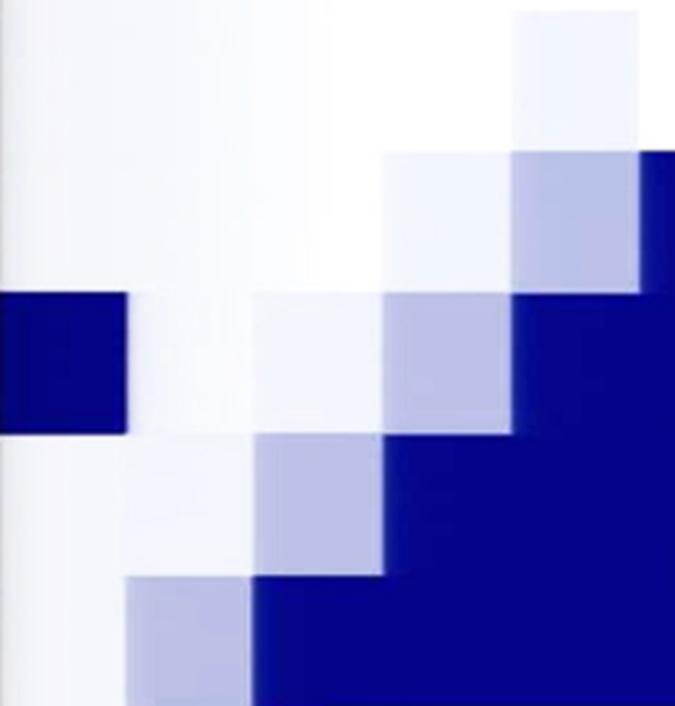


Microsoft®

Research

Each year Microsoft Research hosts hundreds of influential speakers from around the world including leading scientists, renowned experts in technology, book authors, and leading academics, and makes videos of these lectures freely available.

2011 © Microsoft Corporation. All rights reserved.



Deep Neural Networks for Speech and Image Processing

Alex Acero
Microsoft Research
May 24th, 2012

Agenda

- Intro to neuroscience
- Artificial Neural Networks
- Deep Neural Networks
- Application to Speech Recognition

Carnegie Mellon 1990



Neurons

- Human brain:

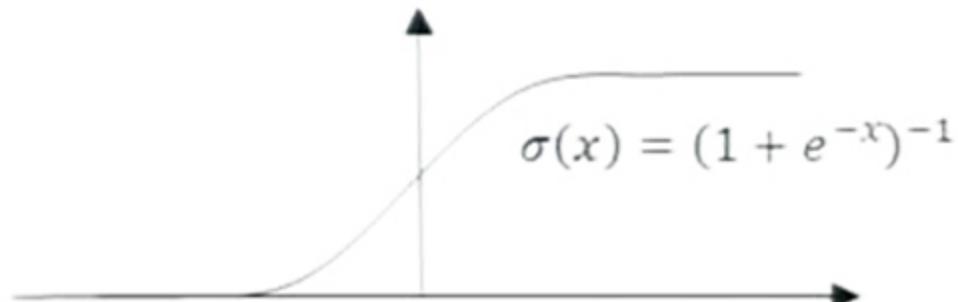
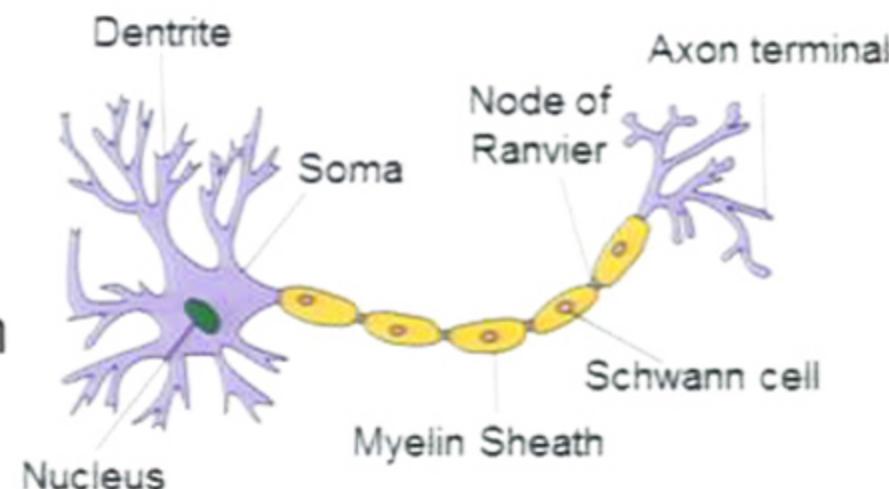
- 100 billion neurons (10^{11})
 - ~7000 synapses per neuron

- Neurons are

- Non-deterministic
 - slow: 1ms

- Sigmoid nonlinearity

$$\sigma\left(\sum w_i v_i - b\right)$$



Neuroscience



Neuroscience

■ Mysteries in neuroscience

- What's the difference between a human brain and that of a monkey?
- Can we cure Alzheimer?



