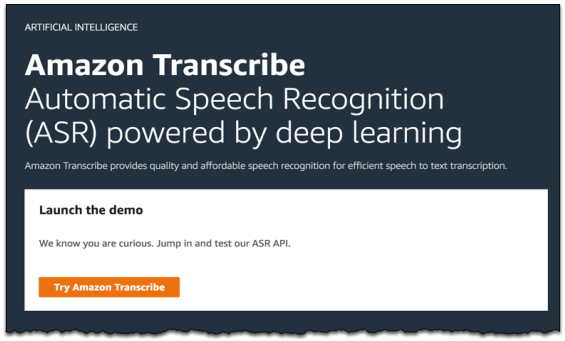
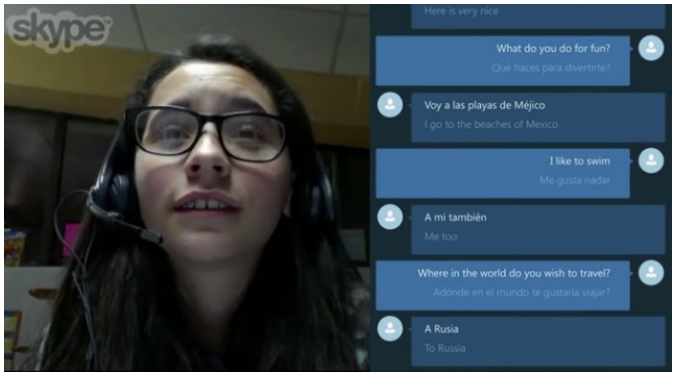
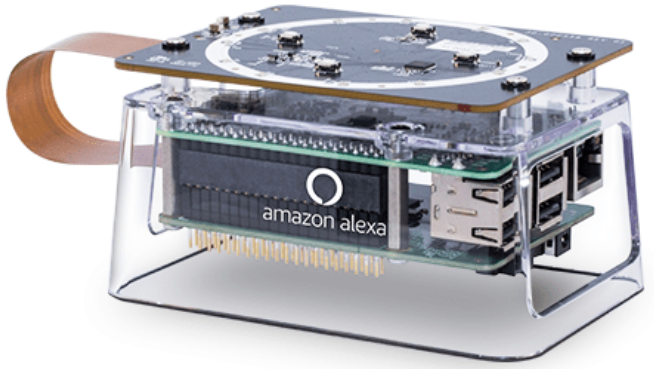
Speech Recognition with Deep Neural Networks

