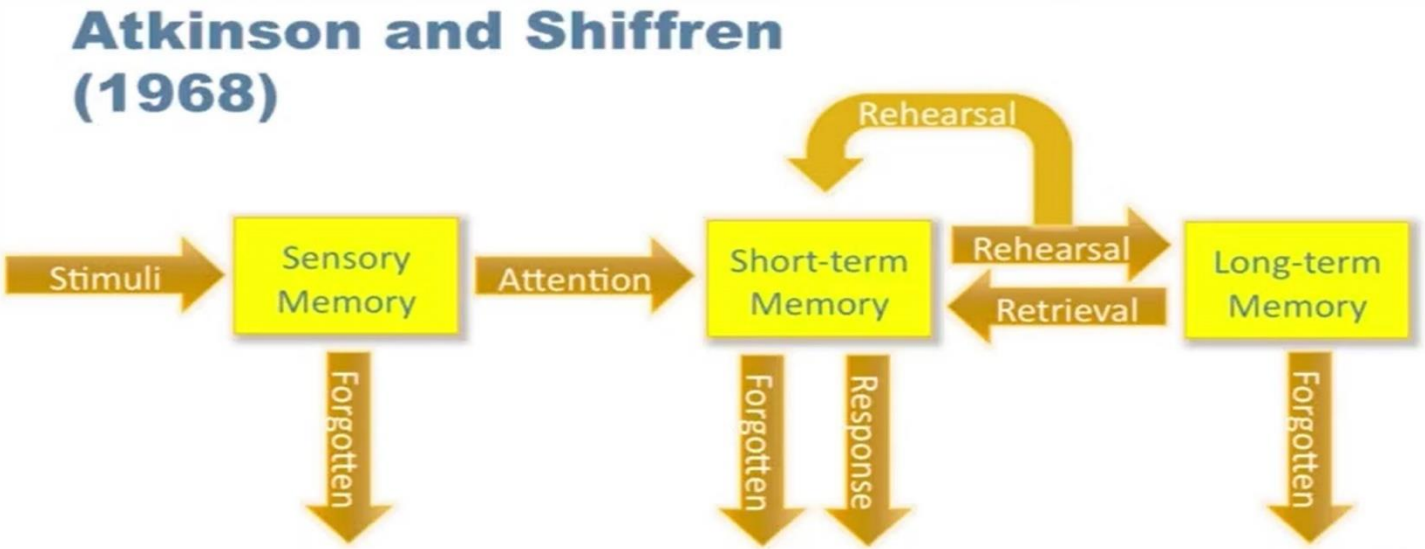
(slide from Y. Bengio)

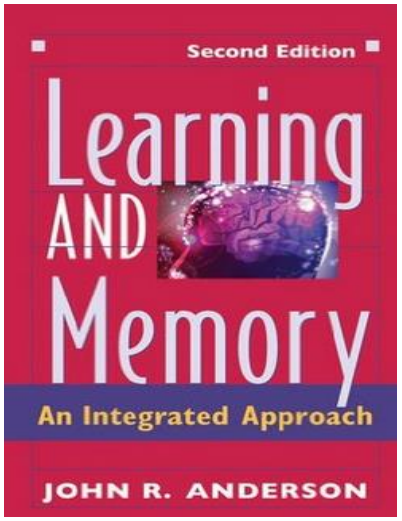


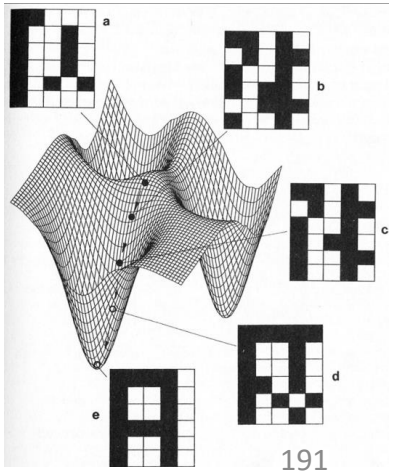
Microsoft Research

Popular theories of human memory/attention

The attention and memory models discussed so far are far from human memory/attention Mechanisms (https://en.wikipedia.org/wiki/Atkinson%E2%80%93Shiffrin_memory_model):