Neuroscience

- Mysteries in neuroscience

- What's the difference between a human brain and that of a monkey?
 - Can we cure Alzheimer?

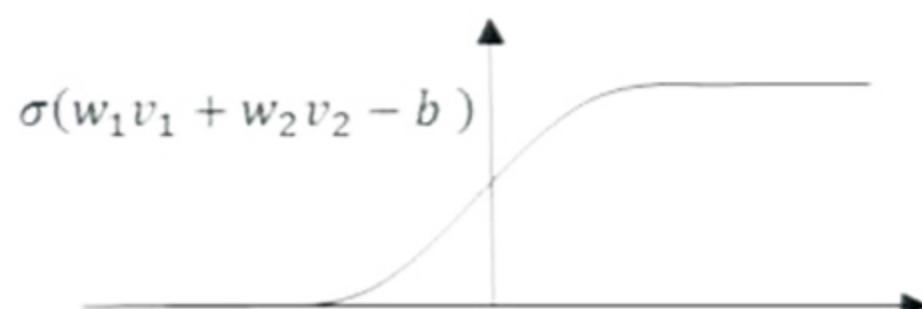
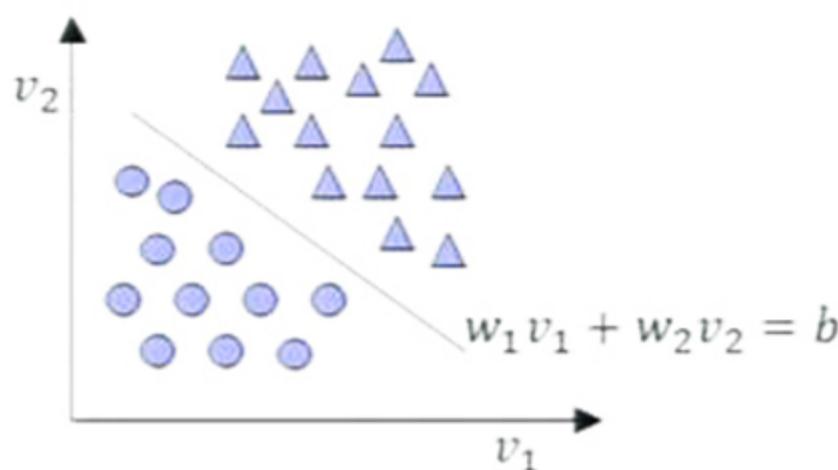
- Neurons pick up patterns

- Hebbian rule: “Neurons that fire together, wire together”



Perceptron: the birth of ANN

- Rosenblatt (1958)
- Linear classifier



Perceptron learning

■ Minimize errors

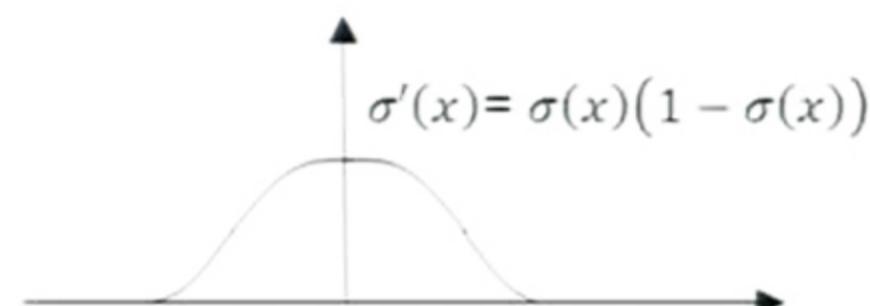
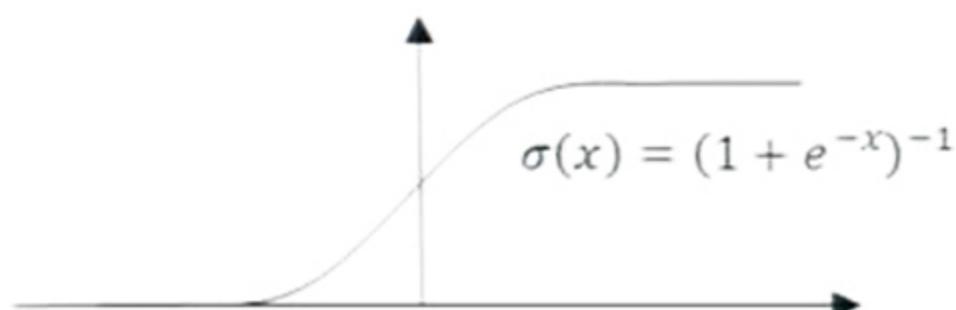
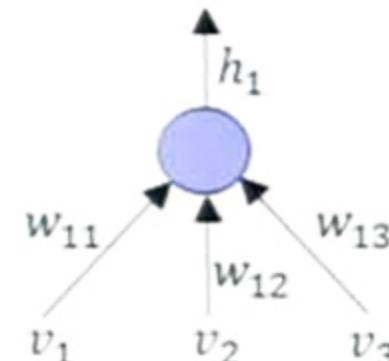
$$E = \sum_t e_1^2(t)$$

