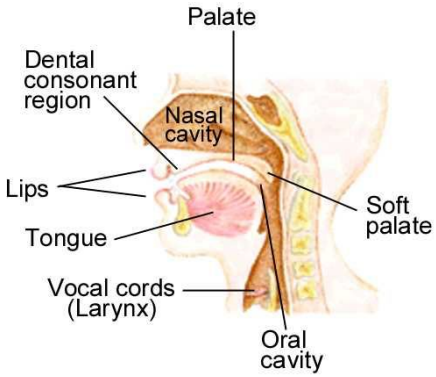
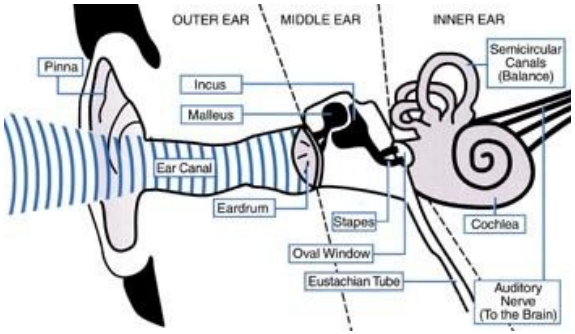
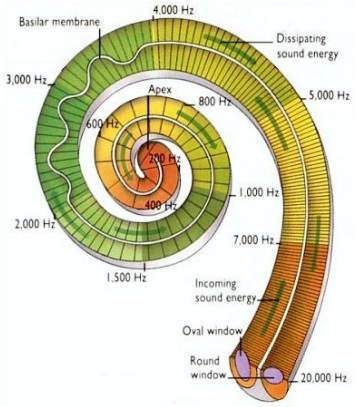
Speech Recognition with Deep Neural Networks

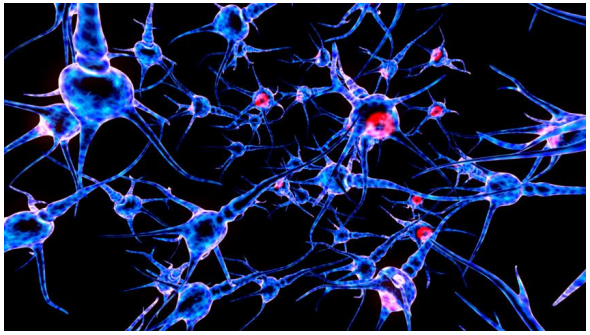
José A. R. Fonollosa
Universitat Politècnica de Catalunya
Barcelona, January 26, 2017