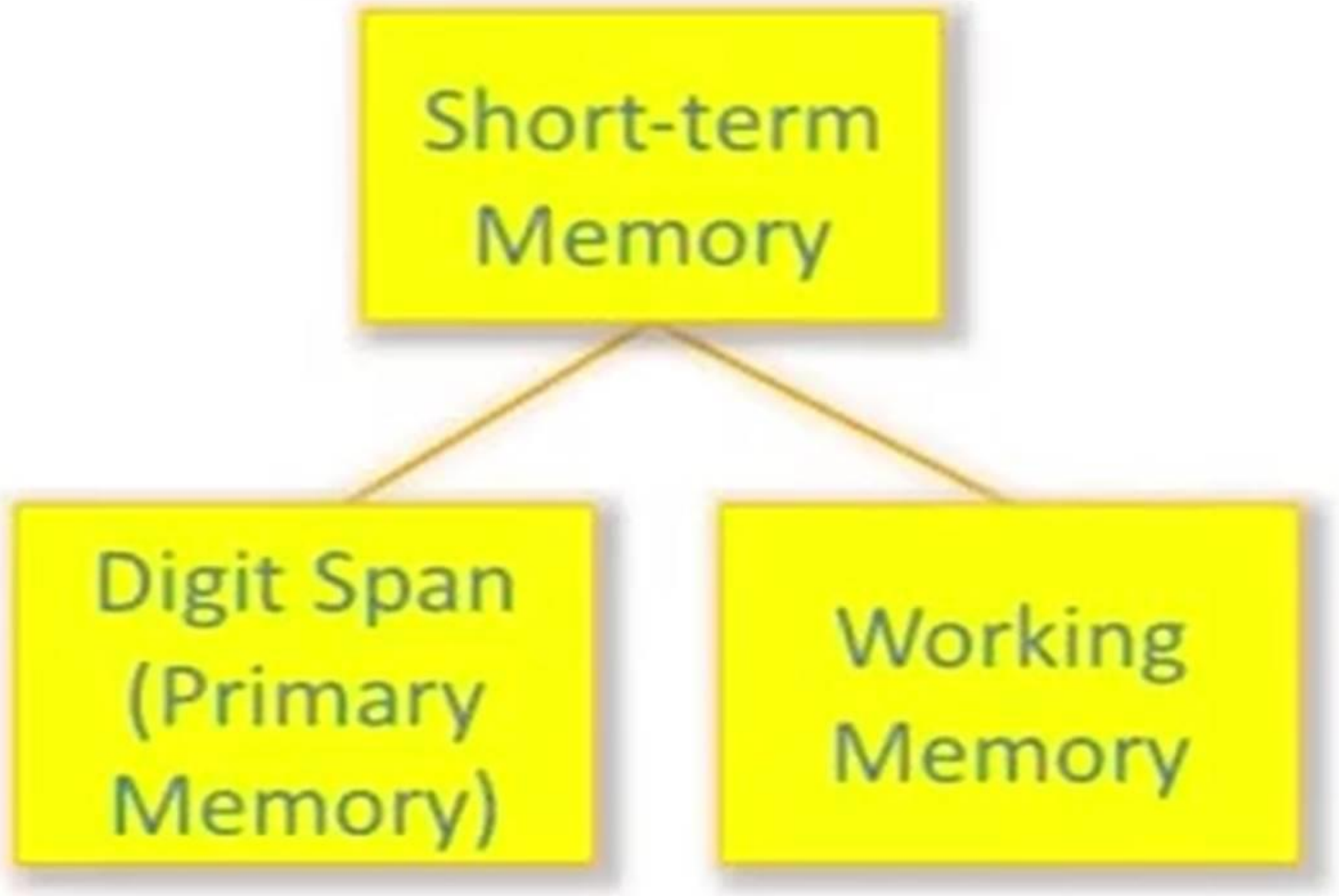




Hopfield nets store (associative) memories as attractors of the dynamic network

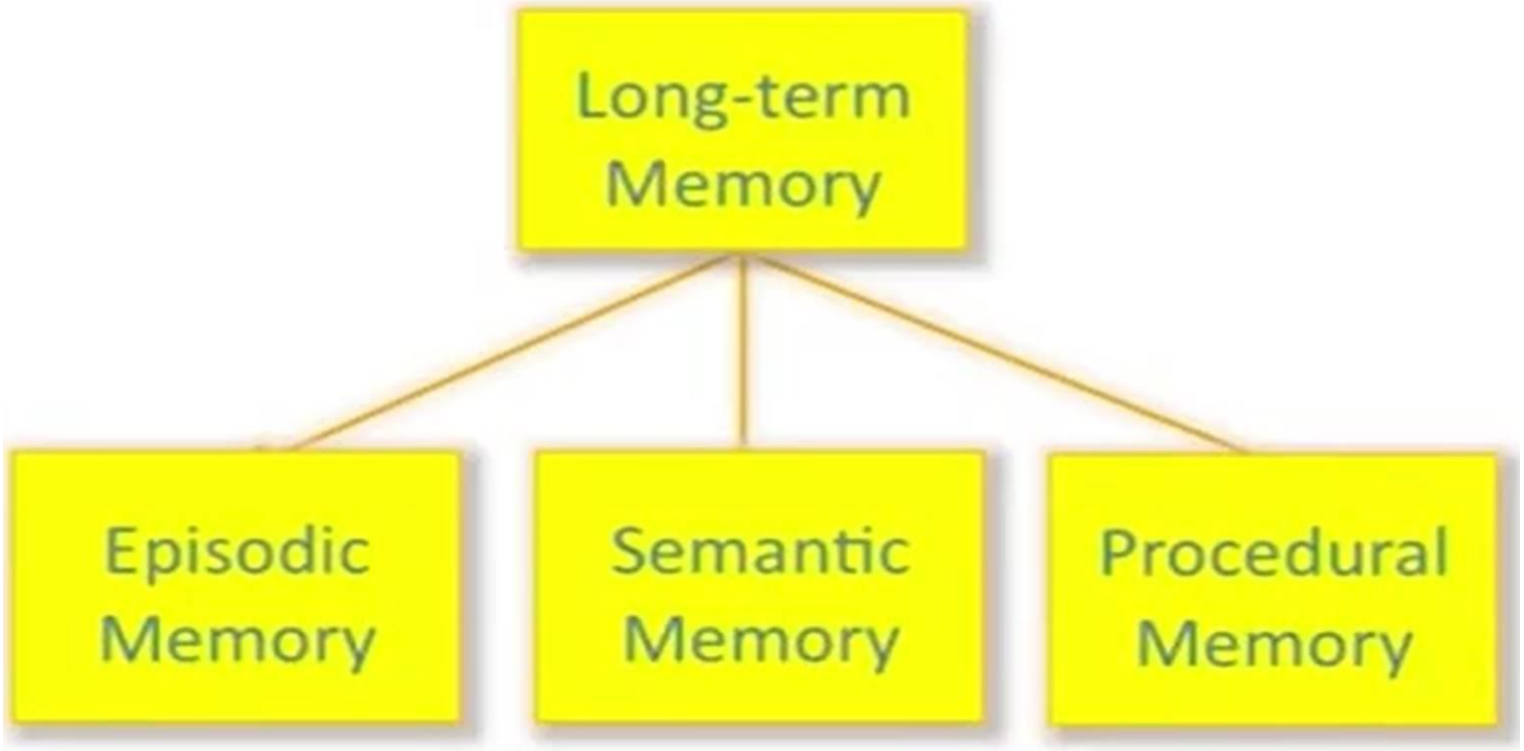
Microsoft Research

LSTM mainly models short-term memory



LSTM does not model long-term memory well

- LSTM makes short-term memory lasting via a simple “unit-loop” mechanism, very different from long-term memory in human cognition
- Review of a very recent modeling study on episodic and semantic memories, extending the basic LSTM formulation



Georgia Tech School of Psychology
College of Sciences

TYPES OF LONG-TERM MEMORY

Episodic Memory	"When was the last time you rode a bicycle?"	Internal Diary	Retrieval by conscious contextual cues
Semantic Memory	"What is a bicycle?"	Internal Encyclopedia	Retrieval by conscious conceptual cues
Procedural Memory	Bike-riding Skill	Internal Computer Program	Unconscious automatic play-back

Slides from: Coursera, 2014

Ask Me Anything: Dynamic Memory Networks for Natural Language Processing

Ankit Kumar Ozan Irsoy Jonathan Su James Bradbury Robert English