$$e_1 = h_1 - l_1$$

$$h_1 = \sigma \left(\sum_{i=1}^3 w_{1i} v_i \right)$$

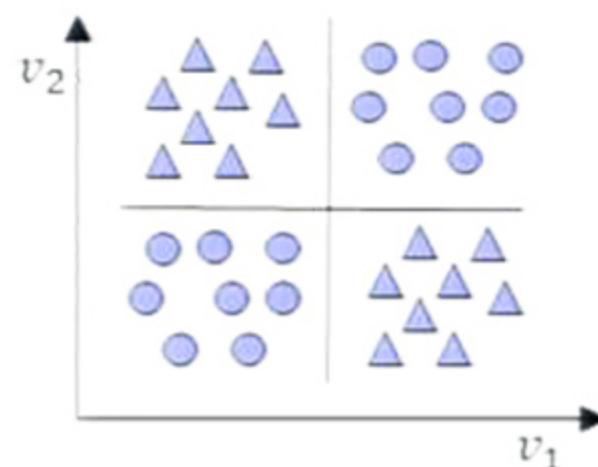
$$w_{1j}^{k+1} = w_{1j}^k - \alpha \frac{\partial E}{\partial w_{1j}}$$

$$\frac{\partial E}{\partial w_i} = 2 \sum_t e_1(t) h_1(t) [1 - h_1(t)] v_i(t)$$

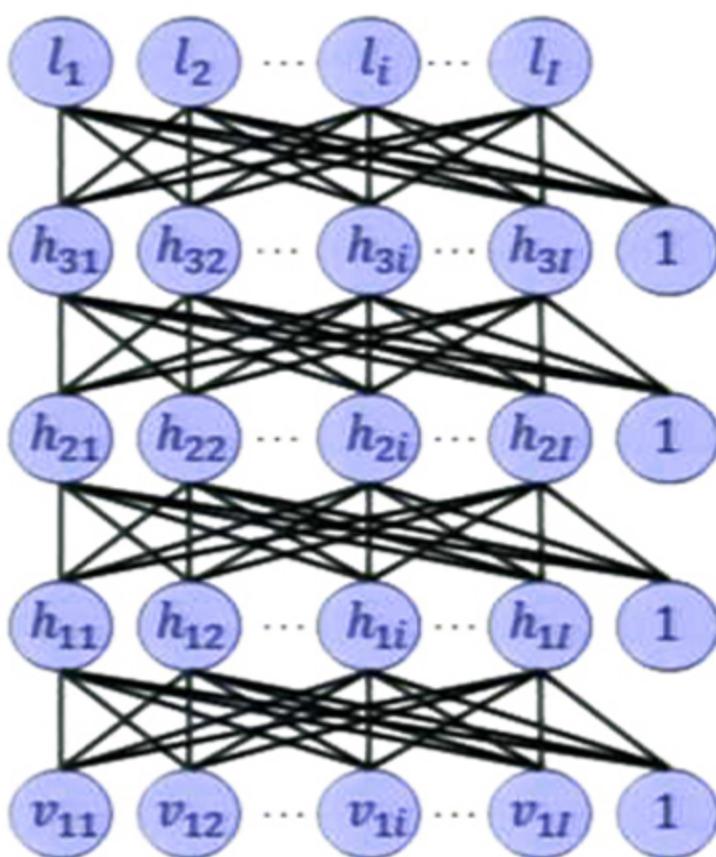


Neural Networks Winter starts

- Minsky and Papert (1969)
 - XOR cannot be modeled with a perceptron



Artificial Neural Networks



1948: Alan Turing proposes artificial neural networks (ANN)

Back propagation

- Bryson and Hoback (1969) invent it
- Hinton (1974) rediscovers it

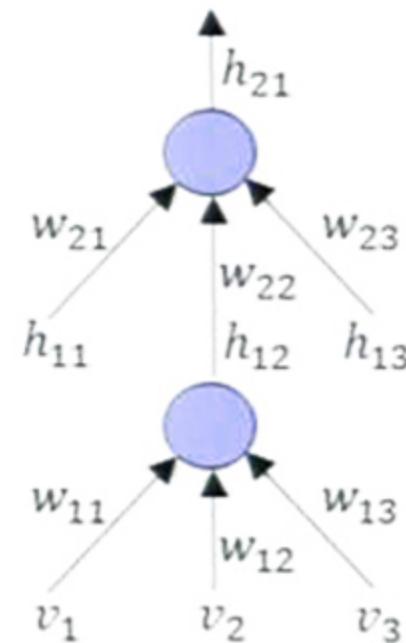
$$E = \sum_t e_1^2(t) \quad e_1 = h_{21} - l_1$$