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

Speech Recognition

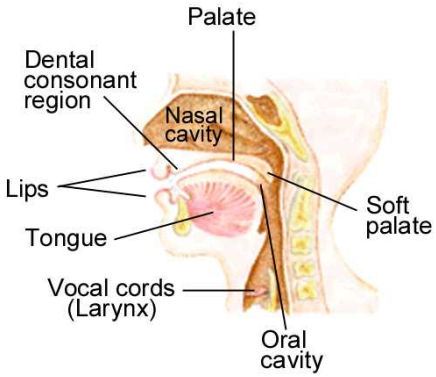


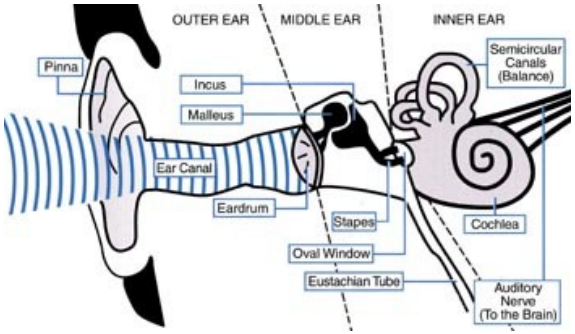


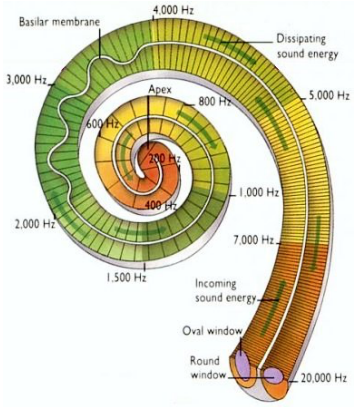


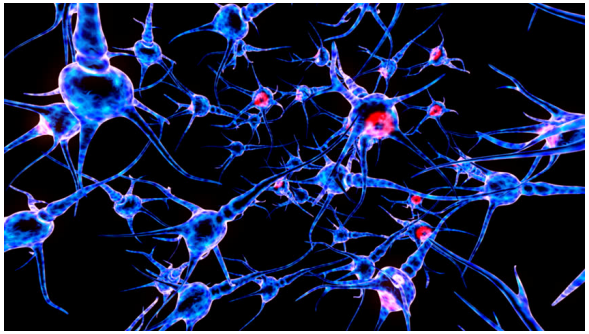
Apple, Microsoft, Google, Baidu, Amazon in a speech race
More than 10 percent of search is now made by voice (Google, Nov. 2017)

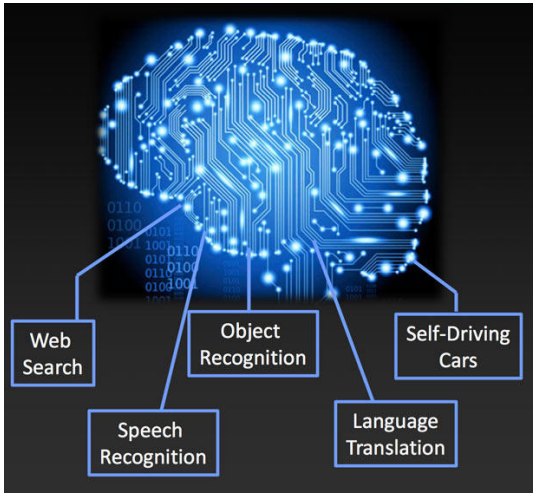
From speech processing to deep learning







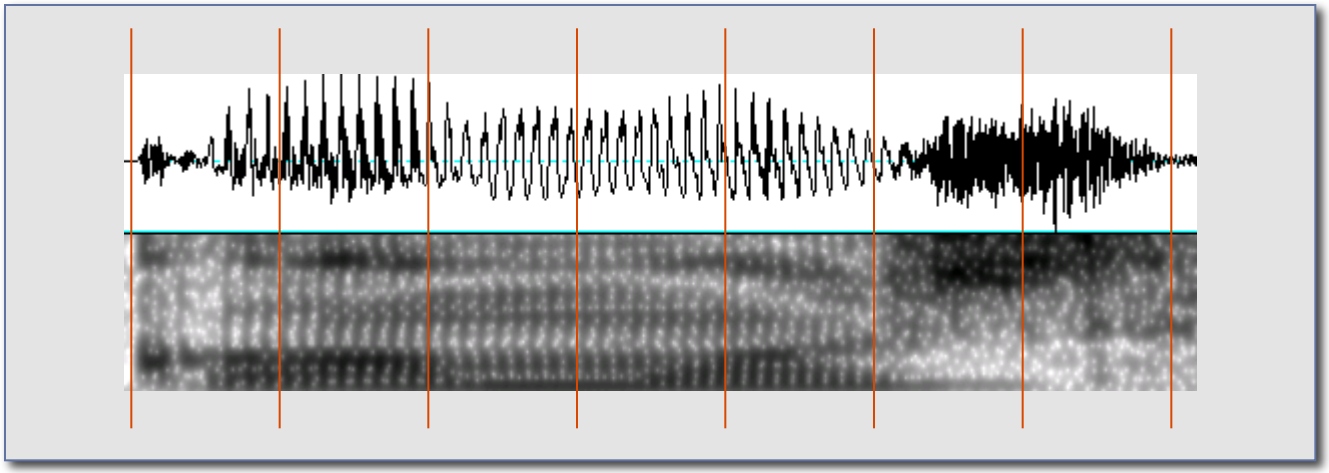




Master MET course (Spring): **Speech Technologies (ASR, TTS, Speech Translation, NLP)**