Brian Pierce Peter Ondruska Mohit Iyyer Ishaan Gulrajani Richard Socher
firstname@metamind.io
MetaMind
Palo Alto, CA

Abstract

Most tasks in natural language processing can be cast into question answering (QA) problems over language input. We introduce the dynamic memory network (DMN), a unified neural network framework which processes input sequences and questions, forms semantic and episodic memories, and generates relevant answers. Questions trigger an iterative attention process which allows the model to condition its attention on the result of previous iterations. These results are then reasoned over in a hierarchical recurrent sequence model to generate answers.

Going beyond L-STM --- towards more realist long-term memory (episodic & semantic)

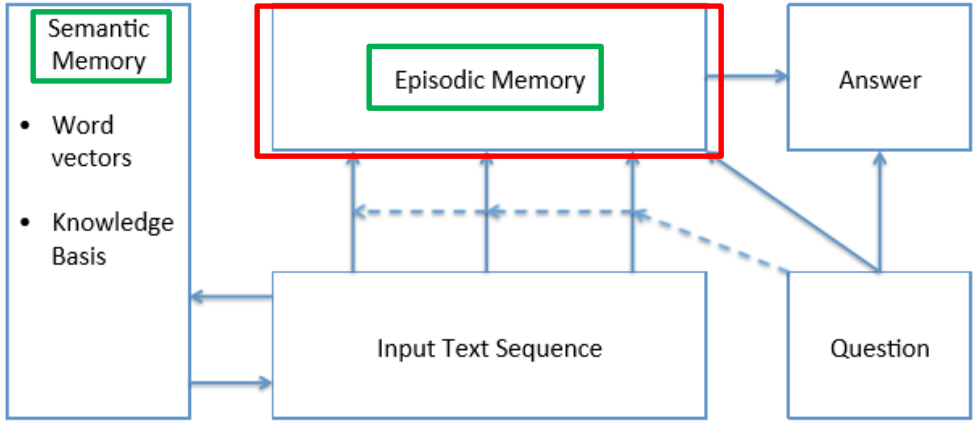


Figure 2: Overview of DMN modules. Communication between them is indicated by arrows and uses only vector representations. Questions trigger gates which allow vectors for certain input words or sentences to be given to the episodic memory module. The final state of the episodic memory is the input to the answer module.