$$h_{21} = \sigma \left(\sum_{i=1}^3 w_{2i} h_{1i} \right) \quad h_{12} = \sigma \left(\sum_{i=1}^3 w_{1i} v_i \right)$$

$$w_{1i}^{k+1} = w_{1i}^k - \alpha \frac{\partial E}{\partial w_{1i}}$$

$$\frac{\partial E}{\partial w_{1i}} = 2 \sum_t e_1(t) h_{21}(t) [1 - h_{21}(t)] w_{22} \frac{\partial h_{12}(t)}{\partial w_{1i}}$$

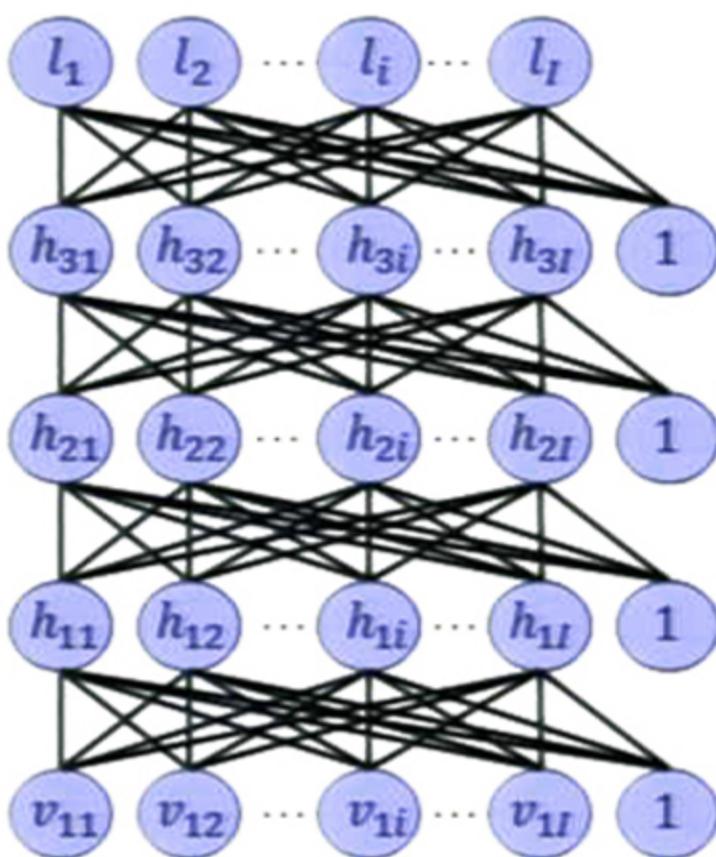
$$\frac{\partial h_{12}(t)}{\partial w_{1i}} = h_{12}(t) [1 - h_{12}(t)] v_i(t)$$



Chain rule



Artificial Neural Networks



1948: Alan Turing proposes artificial neural networks (ANN)

Back propagation

- Bryson and Hoback (1969) invent it
- Hinton (1974) rediscovers it

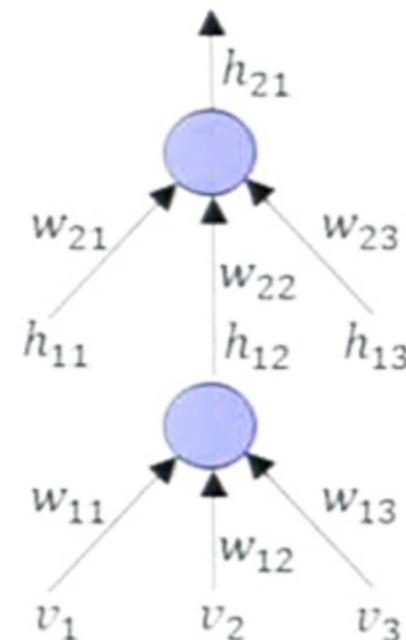
$$E = \sum_t e_1^2(t) \quad e_1 = h_{21} - l_1$$

$$h_{21} = \sigma \left(\sum_{i=1}^3 w_{2i} h_{1i} \right) \quad h_{12} = \sigma \left(\sum_{i=1}^3 w_{1i} v_i \right)$$

$$w_{1i}^{k+1} = w_{1i}^k - \alpha \frac{\partial E}{\partial w_{1i}}$$

$$\frac{\partial E}{\partial w_{1i}} = 2 \sum_t e_1(t) h_{21}(t) [1 - h_{21}(t)] w_{22} \frac{\partial h_{12}(t)}{\partial w_{1i}}$$

$$\frac{\partial h_{12}(t)}{\partial w_{1i}} = h_{12}(t) [1 - h_{12}(t)] v_i(t)$$



Chain rule



ANN in Speech Recognition: The classic period

- 1988 Morgan & Bourlard use NN for ASR
- 1989 Waibel et al. propose TDNN
- 1990: Robinson et al propose Recurrent NN

...