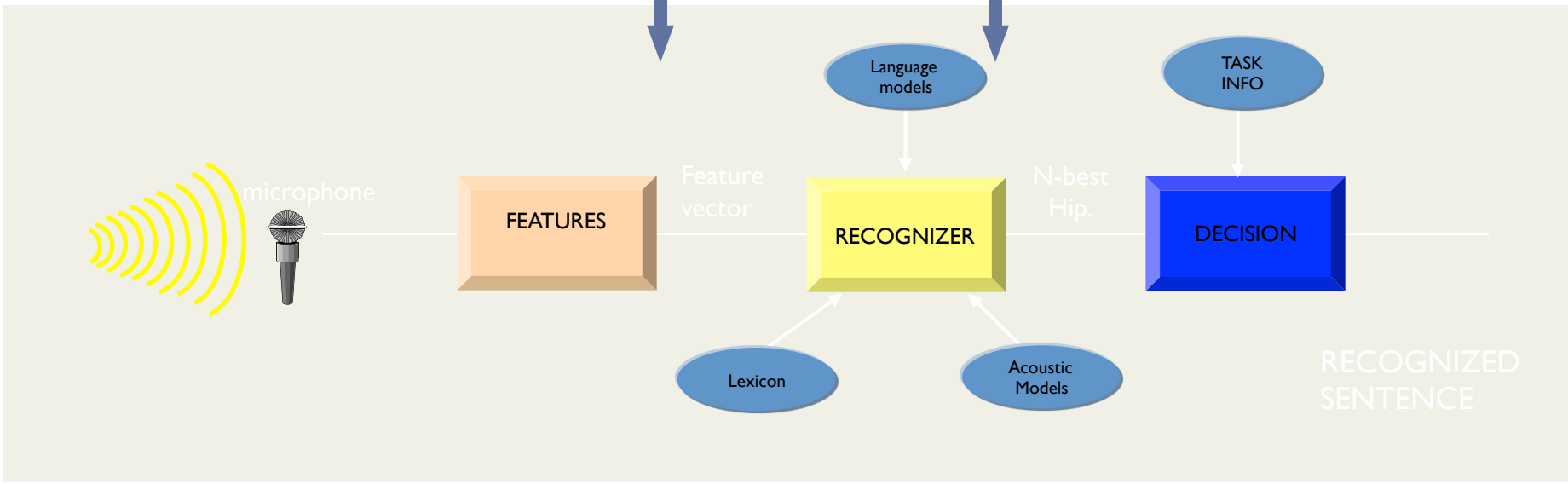
Recognition system

Input:
100 fps

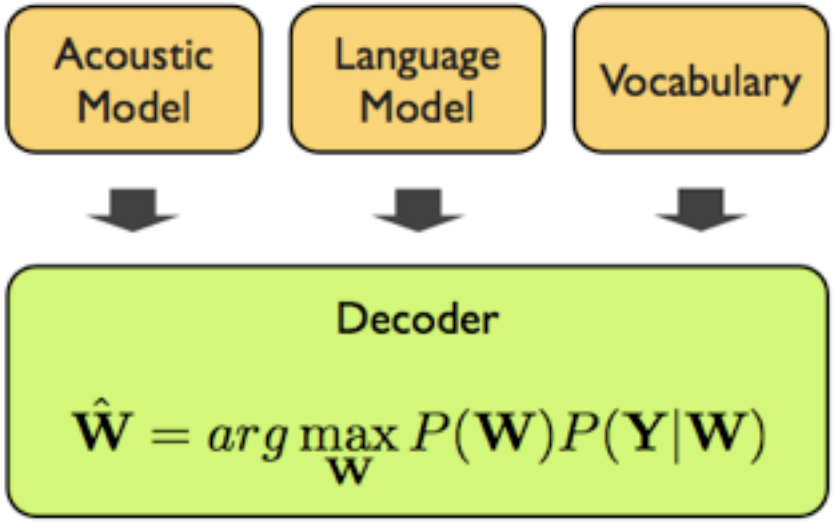


Output:
110-160
wpm
(2-3 wps)