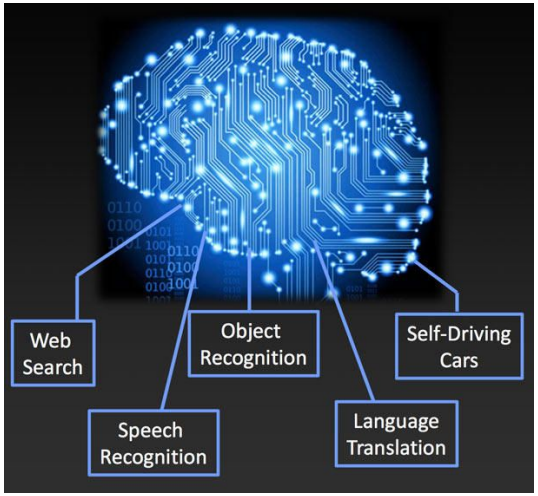
From speech processing to deep learning







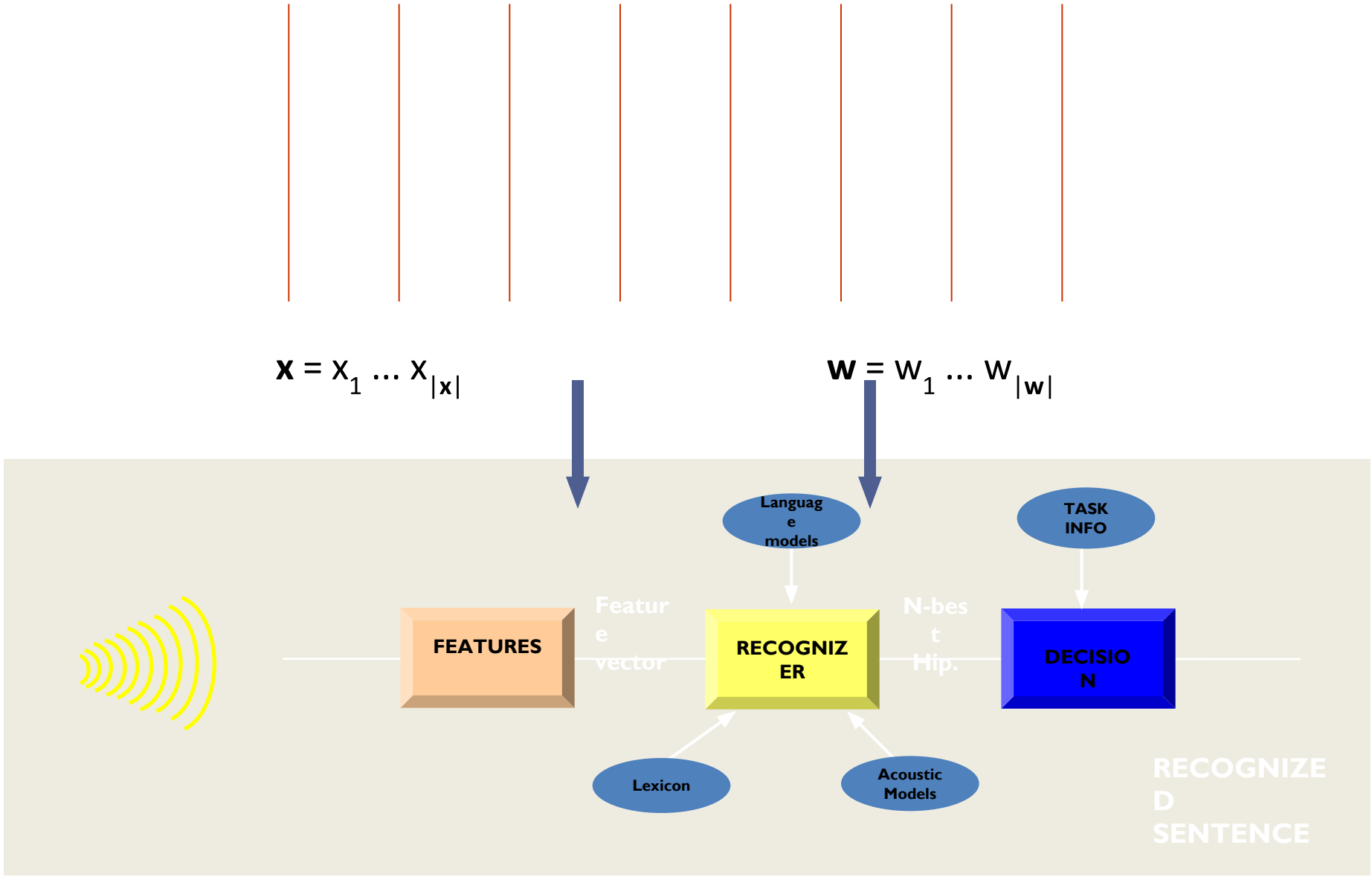




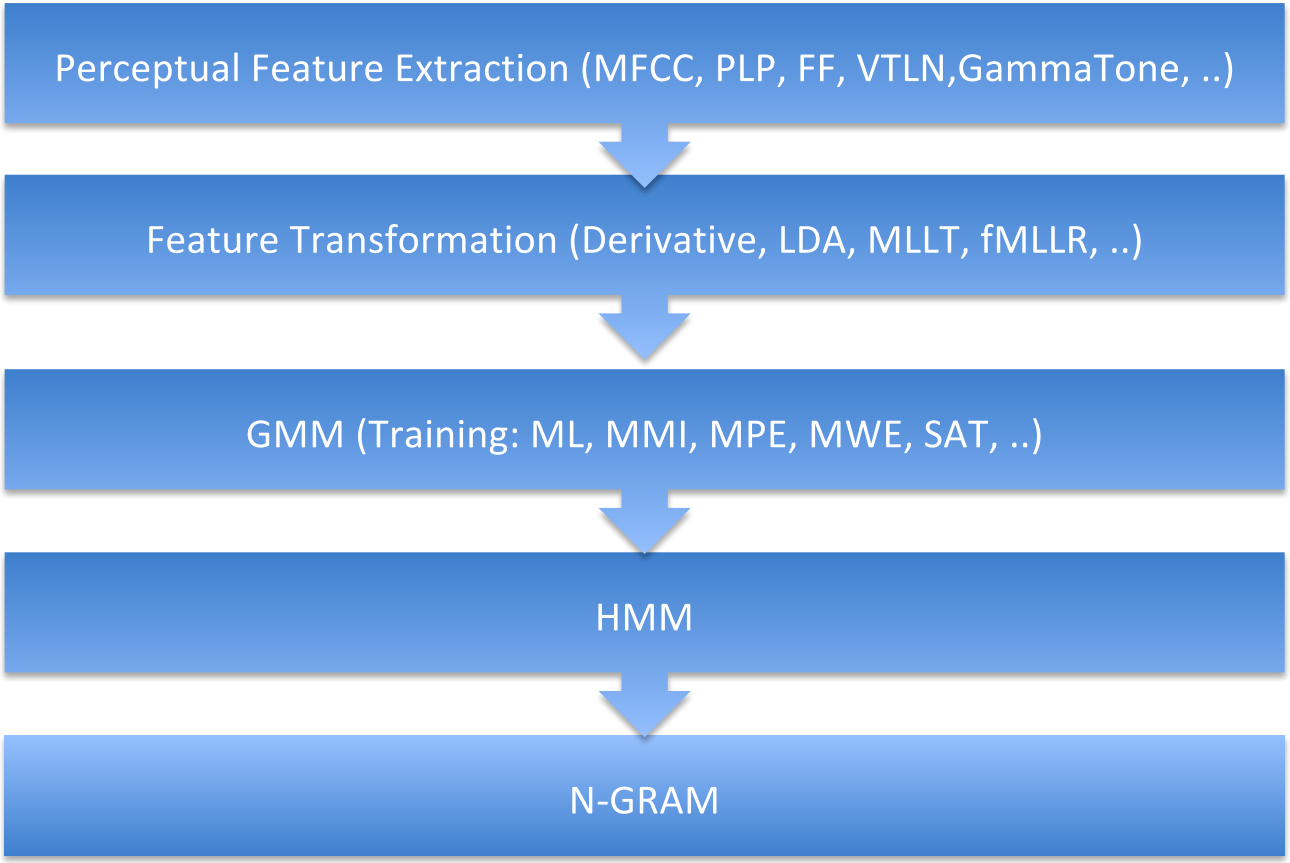
Towards end-to-end RNN Speech Recognition?

- Architectures
  - GMM-HMM: 30 years of feature engineering
  - DNN-GMM-HMM: Trained features
  - DNN-HMM: TDNN, LSTM, RNN, MS
  - DNN for language modeling (RNN)
  - End-to-end DNN?
- Examples
  - Alex Graves (Google)
  - Deep Speech (Baidu)