The second winter of ANN

- HMMs became dominant technology for ASR in 1990s because:
- It performed as well or better than ANN
- But it was a lot faster to train so HMMs could benefit from large training corpora whereas ANN could not

A renaissance of Neural Networks



- 2006: Hinton invents Deep Belief Networks (DBN):
 - Pre-train each layer from bottom up
 - Each pair of layers is an Restricted Boltzmann Machine (RBM), 1983
 - Jointly fine-tune all layers using back-propagation
- MNIST: handwritten digit recognition
- Great results due to good initialization



Restricted Boltzmann Machines I

Given \mathbf{v} and \mathbf{h} binary valued vectors

$$p(\mathbf{v}, \mathbf{h}) = \frac{e^{-E(\mathbf{v}, \mathbf{h})}}{Z} \quad E(\mathbf{v}, \mathbf{h}) = -\mathbf{b}^T \mathbf{v} - \mathbf{c}^T \mathbf{h} - \mathbf{v}^T \mathbf{W} \mathbf{h} \quad Z = \sum_{\mathbf{v}, \mathbf{h}} e^{-E(\mathbf{v}, \mathbf{h})}$$

$$p(\mathbf{h}|\mathbf{v}) = \frac{e^{-E(\mathbf{v}, \mathbf{h})}}{\sum_{\bar{\mathbf{h}}} e^{-E(\mathbf{v}, \bar{\mathbf{h}})}} = \frac{e^{\mathbf{b}^T \mathbf{v} + \mathbf{c}^T \bar{\mathbf{h}} + \mathbf{v}^T \mathbf{W} \bar{\mathbf{h}}}}{\sum_{\bar{\mathbf{h}}} e^{\mathbf{b}^T \mathbf{v} + \mathbf{c}^T \bar{\mathbf{h}} + \mathbf{v}^T \mathbf{W} \bar{\mathbf{h}}}} = \frac{e^{\mathbf{c}^T \bar{\mathbf{h}} + \mathbf{v}^T \mathbf{W} \bar{\mathbf{h}}}}{\sum_{\bar{\mathbf{h}}} e^{\mathbf{c}^T \bar{\mathbf{h}} + \mathbf{v}^T \mathbf{W} \bar{\mathbf{h}}}} = \prod_i p(h_i | \mathbf{v})$$

$$p(h_i = 1 | \mathbf{v}) = \sigma(c_i + \mathbf{v}^T \mathbf{W}_i)$$

Restricted Boltzmann Machines II

- Posterior of binary visible units

$$p(v_i = 1 | \mathbf{h}) = \sigma(b_i + \mathbf{w}_i^T \mathbf{h})$$

- When visible units are Gaussian

$$E(\mathbf{v}, \mathbf{h}) = \frac{1}{2} (\mathbf{v} - \mathbf{b})^T (\mathbf{v} - \mathbf{b}) - \mathbf{c}^T \mathbf{h} - \mathbf{v}^T \mathbf{W} \mathbf{h}$$

- still

$$p(h_i = 1 | \mathbf{v}) = \sigma(c_i + \mathbf{v}^T \mathbf{w}_i)$$

- Posteriors of visible units are Gaussian

$$p(\mathbf{v} | \mathbf{h}) = N(\mathbf{v}, \mathbf{b} + \mathbf{h} \mathbf{W}^T, I)$$

RBM estimation

- ML parameter estimation

$$\hat{\mathbf{c}}, \hat{\mathbf{b}}, \hat{\mathbf{W}} = \underset{\mathbf{c}, \mathbf{b}, \mathbf{W}}{\operatorname{argmax}} p(\mathbf{v} | \mathbf{c}, \mathbf{b}, \mathbf{W}) = \underset{\mathbf{c}, \mathbf{b}, \mathbf{W}}{\operatorname{argmax}} \sum_{\mathbf{h}} p(\mathbf{v}, \mathbf{h} | \mathbf{c}, \mathbf{b}, \mathbf{W})$$