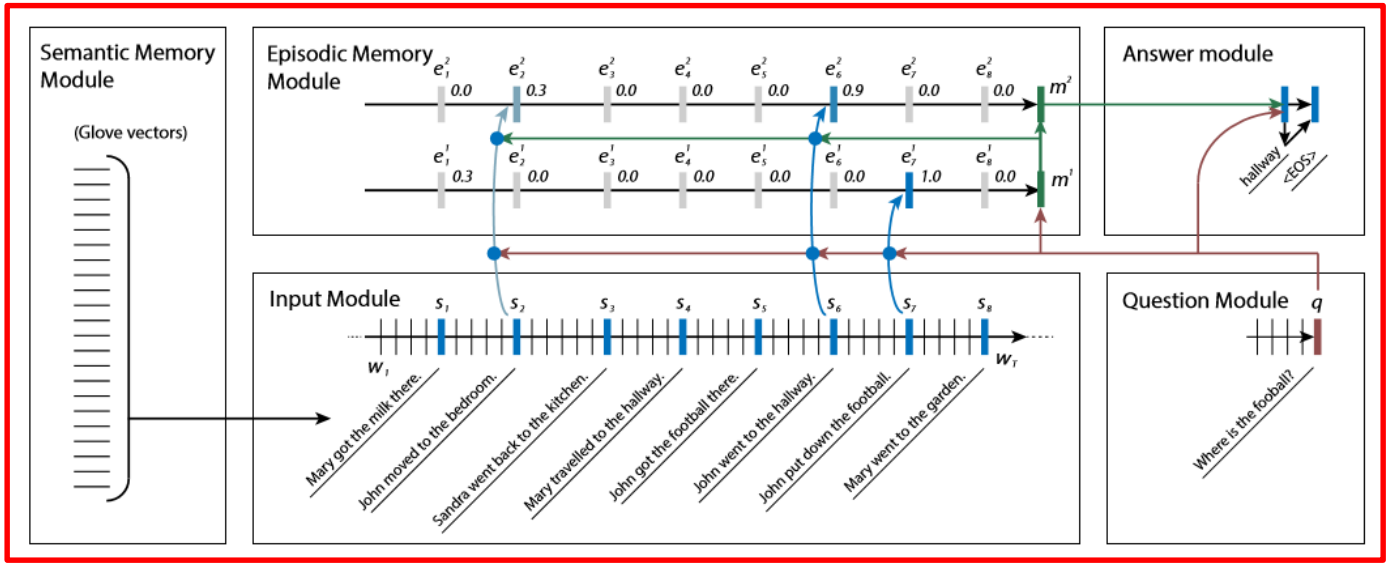


Figure 3: Real example of an input sentence sequence and the attention gates that are triggered by a specific question. Gate values g_t^i are shown above the corresponding vectors. The gates change with each search over inputs. We do not draw connections for gates that are close to zero. See Section 4.1 for details on the dataset that this example comes from.

Towards Modeling Cognitive Functions: Memory and Attention

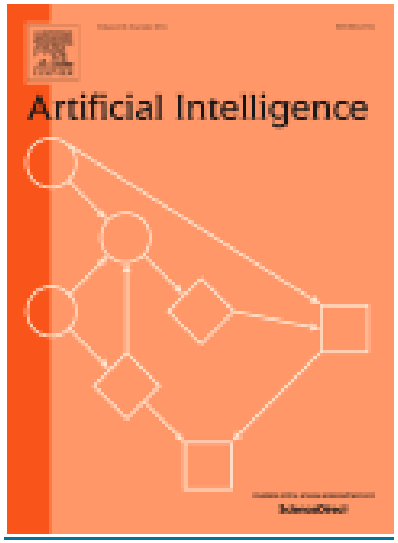
-seq-to-seq learning via LSTM with attention mechanism
-memory nets and neural Turing machines
-dynamic memory nets
-from seq2seq to seq2struct & to struct2struct

Review: embedding in the form of “flat” vectors

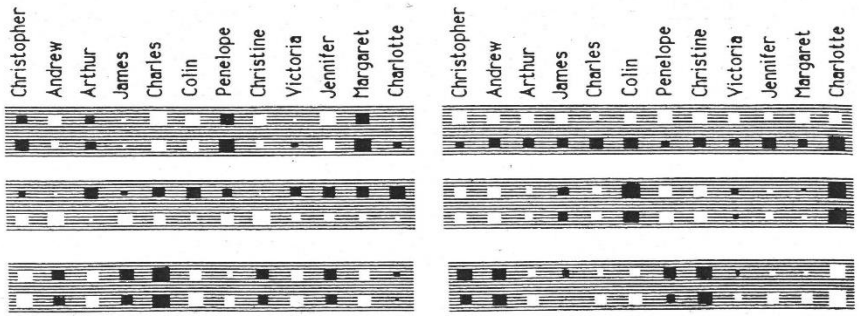
- **A linguistic or physical entity or a simple “relation”**

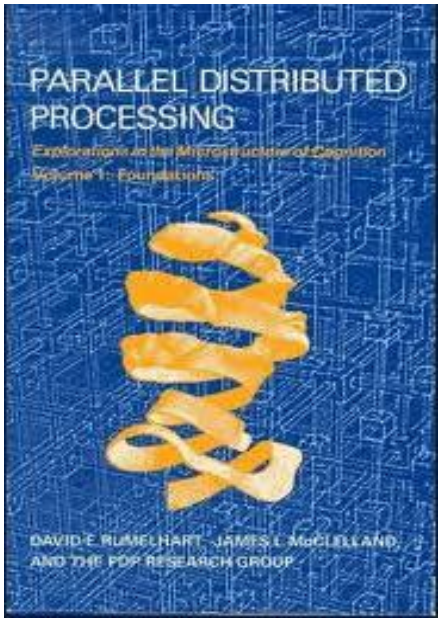
mapping via distributed representations by NN

A low-dim continuous-space vector or embedding



Special Issue, vol. 46 (1990)
Connectionist Symbol Processing
(4 articles)





PDP book, 1986

Extension: "flat" vectors →structures (tree/graph)

- Structured embedding vectors via tensor-product rep.

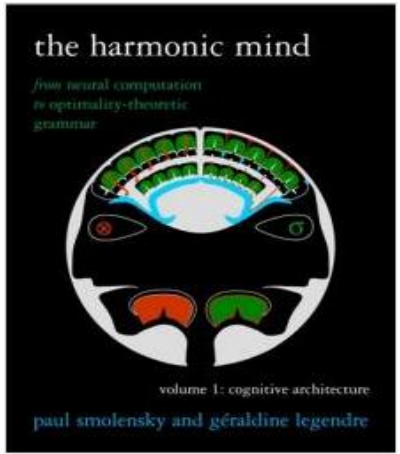
symbolic semantic parse tree (complex relation)

Then, reasoning in symbolic-space (traditional AI) can be beautifully carried out in the continuous-space in human cognitive and neural-net (i.e., connectionist) terms