Recognition system



GMM-HMM



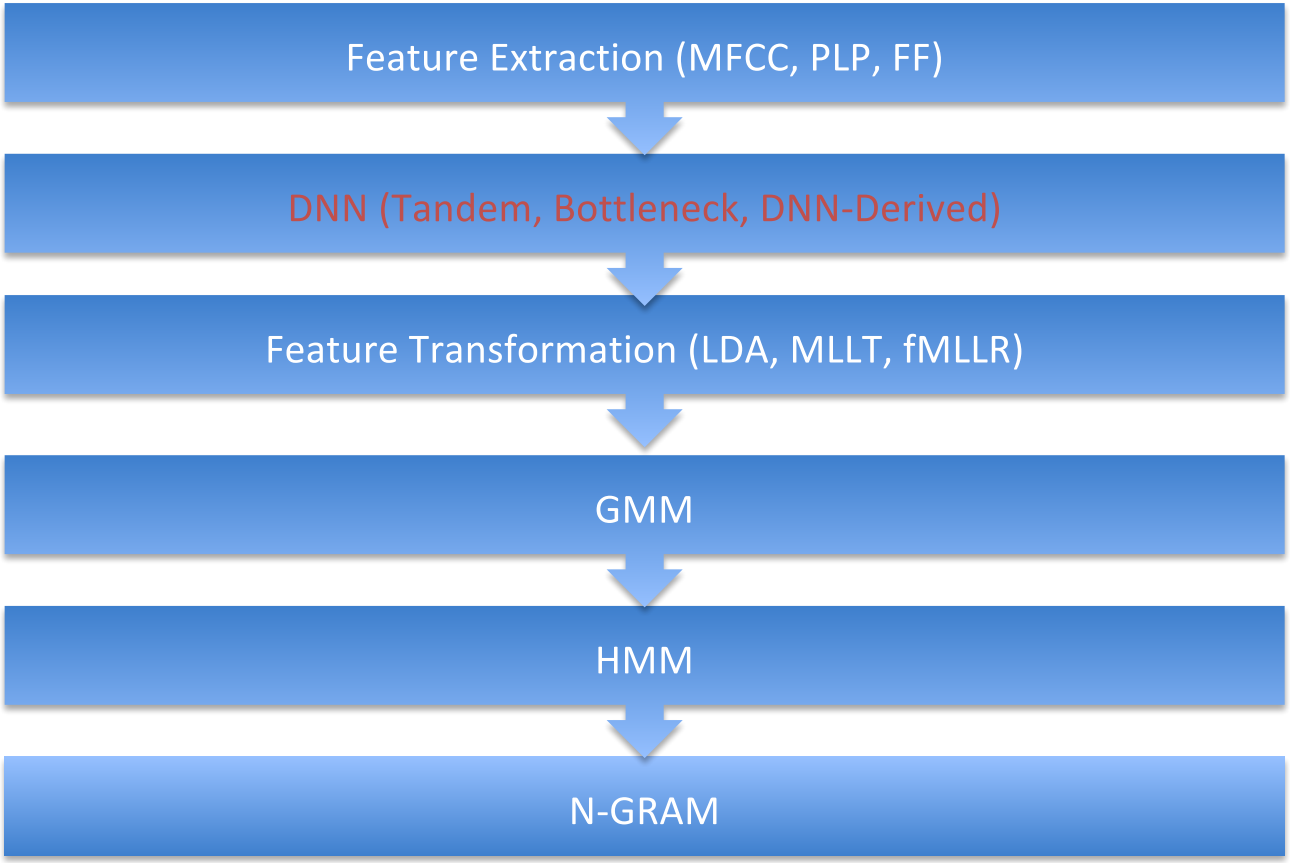
Acoustic Model

Phonetic inventory

Pronunciation Lexicon

Language Model

DNN-GMM-HMM



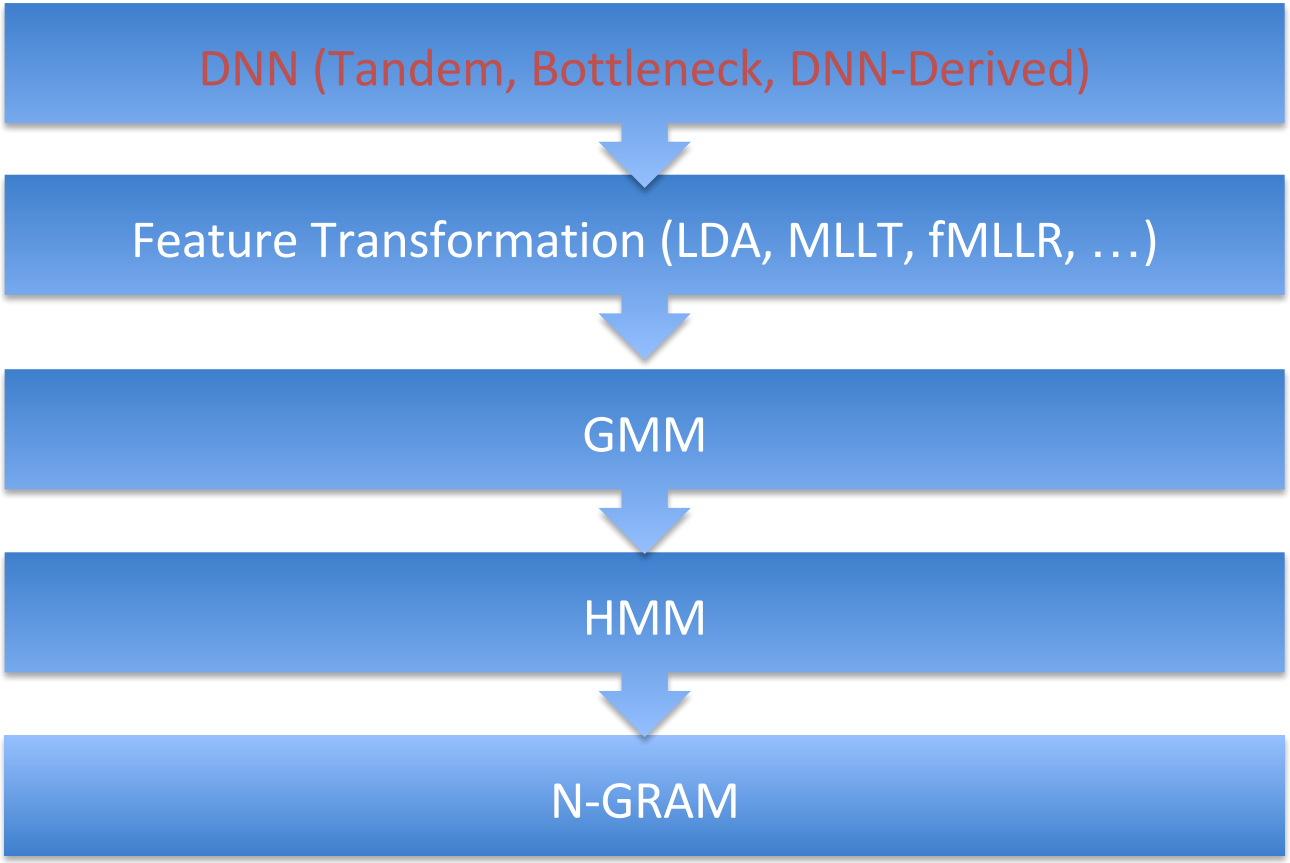
Acoustic Model

Phonetic inventory

Pronunciation Lexicon

Language Model

DNN-GMM-HMM



Acoustic Model

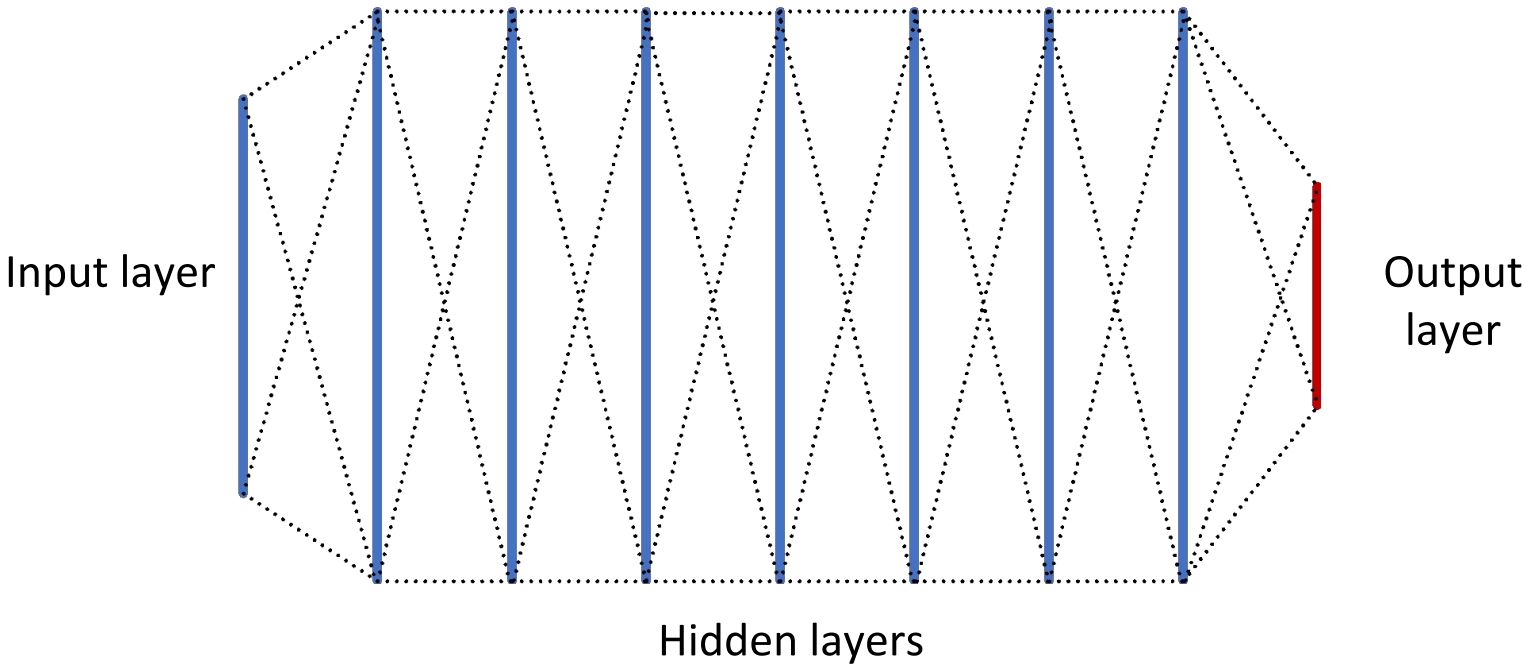
Phonetic inventory

Pronunciation Lexicon