From speech $\mathbf{x} = x_1 \dots x_{|\mathbf{x}|}$ $\mathbf{w} = w_1 \dots w_{|\mathbf{w}|}$ to words



Classic system: 3 models

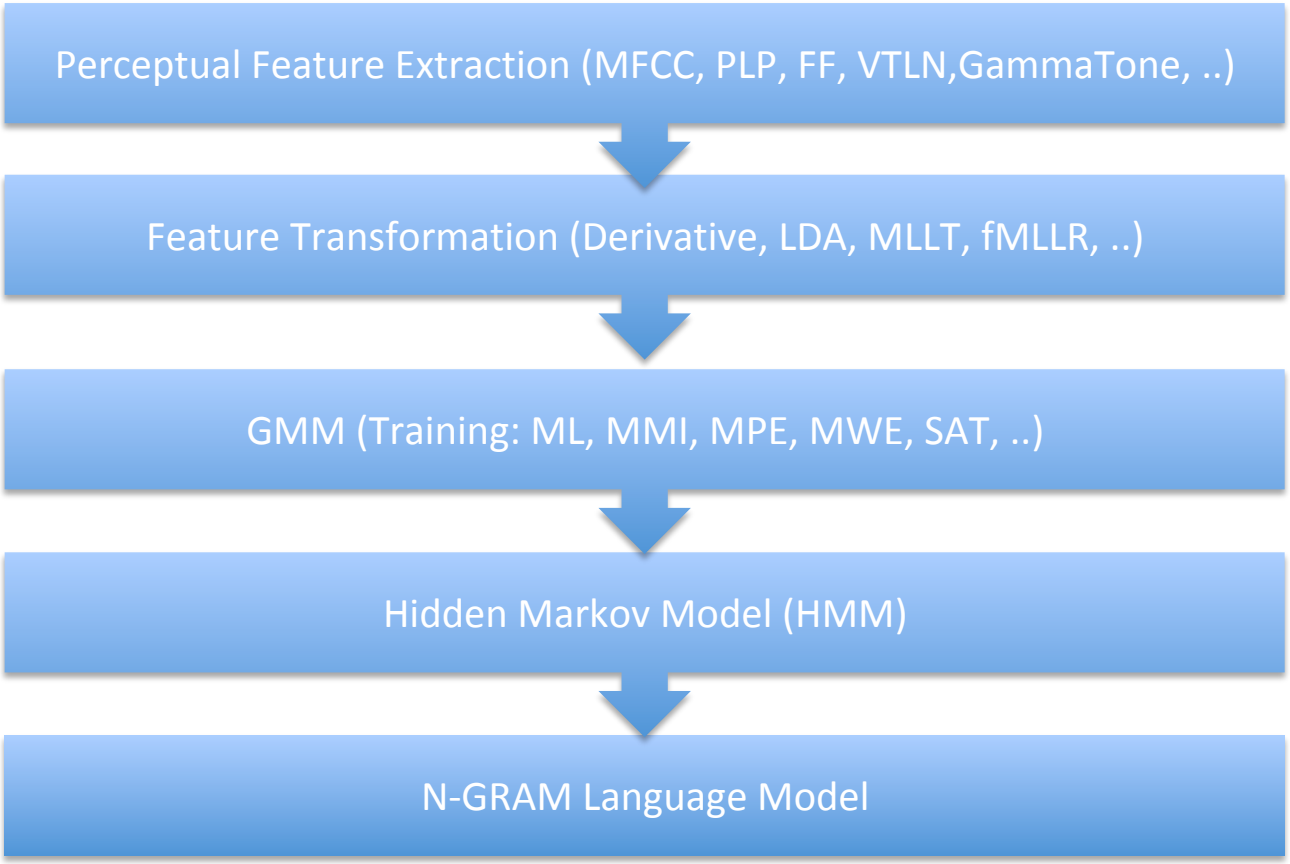


1) **Lexicon or Vocabulary**: pronunciation dictionary
   W **ER** L D -> WORLD
   HH EH L OW -> HELLO
   - Table, Rules or Machine Learning from dictionary
2) **Acoustic Model**: probability input feature x for each phone (in a phonetic context)
   p(x | **ER**) context-independent models or p(x | W-**ER**+L) context-dependent models.
   - Learned from **speech** databases (with transcription)
3) **Language Model**: probability of a sequence of words p(**w**) = p(w_1 ... $w_{|w|}$)
   Did I just say "It's fun to recognize speech?" or "It's fun to wreck a nice beach?"
   - Learned from **text**

Towards end-to-end Speech Recognition?