Smolensky & Legendre: **The Harmonic Mind, MIT Press, 2006**

From Neural Computation to Optimality-Theoretic Grammar

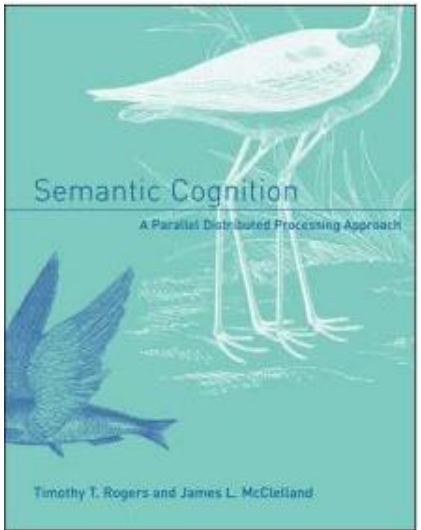
Volume I: Cognitive Architecture; Volume 2: Linguistic Implications

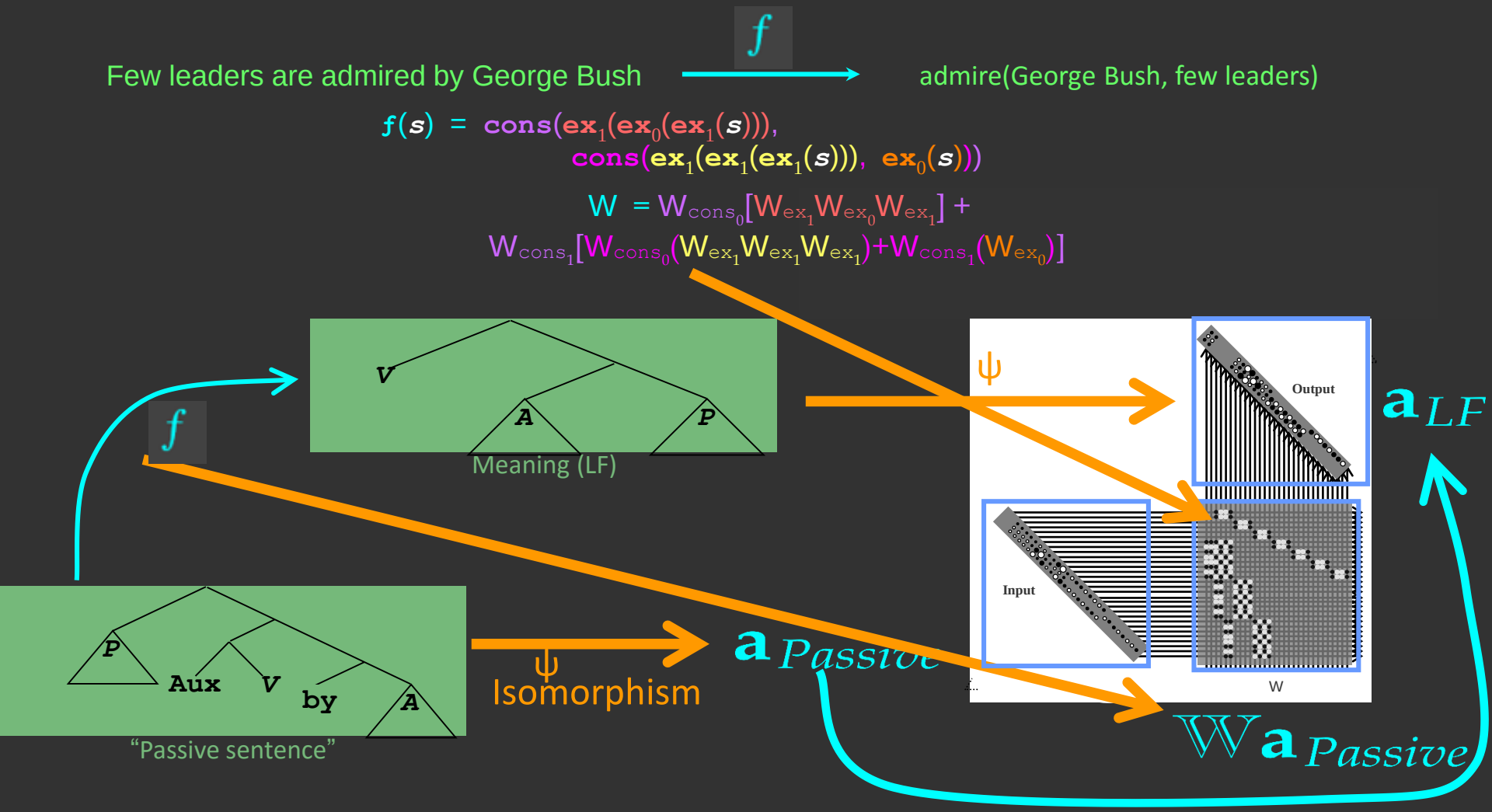


Rogers & McClelland

Semantic Cognition

MIT Press, 2006





Slide from Paul Smolensky, 2015

Summary & Perspective

- Speech recognition is the first success example of deep learning at industry scale
- Deep learning is very effective in speech recognition, speech translation (Skype Translator), image recognition (Onedrive Image tagging), image captioning, language understanding (Cortana), semantic intelligence, multimodal and multitask learning, web search, advertising, entity search (Insights for MS Office), user and business activity prediction, etc.
- Enabling factors:
  - Big datasets for training deep models
  - Powerful GPGPU computing
  - Innovations in deep learning architectures and algorithms
    - How to discover distant supervision signals free from human labeling
    - How to build deep learning systems grounded on exploiting such “smart” signals (example: DSSM)
    - …