Language Model

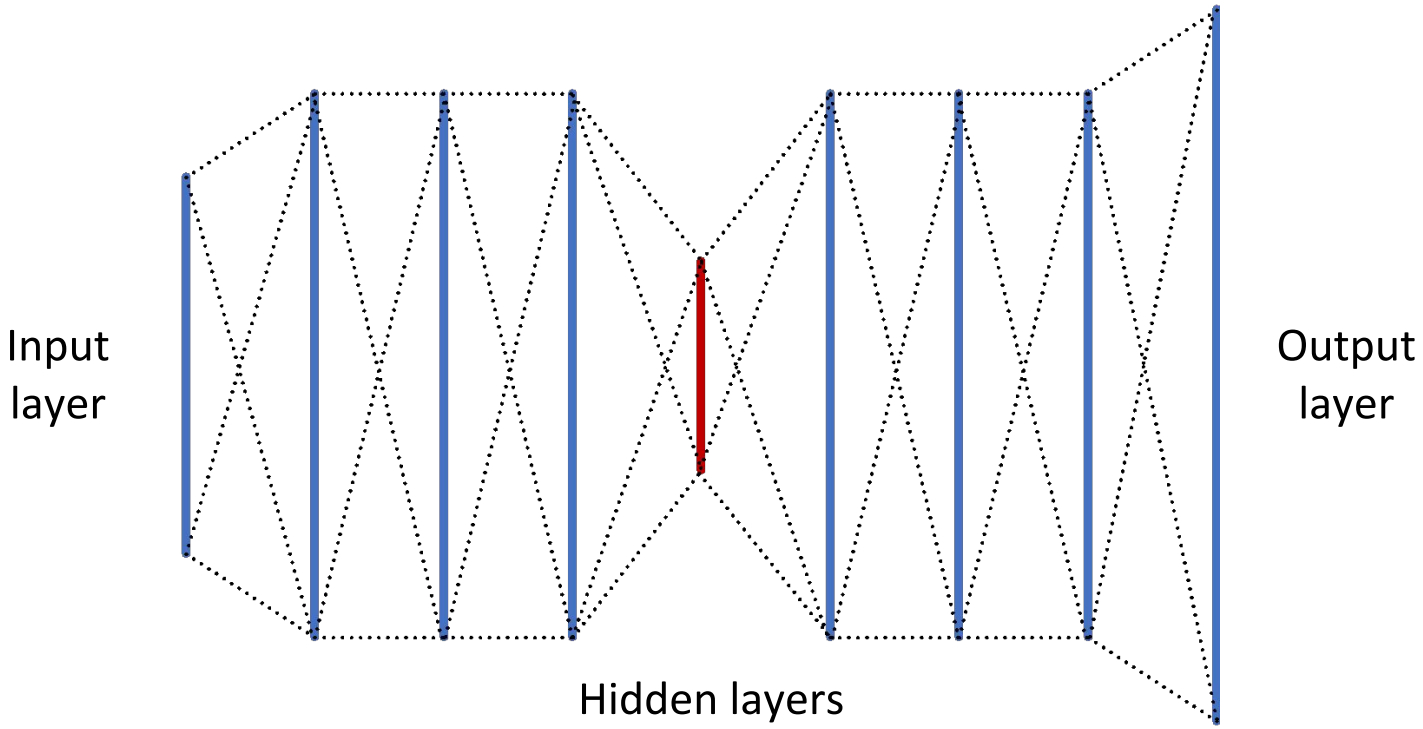
Tandem

- **MLP outputs as input to GMM**



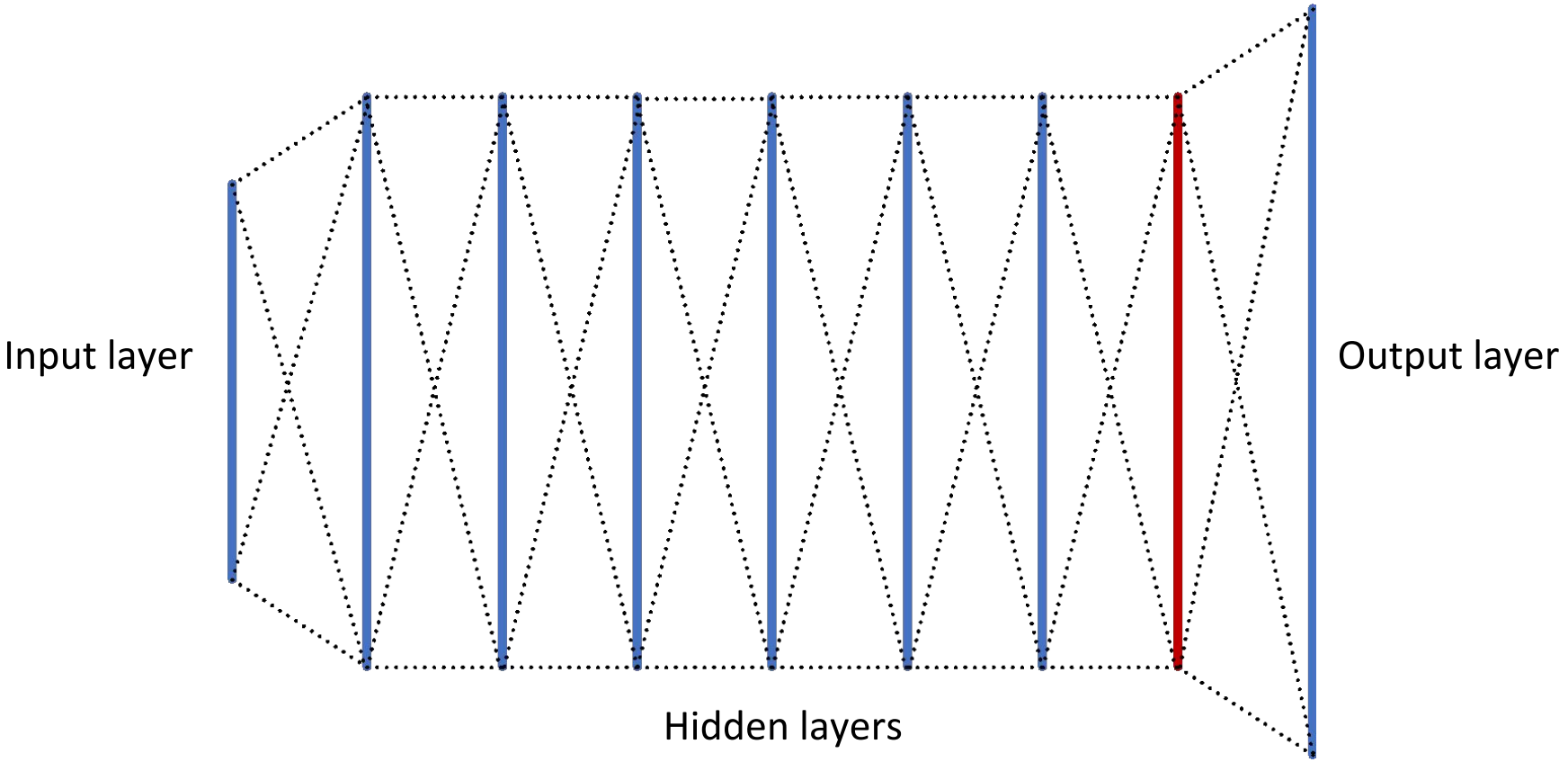
Bottleneck Features

- **Use one narrow hidden layer. Supervised or unsupervised training (autoencoder)**



DNN-Derived Features

- Zhi-Jie Yan, Qiang Huo, Jian Xu: A scalable approach to using DNN-derived features in GMM-HMM based acoustic modeling for LVCSR. INTERSPEECH 2013: 104-108