- Architectures
  - No neural networks.
    - GMM-HMM: 30 years of feature engineering
  - **DNN** + Hidden Markov Models + n-gram LM
    - DNN-GMM-HMM: Trained features
    - DNN-HMM: TDNN (Conv1d), BLSTM, Conv2d
  - **DNN** for language modeling
  - All-neural, **end-to-end trained**, seq2seq models

GMM-HMM



Acoustic Model

Phonetic inventory

Pronunciation Lexicon

Language Model

30 years of feature engineering!

Acoustic Modeling with GMMs

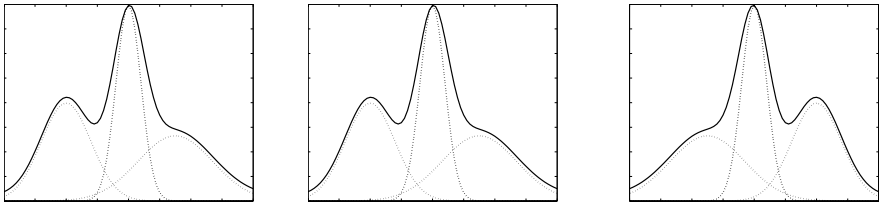
Transcription: Samson

Pronunciation: S – AE – M – S –AH – N

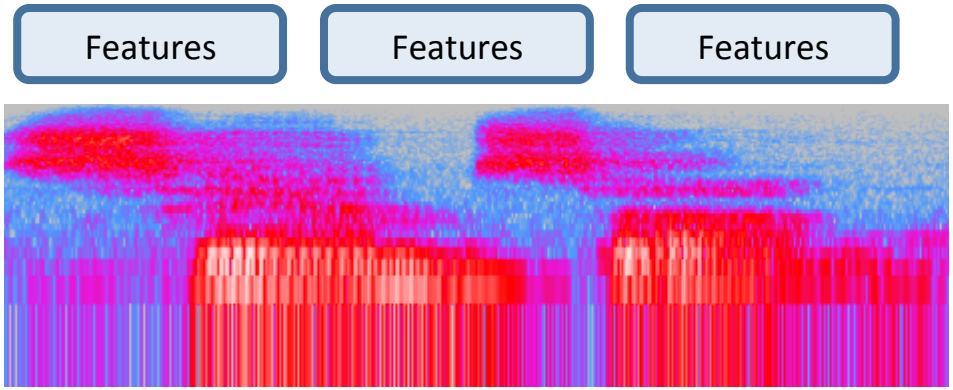
Sub-phones : 942 – 6 – 37 – 8006 – 4422 …

Hidden Markov Model (HMM):

Acoustic Model:



Audio Input:



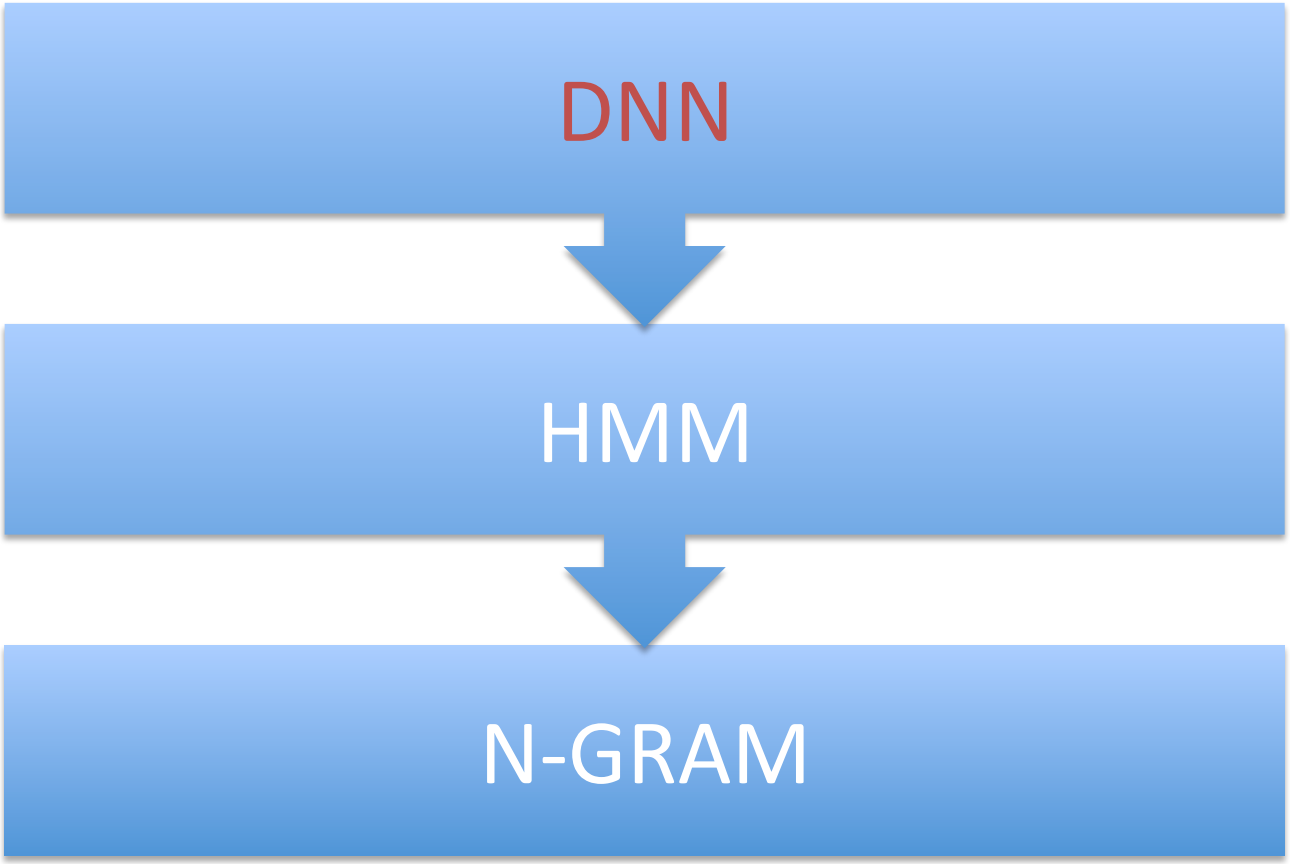
Gaussian Mixture Models:

P(x|s)