Summary & Perspective

- Speech recognition: **all** low-hanging fruits are taken
  - i.e. **more** innovation and hard work needed than before
- Image recognition: **most** low-hanging fruits are taken
- Natural Language: does not seem there is much low-hanging fruit there
  - i.e. **even more** innovation and hard work needed than before
- Big data analytics (e.g. user behavior, business activities, etc):

  - A new frontier
- Small data: deep learning may still win (e.g. 2012 Kaggle's drug discovery)
- Perceptual data: deep learning methods always win, and win big
- Be careful: data with adversarial nature; data with odd variability

Issues for "Near" Future of Deep Learning

- For perceptual tasks (e.g. speech, image/video, gesture, etc.)
  - With supervised data: what will be the limit for growing accuracy wrt increasing amounts of labeled data?
  - Beyond this limit or when labeled data are exhausted or non-economical to collect, will novel and effective unsupervised deep learning emerge and what will they be (e.g. deep generative models)?
  - Many new innovations are to come, likely in the area of unsupervised learning
- For cognitive tasks (e.g. natural language, reasoning, knowledge, decision making, etc.)
  - Will supervised deep learning (e.g. MT) beat the non-deep-learning state of the art like speech/image recognition?
  - How to distill/exploit "distant" supervision signals for supervised deep learning?
  - Will dense vector embedding be sufficient for language? Do we really need to directly encode and recover syntactic/semantic structure of language?
  - Even more new innovations are to come, likely in the area of new architectures and learning methods pertaining to distant supervised learning

The Future of Deep Learning

- Continued rapid progress in language processing methods and applications by both industry and academia
- From image to video processing/understanding
- From supervised learning (huge success already) to **unsupervised** learning (not much success yet but ideas abound)
- From perception to cognition
  - More exploration of attention modeling
  - Combine representation learning with complex knowledge extraction & reasoning
  - Modeling human memory functions more faithfully
  - Learning to act and control (deep reinforcement learning)
- Successes in business applications will propel more rapid advances in deep learning (positive feedbacks)

Additional References

- Auli, M., Galley, M., Quirk, C. and Zweig, G., 2013. Joint language and translation modeling with recurrent neural networks. In EMNLP.
- Auli, M., and Gao, J., 2014. Decoder integration and expected bleu training for recurrent neural network language models. In ACL.
- Baker, J., Li Deng, Jim Glass, S. Khudanpur, C.-H. Lee, N. Morgan, and D. O'Shgughnessy, Research Developments and Directions in Speech Recognition and Understanding, Part 1, in *IEEE Signal Processing Magazine*, vol. 26, no. 3, pp. 75-80, 2009.
- Baker, J., Li Deng, S. Khudanpur, C.-H. Lee, J. Glass, and N. Morgan, Updated MINDS Report on Speech Recognition and Understanding, in *IEEE Signal Processing Magazine*, vol. 26, no. 4, July 2009.
- Bengio, Y., 2009. Learning deep architectures for AI. Foundumental Trends Machine Learning, vol. 2.
- Bengio, Y., Courville, A., and Vincent, P. 2013. Representation learning: A review and new perspectives. IEEE Trans. PAMI, vol. 38, pp. 1798-1828.
- Bengio, Y., Ducharme, R., and Vincent, P., 2000. A Neural Probabilistic Language Model, in NIPS.
- Bridle, J., L. Deng, J. Picone, H. Richards, J. Ma, T. Kamm, M. Schuster, S. Pike, R. Reagan, An investigation of segmental hidden dynamic models of speech coarticulation for automatic speech recognition. *Final Report for 1998 Workshop on Langauge Engineering, CLSP* (Johns Hopkins, 1998.
- Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., and Kuksa, P., 2011. Natural language processing (almost) from scratch. in JMLR, vol. 12.
- Dahl, G., Yu, D., Deng, L., and Acero, 2012. A. Context-dependent, pre-trained deep neural networks for large vocabulary speech recognition, IEEE Trans. Audio, Speech, & Language Proc., Vol. 20 (1), pp. 30-42.
- Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T., and Harshman, R. 1990. Indexing by latent semantic analysis. J. American Society for Information Science, 41(6): 391-407
- Deng, L. A dynamic, feature-based approach to the interface between phonology & phonetics for speech modeling and recognition, *Speech Communication*, vol. 24, no. 4, pp. 299-323, 1998.
- Deng, L. Computational Models for Speech Production, in Computational Models of Speech Pattern Processing, pp. 199-213, Springer Verlag, 1999.
- Deng, L. Switching Dynamic System Models for Speech Articulation and Acoustics, in Mathematical Foundations of Speech and Language Processing, vol. 138, pp. 115-134, Springer Verlag, 2003.
- Deng, L., K. Hassanein, M. Elmasry, Analysis of the correlation structure for a neural predictive model with application to speech recognition. Neural Networks. **7**(2), 331–339, 1994.
- Deng L., G. Ramsay, and D. Sun, Production models as a structural basis for automatic speech recognition," Speech Communication (special issue on speech production modeling), in *Speech Communication*, vol. 22, no. 2, pp. 93-112, August 1997.
- Deng L. and J. Ma, Spontaneous Speech Recognition Using a Statistical Coarticulatory Model for the Vocal Tract Resonance Dynamics, *Journal of the Acoustical Society of America*, 2000.
- Deng, L. and O'Shaughnessy, O. *Speech Processing—A Dynamic and Optimization-Oriented Approach*, Marcel Dekker, New York, 2003
- Deng L. and Yu, L. DEEP LEARNING: Methods and Applications, NOW Publishings, 2014.
- Deng L. and Xiao Li, Machine Learning Paradigms for Speech Recognition: An Overview, in *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 5, pp. 1060-1089, May 2013.
- Deng L. and Xuedong Huang, Challenges in Adopting Speech Recognition, in *Communications of the ACM*, vol. 47, no. 1, pp. 11-13, January 2004.
- Deng, L., Seltzer, M., Yu, D., Acero, A., Mohamed, A., and Hinton, G., Binary Coding of Speech Spectrograms Using a Deep Auto-encoder, Interspeech, 2010.
- Deng, L., Tur, G, He, X, and Hakkani-Tur, D. 2012. Use of kernel deep convex networks and end-to-end learning for spoken language understanding, Proc. IEEE Workshop on Spoken Language Technologies.
- Deng, L., Yu, D. and Acero, A. 2006. Structured speech modeling, IEEE Trans. on Audio, Speech and Language Processing, vol. 14, no. 5, pp. 1492-1504.
- Deng, L., Geoffrey Hinton, and Brian Kingsbury, New types of deep neural network learning for speech recognition and related applications: An overview, ICASSP, 2013.
- Deng L. and Dong Yu, Use of Differential Cepstra as Acoustic Features in Hidden Trajectory Modeling for Phonetic Recognition, ICASSP, 2007.
- Deng, L., Xiaodong He, and Jianfeng Gao, Deep Stacking Networks for Information Retrieval, ICASSP, 2013.
- Deng L. and Dong Yu, Deep Convex Network: A Scalable Architecture for Speech Pattern Classification, in *Interspeech*, 2011.
- Deng, L., Yu, D., and Platt, J. 2012. Scalable stacking and learning for building deep architectures, Proc. ICASSP.
- Deoras, A., and Sarikaya, R., 2013. Deep belief network based semantic taggers for spoken language understanding, in INTERSPEECH.
- Devlin, J., Zbib, R., Huang, Z., Lamar, T., Schwartz, R., and Makhoul, J., 2014. Fast and Robust Neural Network Joint Models for Statistical Machine Translation, ACL.