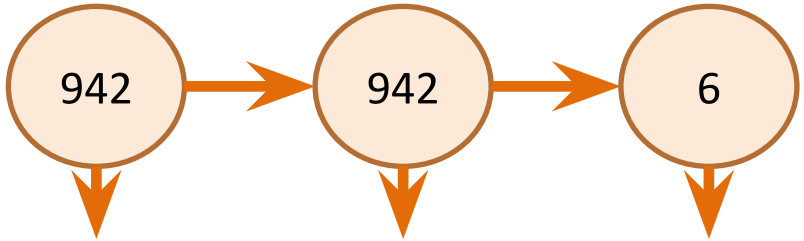
Acoustic Modeling with GMMs

Transcription: Samson

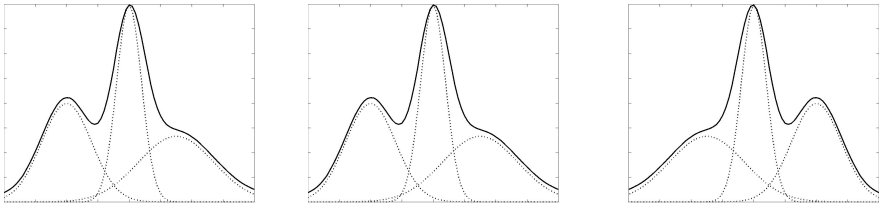
Pronunciation: S – AE – M – S –AH – N

Sub-phones : 942 – 6 – 37 – 8006 – 4422 …

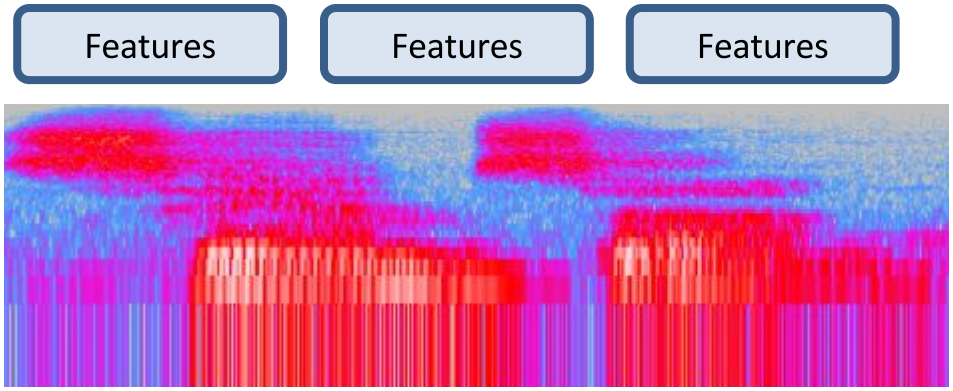
Hidden Markov Model (HMM):



Acoustic Model:

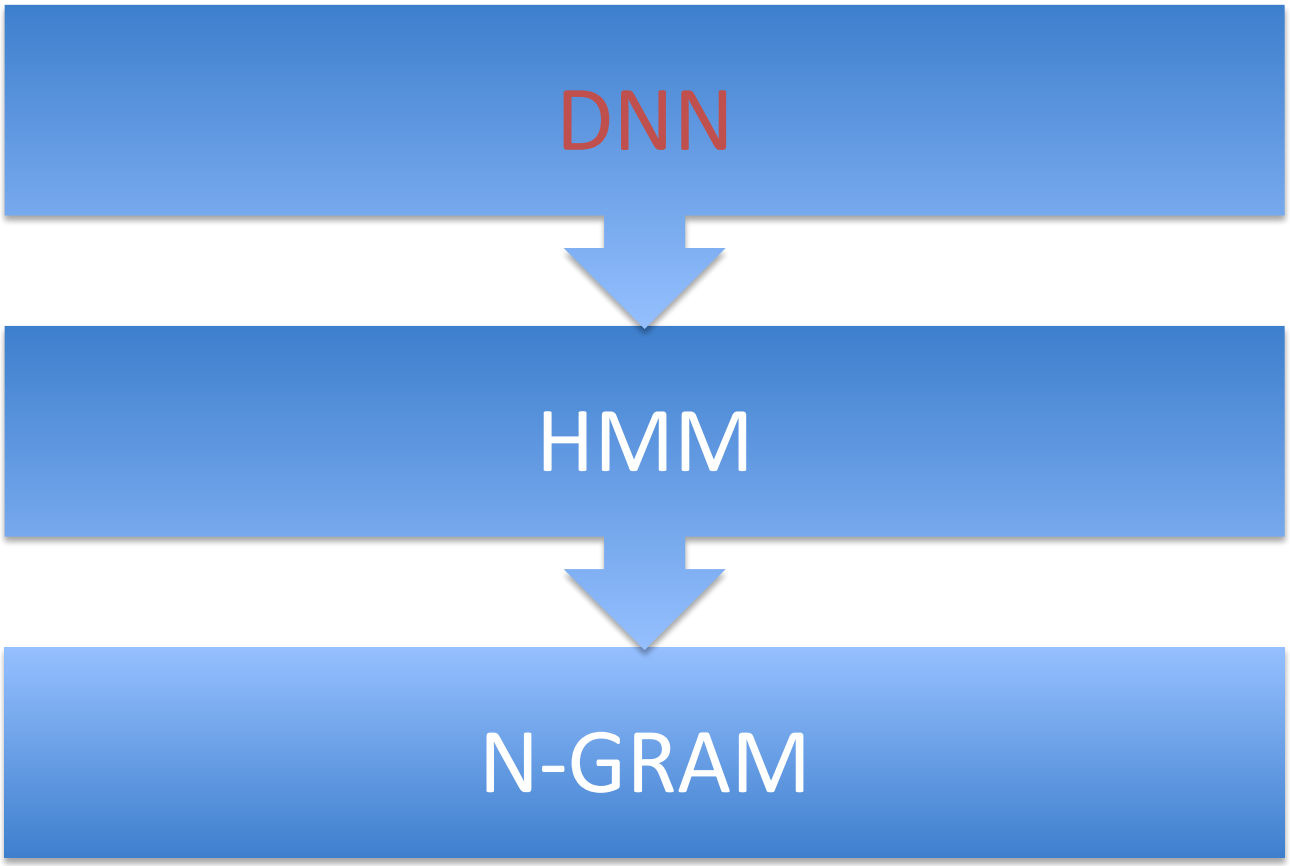


Audio Input:



GMM models:
P(x|s)
x: input features
s: HMM state

DNN-HMM



Acoustic Model

Phonetic inventory

Pronunciation Lexicon

Language Model

DNN Hybrid Acoustic Models

Transcription:

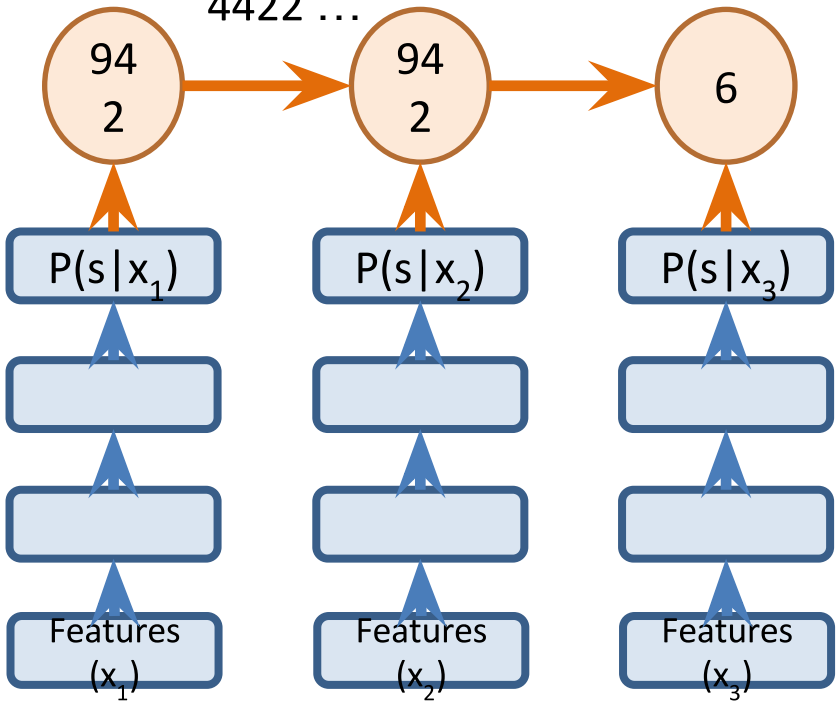
Samson

Pronunciation:

S – AE – M – S –AH – N

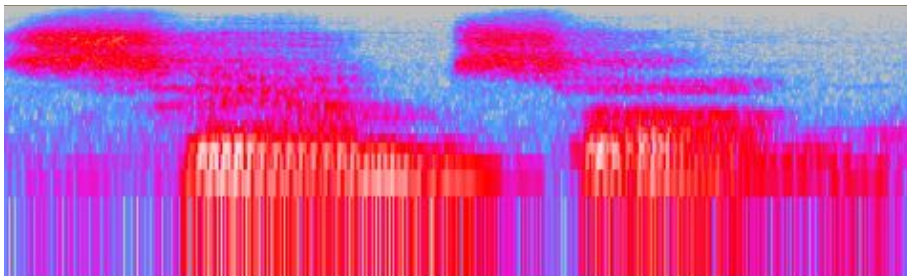
Sub-phones:

942 – 6 – 37 – 8006 – 4422 …



Hidden Markov Model (HMM):

Acoustic Model:

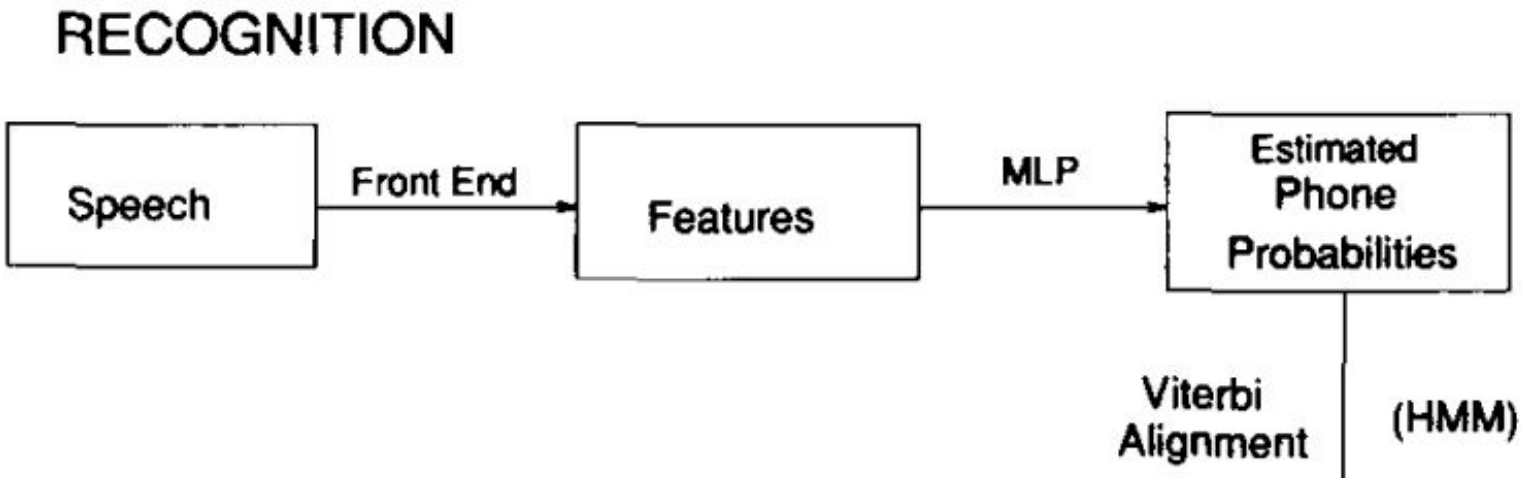


Use a DNN to approximate: P(s|x)

Apply Bayes' Rule: P(x|s) = P(s|x) * P(x) / P(s)

DNN * Constant / State prior

Not Really a New Idea



Renals, Morgan, Bourland, Cohen, & Franco. 1994.

Hybrid MLPs on Resource Management

full context-dependent Decipher system, and the MIX system is a simple interpolation between the CD-HMM and the CI-MLP.)