- is highly non-linear ☹

Contrastive Divergence

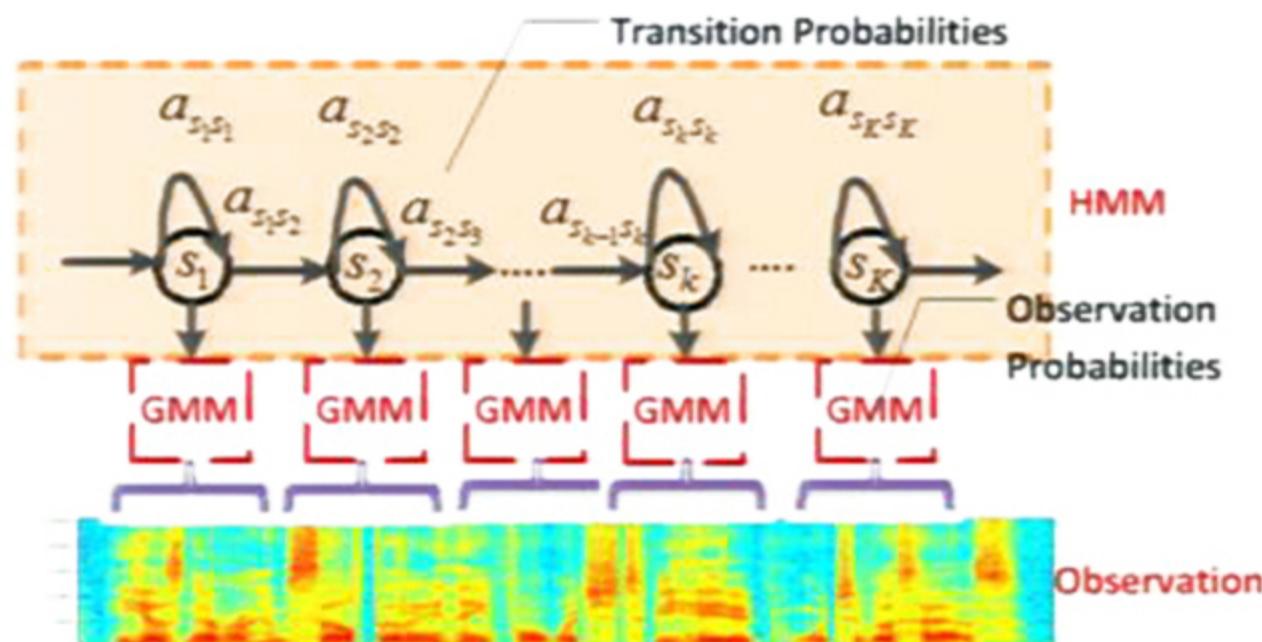
- $\Delta w_{ij} = \langle v_i h_j \rangle_{data} - \langle v_i h_j \rangle_{model}$
- Approximate $\langle v_i h_j \rangle_{model}$
 - i. Initialize v_0 at data
 - ii. Sample $\mathbf{h}_0 \sim p(\mathbf{h}|\mathbf{v}_0)$
 - iii. Sample $\mathbf{v}_1 \sim p(\mathbf{v}|\mathbf{h}_0)$
 - iv. Sample $\mathbf{h}_1 \sim p(\mathbf{h}|\mathbf{v}_1)$
 - v. Call $(\mathbf{v}_1, \mathbf{h}_1)$ a sample from the model.
- $(\mathbf{v}_\infty, \mathbf{h}_\infty)$ is a true sample from the model. $(\mathbf{v}_1, \mathbf{h}_1)$ is a very rough estimate but works

Neural Network training

- RBM pre-training (contrastive divergence)
- Back-propagation

State-of-the-art: GMM-HMM

- Generatively model frames of acoustic data with two stochastic processes:
 - A hidden Markov process to model state transition
 - A Gaussian mixture model to generate observations
- Trained with maximum likelihood (ML) criterion using EM followed by discriminative training (e.g. MPE)