x: input features

s: HMM state

DNN-HMM



Acoustic Model

Phonetic inventory

Pronunciation Lexicon

Language Model

DNN Hybrid Acoustic Models

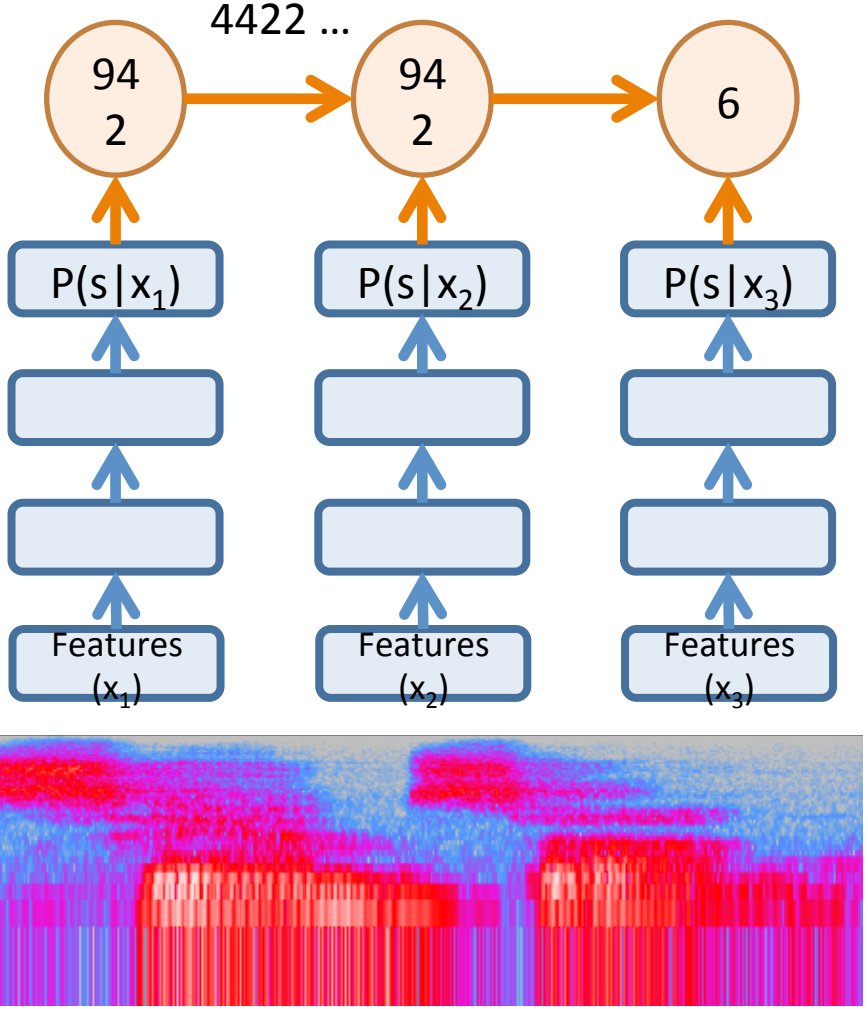
Transcription:

Pronunciation:

Sub-phones :

Hidden Markov Model (HMM):

Acoustic Model:



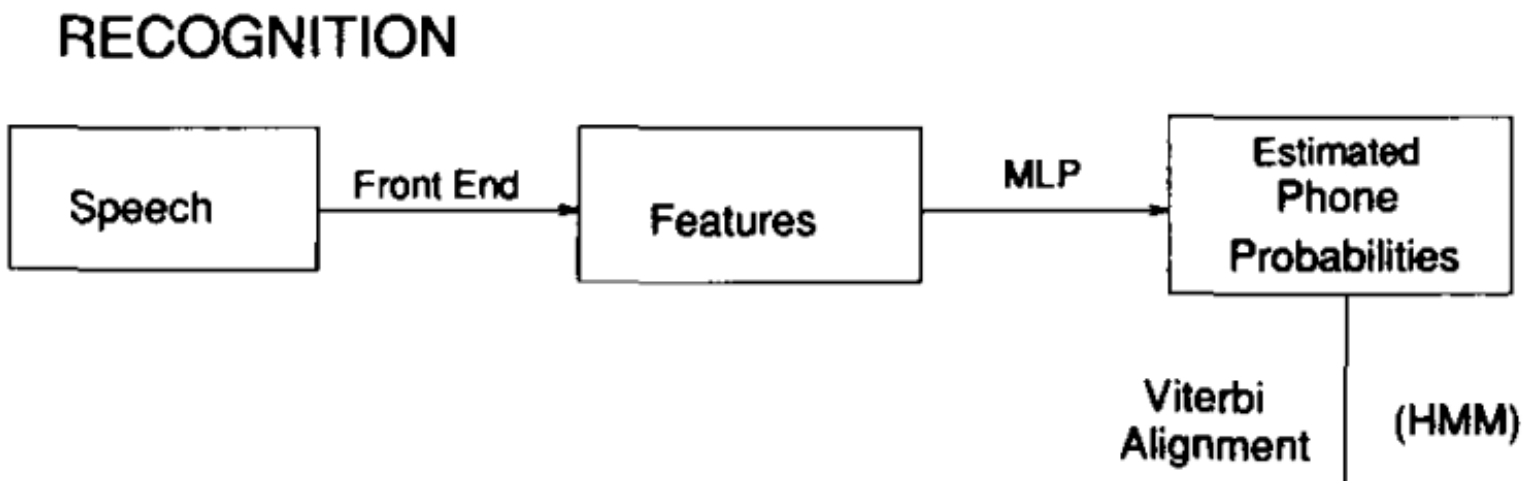
Use a DNN to approximate: P(s|x)