	% error		
Test Set	CI-MLP	CD-HMM	MIX
Feb 91	5.8	3.8	3.2
Sep 92a	10.9	10.1	7.7
Sep 92b	9.5	7.0	5.7

TABLE II
RESULTS USING THE THREE TEST SETS USING NO GRAMMAR (PERLPEXITY 991)

	% error

Renals, Morgan, Bourland, Cohen, & Franco. 1994.

Hybrid Systems now Dominate ASR

[TABLE 3] A COMPARISON OF THE PERCENTAGE WERs USING DNN-HMMs AND GMM-HMMs ON FIVE DIFFERENT LARGE VOCABULARY TASKS.

TASK	HOURS OF TRAINING DATA	DNN-HMM	GMM-HMM WITH SAME DATA	GMM-HMM WITH MORE DATA
SWITCHBOARD (TEST SET 1)	309	18.5	27.4	18.6 (2,000 H)
SWITCHBOARD (TEST SET 2)	309	16.1	23.6	17.1 (2,000 H)
ENGLISH BROADCAST NEWS	50	17.5	18.8	
BING VOICE SEARCH (SENTENCE ERROR RATES)	24	30.4	36.2	
GOOGLE VOICE INPUT	5,870	12.3		16.0 (>> 5,870 H)
YOUTUBE	1,400	47.6	52.3	

Hinton et al. 2012.

What's Different in Modern DNNs?

- Fast computers = run many experiments
- Many more parameters
- Deeper nets improve on shallow nets
- Architecture choices (easiest is replacing sigmoid)
- Pre-training *does not matter*. Initially we thought this was the new trick that made things work

Adding More Parameters 15 Years Ago

Size matters: An empirical study of neural network training for LVCSR. Ellis & Morgan. ICASSP. 1999.

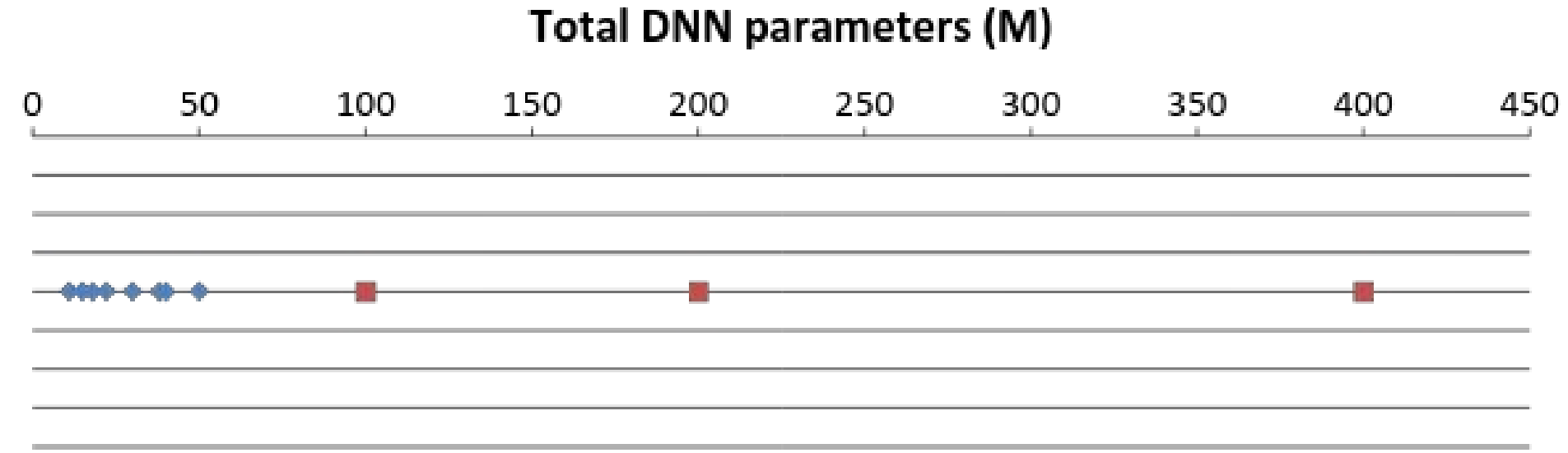
Hybrid NN. 1 hidden layer. 54 HMM states.

74hr broadcast news task

"…improvements are almost always obtained by increasing either or both of the amount of training data or the number of network parameters … We are now planning to train an 8000 hidden unit net on 150 hours of data … this training will require over three weeks of computation."

Adding More Parameters Now

- Comparing total number of parameters (in millions) of previous work versus our new experiments



Maas, Hannun, Qi, Lengerich, Ng, & Jurafsky.

Sample of Results

- 2,000 hours of conversational telephone speech
- Kaldi baseline recognizer (GMM)
- DNNs take 1 -3 weeks to train

Acoustic Model	Training hours	Dev CrossEnt	Dev Acc(%)	FSH WER
GMM	2,000	N/A	N/A	32.3
DNN 36M	300	2.23	49.9	24.2
DNN 200M	300	2.34	49.8	23.7
DNN 36M	2,000	1.99	53.1	23.3
DNN 200M	2,000	1.91	55.1	21.9

Maas, Hannun, Qi, Lengerich, Ng, & Jurafsky. In submission.

Depth Matters (Somewhat)

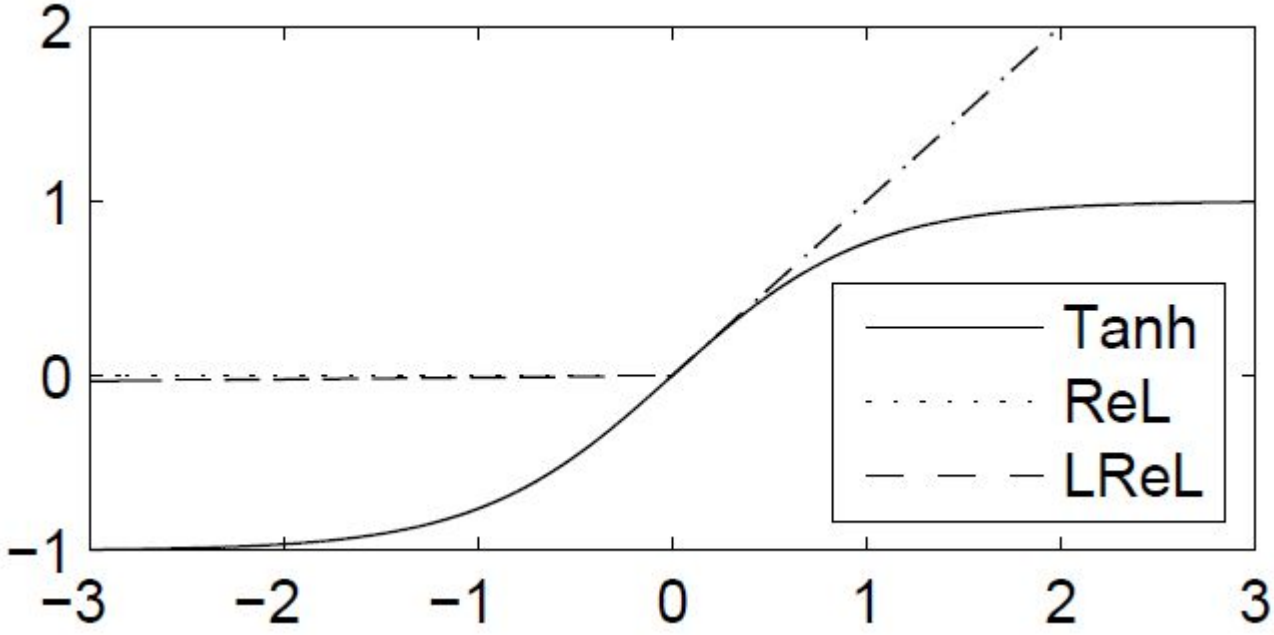
Table 1: Effect of CD-DNN-HMM network depth on WER (%) on Hub5'00-SWB using the 309-hour Switchboard training set. DBN pretraining is applied.