Additional References

- Frome, A., Corrado, G., Shlens, J., Bengio, S., Dean, J., Ranzato, M., and Mikolov, T., 2013. DeViSE: A Deep Visual-Semantic Embedding Model, Proc. NIPS.
- Gao, J., He, X., Yih, W-t., and Deng, L. 2014a. Learning continuous phrase representations for translation modeling. In ACL.
- Gao, J., He, X., and Nie, J-Y. 2010. Clickthrough-based translation models for web search: from word models to phrase models. In CIKM.
- Gao, J., Pantel, P., Gamon, M., He, X., and Deng, L. 2014. Modeling interestingness with deep neural networks. In EMNLP
- Gao, J., Toutanova, K., Yih., W-T. 2011. Clickthrough-based latent semantic models for web search. In SIGIR.
- Gao, J., Yuan, W., Li, X., Deng, K., and Nie, J-Y. 2009. Smoothing clickthrough data for web search ranking. In SIGIR.
- Gao, J., and He, X. 2013. Training MRF-based translation models using gradient ascent. In NAACL-HLT.
- Graves, A., Jaitly, N., and Mohamed, A., 2013a. Hybrid speech recognition with deep bidirectional LSTM, Proc. ASRU.
- Graves, A., Mohamed, A., and Hinton, G., 2013. Speech recognition with deep recurrent neural networks, Proc. ICASSP.
- He, X. and Deng, L., 2013. Speech-Centric Information Processing: An Optimization-Oriented Approach, in Proceedings of the IEEE.
- He, X. and Deng, L., 2012. Maximum Expected BLEU Training of Phrase and Lexicon Translation Models , ACL.
- He, X., Deng, L., and Chou, W., 2008. Discriminative learning in sequential pattern recognition, Sept. IEEE Sig. Proc. Mag.
- Hinton, G., Deng, L., Yu, D., Dahl, G., Mohamed, A., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T., and Kingsbury, B., 2012. Deep Neural Networks for Acoustic Modeling in Speech Recognition, IEEE Signal Processing Magazine, vol. 29, no. 6, pp. 82-97.
- Hinton, G., and Salakhutdinov, R., 2010. Discovering binary codes for documents by learning deep generative models. Topics in Cognitive Science.
- Hu, Y., Auli, M., Gao, Q., and Gao, J. 2014. Minimum translation modeling with recurrent neural networks. In EACL.
- Huang, E., Socher, R., Manning, C, and Ng, A. 2012. Improving word representations via global context and multiple word prototypes, Proc. ACL.
- Huang, P., He, X., Gao, J., Deng, L., Acero, A., and Heck, L. 2013. Learning deep structured semantic models for web search using clickthrough data. In CIKM.
- Hutchinson, B., Deng, L., and Yu, D., 2012. A deep architecture with bilinear modeling of hidden representations: Applications to phonetic recognition, Proc. ICASSP.
- Hutchinson, B., Deng, L., and Yu, D., 2013. Tensor deep stacking networks, IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 35, pp. 1944 - 1957.
- Kiros, R., Zemel, R., and Salakhutdinov, R. 2013. Multimodal Neural Language Models, Proc. NIPS Deep Learning Workshop.
- Koehn, P. 2009. Statistical Machine Translation. Cambridge University Press.
- Krizhevsky, A., Sutskever, I, and Hinton, G., 2012. ImageNet Classification with Deep Convolutional Neural Networks, NIPS.
- Le, H-S, Oparin, I., Allauzen, A., Gauvain, J-L., Yvon, F., 2013. Structured output layer neural network language models for speech recognition, IEEE Transactions on Audio, Speech and Language Processing.
- LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. 1998. Gradient-based learning applied to document recognition, Proceedings of the IEEE, Vol. 86, pp. 2278-2324.
- Lee, L., H. Attias, Li Deng, and P. Fieguth, A Multimodal Variational Approach to Learning and Inference in Switching State Space Models, ICASSP, 2004.
- Ma, J. and L. Deng, Target-directed mixture dynamic models for spontaneous speech recognition. IEEE Trans. Audio Speech Process. 12(1), 47–58, 2004
- Mesnil, G., He, X., Deng, L., and Bengio, Y., 2013. Investigation of Recurrent-Neural-Network Architectures and Learning Methods for Spoken Language Understanding, in Interspeech.
- Mikolov, T. 2012. Statistical Language Models based on Neural Networks, Ph.D. thesis, Brno University of Technology.
- Mikolov, T., Chen, K., Corrado, G., and Dean, J. 2013. Efficient estimation of word representations in vector space, Proc. ICLR.
- Mikolov, T., Kombrink,. S., Burget, L., Cernocky, J., Khudanpur, S., 2011. Extensions of Recurrent Neural Network LM. ICASSP.
- Mikolov, T., Yih, W., Zweig, G., 2013. Linguistic Regularities in Continuous Space Word Representations. In NAACL-HLT.
- Mohamed, A., G. Dahl, G. Hinton, Acoustic modeling using deep belief networks. IEEE Trans. Audio Speech Lang. Process. **20**(1), 14–22, 2012 & 2009
- Mohamed, A., Yu, D., and Deng, L. 2010. Investigation of full-sequence training of deep belief networks for speech recognition, Proc. Interspeech.
- Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., and Ng, A. 2011. Multimodal deep learning, Proc. ICML.