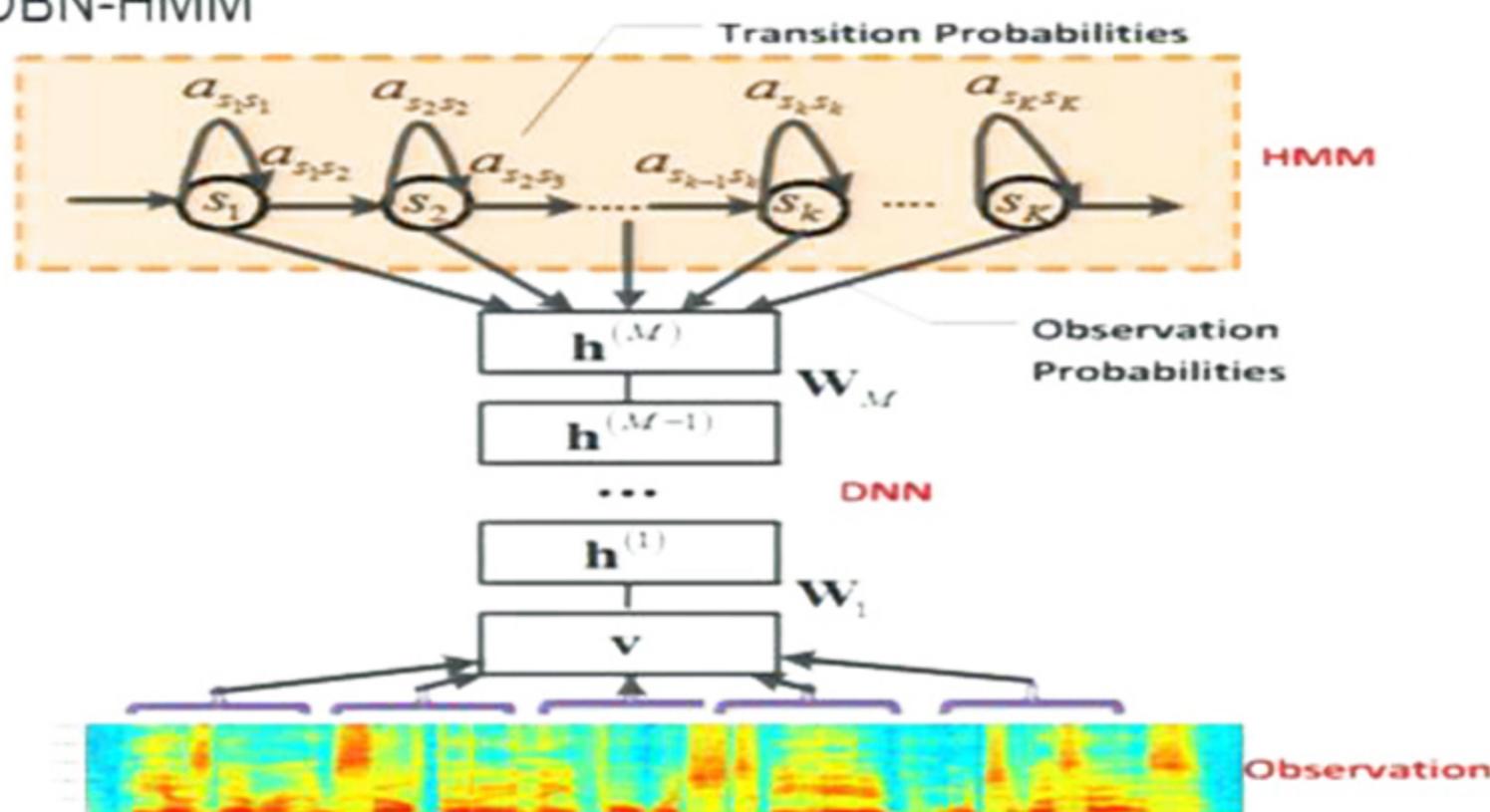


Context Dependent DNN-HMM

Dong Yu & Li Deng



- Extend from phoneme recognition (Mohamed et al. 2009) to LVCSR
- Extend from using CI phone state to using senones as DBN output
- Introduced priors, transition prob tuning, and DBN labels in the DBN-HMM



Context Dependent DNN-HMM



- Convert state posterior to state likelihood [Renals et al., 1994]

$$p_{o|s}(o|s) = \frac{p_{s|o}(s|o)}{p_s(s)} p_o(o)$$

- $p_o(o)$ is constant with input
 - o = feature vector augmented with neighbors (5+5) [Renals et al., 1994]
 - new: classes s are *conventional model's senones* directly [Yu et al. 2010]
 - in our system: ~9000
 - long-standing assumption: too many to be accurately modeled by MLP
 - the key ingredient for large WER reduction
- hence name: **CD-DNN-HMMs**

Switchboard Experiments

Frank Seide, Dong Yu, Gang Li



- training:
 - SWBD-I corpus (309h)
 - PLP with derivatives, windowed MVN, HLDA → 39 dim
 - usual left-to-right HMMs, 9304 senones
 - GMM baseline: 40 Gaussians/state; BMMI discriminative training
- recognition:
 - speaker-independent single-pass
 - dev set: Hub5'00-SWB NOTE: speaker overlap!
 - eval sets: RT03S & internal STT corpora
 - LM and dict from Fisher transcripts, PP=84
- for comparison: our “best-ever” multi-pass baseline
 - trained on 2000 hours (SWBD-I + Fisher)
 - VTLN, GD, multi-pass MLLR, ROVER

Experimental Results



- 300h Switchboard phone conversations (cf. our best: 1700h)

acoustic model & training	recognition mode	RT03S		Hub5'00
		FSH	SW	SWB
GMM 40-mix. ML. SWB 309h	single-pass SI	30.2		26.5
GMM 40-mix. BMMI. SWB 309h	single-pass SI	27.4		23.6
CD-DNN 7 layers x 2048. SWB 309h. this paper (rel. change GMM BMMI — CD-DNN)	single-pass SI			
GMM 72-mix. BMMI. Fisher 2000h	multi-pass adaptive	18.6		17.1