$L \times N$	WER	$1 \times N$	WER
1×2k	24.2	–	–
2×2k	20.4	–	–
3×2k	18.4	–	–
4×2k	17.8	–	–
5×2k	17.2	1×3772	22.5
7×2k	17.1	1×4634	22.6
9×2k	17.0	–	–
5×3k	17.0	–	–
–	–	1×16k	22.1

Warning! Depth can also act as a regularizer because it makes optimization more difficult. This is why you will sometimes see very deep networks perform well on TIMIT or other small tasks.

Yu, Seltzer, Li, Huang, Seide. 2013.

Architecture Choices:



Rectified Linear (ReL)

$$h^{(i)} = \max(w^{(i)T}x, 0) = \begin{cases} w^{(i)T}x & w^{(i)T}x > 0 \\ 0 & \text{else} \end{cases}$$

[Glorot et al, AISTATS 2011]

Leaky Rectified Linear (LReL)

$$h^{(i)} = \begin{cases} w^{(i)T}x & w^{(i)T}x > 0 \\ 0.01w^{(i)T}x & \text{else} \end{cases}$$

Rectifier DNNs on

Model	Dev CrossEnt	Dev Acc(%)
GMM Baseline	N/A	N/A
2 Layer Tanh	2.09	48.0
2 Layer ReLU	1.91	51.7
2 Layer LRelU	1.90	51.8
3 Layer Tanh	2.02	49.8
3 Layer RelU	1.83	53.3
3 Layer LRelU	1.83	53.4
4 Layer Tanh	1.98	49.8
4 Layer RelU	1.79	53.9
4 Layer LRelU	1.78	53.9

Maas, Hannun, & Ng,. 2013.

Rectifier DNNs on

Model	Dev CrossEnt	Dev Acc(%)	Switchboard WER	Callhome WER	Eval 2000 WER
GMM Baseline	N/A	N/A	25.1	40.6	32.6
2 Layer Tanh	2.09	48.0	21.0	34.3	27.7
2 Layer ReLU	1.91	51.7	19.1	32.3	25.7
2 Layer LRelU	1.90	51.8	19.1	32.1	25.6
3 Layer Tanh	2.02	49.8	20.0	32.7	26.4
3 Layer RelU	1.83	53.3	18.1	30.6	24.4
3 Layer LRelU	1.83	53.4	17.8	30.7	24.3
4 Layer Tanh	1.98	49.8	19.5	32.3	25.9
4 Layer RelU	1.79	53.9	17.3	29.9	23.6
4 Layer LRelU	1.78	53.9	17.3	29.9	23.7
9 Layer Sigmoid CE [MSR]	--	--	17.0	--	--
7 Layer Sigmoid MMI [IBM]	--	--	13.7	--	--

Maas, Hannun, & Ng,. 2013.

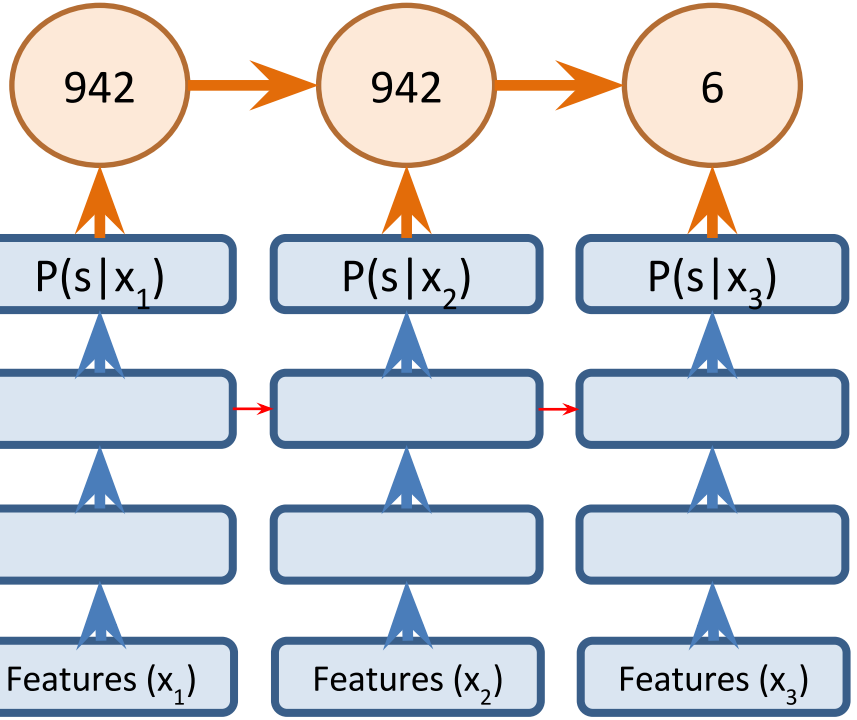
Recurrent DNN

Transcription: Samson

Pronunciation: S – AE – M – S –AH – N

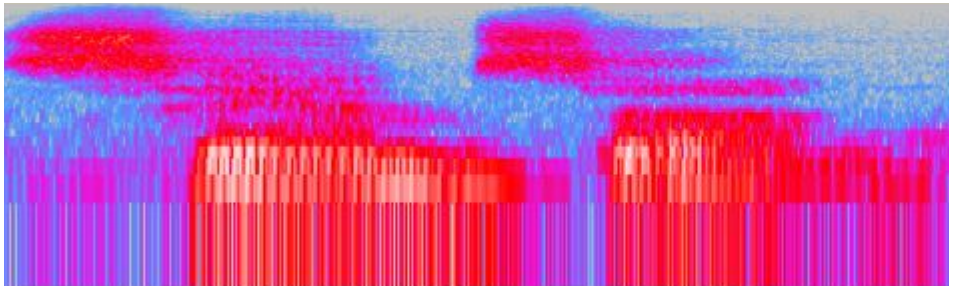
Sub-phones : 942 – 6 – 37 – 8006 – 4422 …

Hidden Markov Model (HMM):



Acoustic Model:

Audio Input:



Other Current Work

- Changing the DNN loss function. Typically using discriminative training ideas already used in ASR
- Reducing dependence on high quality alignments. In the limit you could train a hybrid system from flat start / no alignments
- Multi-lingual acoustic modeling
- Low resource acoustic modeling
- End-to-end neural speech recognition