Additional References

- Picone, P., S. Pike, R. Regan, T. Kamm, J. Bridle, L. Deng, Z. Ma, H. Richards, M. Schuster, 1999. Initial evaluation of hidden dynamic models on conversational speech, ICASSP.
- Robinson, T. An application of recurrent nets to phone probability estimation. IEEE Trans. Neural Networks. **5**(2), 298–305, 1994.
- Sainath, T., Mohamed, A., Kingsbury, B., and Ramabhadran, B. 2013. Convolutional neural networks for LVCSR, Proc. ICASSP.
- Salakhutdinov R., and Hinton, G., 2007 Semantic hashing. in Proc. SIGIR Workshop Information Retrieval and Applications of Graphical Models
- Schwenk, H., Dchelotte, D., Gauvain, J-L., 2006. Continuous space language models for statistical machine translation, in COLING-ACL
- Seide, F., Li, G., and Yu, D. 2011. Conversational speech transcription using context-dependent deep neural networks, Proc. Interspeech
- Shen, Y., He, X., Gao, J., Deng, L., and Mesnil, G. 2014. Learning Semantic Representations Using Convolutional Neural Networks for Web Search, in Proceedings of WWW.
- Shen, Y., He, X., Gao, J., Deng, L., and Mesnil, G. 2014. A convolutional latent semantic model for web search. CIKM
- Shen, X., L. Deng, Maximum likelihood in statistical estimation of dynamical systems: decomposition algorithm and simulation results. Signal Process. **57**, 65–79, 1997.
- Socher, R., Huval, B., Manning, C., Ng, A., 2012. Semantic compositionality through recursive matrix-vector spaces. In EMNLP.
- Socher, R., Lin, C., Ng, A., and Manning, C. 2011. Learning continuous phrase representations and syntactic parsing with recursive neural networks, Proc. ICML.
- Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C., Ng A., and Potts. C. 2013. Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank, Proc. EMNLP
- Song, X. He, X., Gao. J., and Deng, L. 2014. Learning Word Embedding Using the DSSM. MSR Tech Report.
- Song, Y., Wang, H., and He, X., 2014. Adapting Deep RankNet for Personalized Search. Proc. WSDM.
- Tur, G., Deng, L., Hakkani-Tur, D., and He, X., 2012. Towards Deeper Understanding Deep Convex Networks for Semantic Utterance Classification, in ICASSP.
- Togneri, R., L. Deng, Joint state and parameter estimation for a target-directed nonlinear dynamic system model. IEEE Trans. Signal Process. **51**(12), 3061–3070, 2003.
- Vinyals, O., Y. Jia, Li Deng, and Trevor Darrell, Learning with Recursive Perceptual Representations, NIPS, 2012.
- Wright, S., Kanevsky, D., Deng, L., He, X., Heigold, G., and Li, H. Optimization Algorithms and Applications for Speech and Language Processing, in IEEE Transactions on Audio, Speech, and Language Processing, vol. 21, no. 11. 2013.
- Yao, K., Zweig, G., Hwang, M-Y. , Shi, Y., Yu, D., 2013. Recurrent neural networks for language understanding, submitted to Interspeech.
- Yann, D., Tur, G., Hakkani-Tur, D., Heck, L., 2014. Zero-Shot Learning and Clustering for Semantic Utterance Classification Using Deep Learning, in ICLR.
- Yu, D. and Deng, L. Discriminative pretraining of deep neural networks, US Patent, 2013.
- Yu, D., Deng, L., G. Dahl, Roles of pre-training and fine-tuning in context-dependent DBN-HMMs for real-world speech recognition, *NIPS Workshop* 2010.

Thank You