Experimental Results



- 300h Switchboard phone conversations (cf. our best: 1700h)

acoustic model & training	recognition mode	RT03S		Hub5'00
		FSH	SW	SWB
GMM 40-mix. ML. SWB 309h	single-pass SI	30.2		26.5
GMM 40-mix. BMMI. SWB 309h	single-pass SI	27.4		23.6
CD-DNN 7 layers x 2048. SWB 309h. this paper (rel. change GMM BMMI — CD-DNN)	single-pass SI	18.5 (-33%)		16.1 (-32%)
GMM 72-mix. BMMI. Fisher 2000h	multi-pass adaptive	18.6		17.1

Experimental Results



- 300h Switchboard phone conversations (cf. our best: 1700h)

acoustic model & training	recognition mode	RT03S		Hub5'00
		FSH	SW	SWB
GMM 40-mix. ML. SWB 309h	single-pass SI	30.2	40.9	26.5
GMM 40-mix. BMMI. SWB 309h	single-pass SI	27.4	37.6	23.6
CD-DNN 7 layers x 2048. SWB 309h. this paper (rel. change GMM BMMI — CD-DNN)	single-pass SI	18.5 (-53%)	27.5 (-27%)	16.1 (-32%)
GMM 72-mix. BMMI. Fisher 2000h	multi-pass adaptive	18.6	25.2	17.1

acoustic model & training	recognition mode	voicemails		tele-
		MS	LDC	conf
GMM 40-mix. ML. SWB 309h	single-pass SI	45.0	33.5	35.2
GMM 40-mix. BMMI. SWB 309h	single-pass SI	42.4	30.8	33.9
CD-DNN 7 layers x 2048. SWB 309h. this paper (rel. change GMM BMMI — CD-DNN)	single-pass SI	32.9 (-22%)	22.9 (-26%)	24.4 (-28%)

Summary

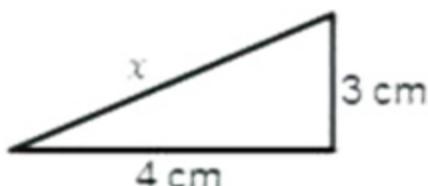
- CD-DNN-HMM scales to “benchmark” data
 - 9000 senones, 300h, STT task
 - unusual 33% relative error reduction
(historically, not many technologies achieved this)
- Key factors
 - **Increase in computing power** allows more experiments:
 - direct modeling of tied triphone states [Yu et al., 2010]
 - effective use of neighbor frames (-14%) [Renals et al., 1994]
 - modeling ability of deep networks (-24%) [Yu et al., 2010]
- Training still a problem:
 - Still slow: need GPUs
 - Can get stuck in local optimum

Natural or Artificial?

Natural or Artificial?

- Artificial neural networks better than natural?

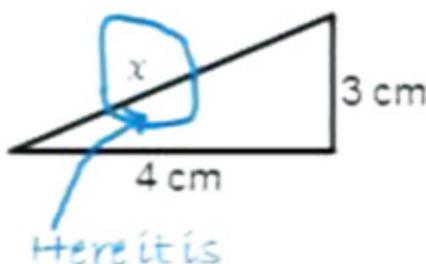
Find x



Natural or Artificial?

- Artificial neural networks better than natural?

Find x



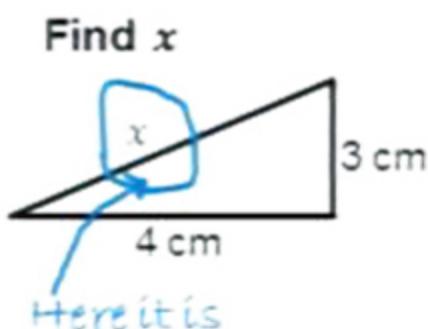
What is the language spoken in Latin America?

Latin



Natural or Artificial?

- Artificial neural networks better than natural?



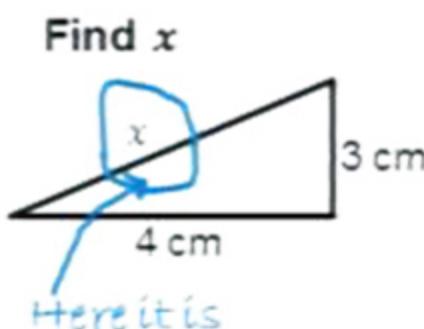
What is the language spoken in Latin America?
Latin



- In human intelligence tasks, ANN might do better than natural ones ... one day

Natural or Artificial?

- Artificial neural networks better than natural?



What is the language spoken in Latin America?
Latin



- In human intelligence tasks, ANN might do better than natural ones ... one day
- But for now, ANN have a lot to learn from nature
 - Randomness, an accident or Darwin at his best?
 - Local learning instead of backprop?

Microsoft Research

FacultySummit 2012

Riviera Maya, Mexico | May 23–25 | In partnership with CONACYT