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

DNN * Constant / State prior

Andrew Maas. Acoustic Modeling with DNN

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 started to dominate in 2012

[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
- Larger databases 100h -> 100.000 h
- Many more parameters
- Deeper nets improve on shallow nets
- Architecture choices (RNN, Attention, ⋯)
- Regularization techniques
- Data augmentation
- Big research effort!

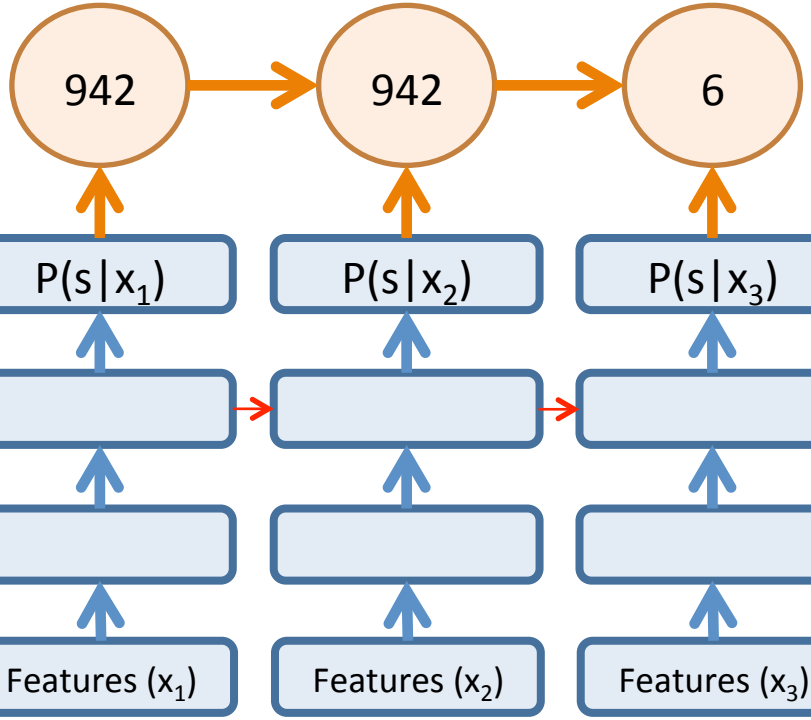
Recurrent DNN

Transcription: Samson

Pronunciation: S – AE – M – S –AH – N

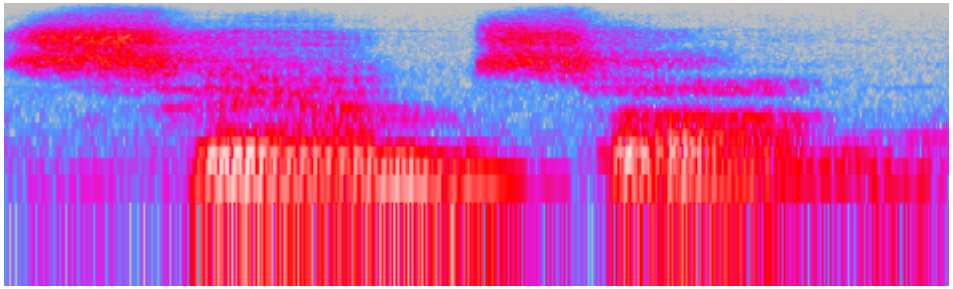
Sub-phones : 942 – 6 – 37 – 8006 – 4422 …

Hidden Markov Model (HMM):

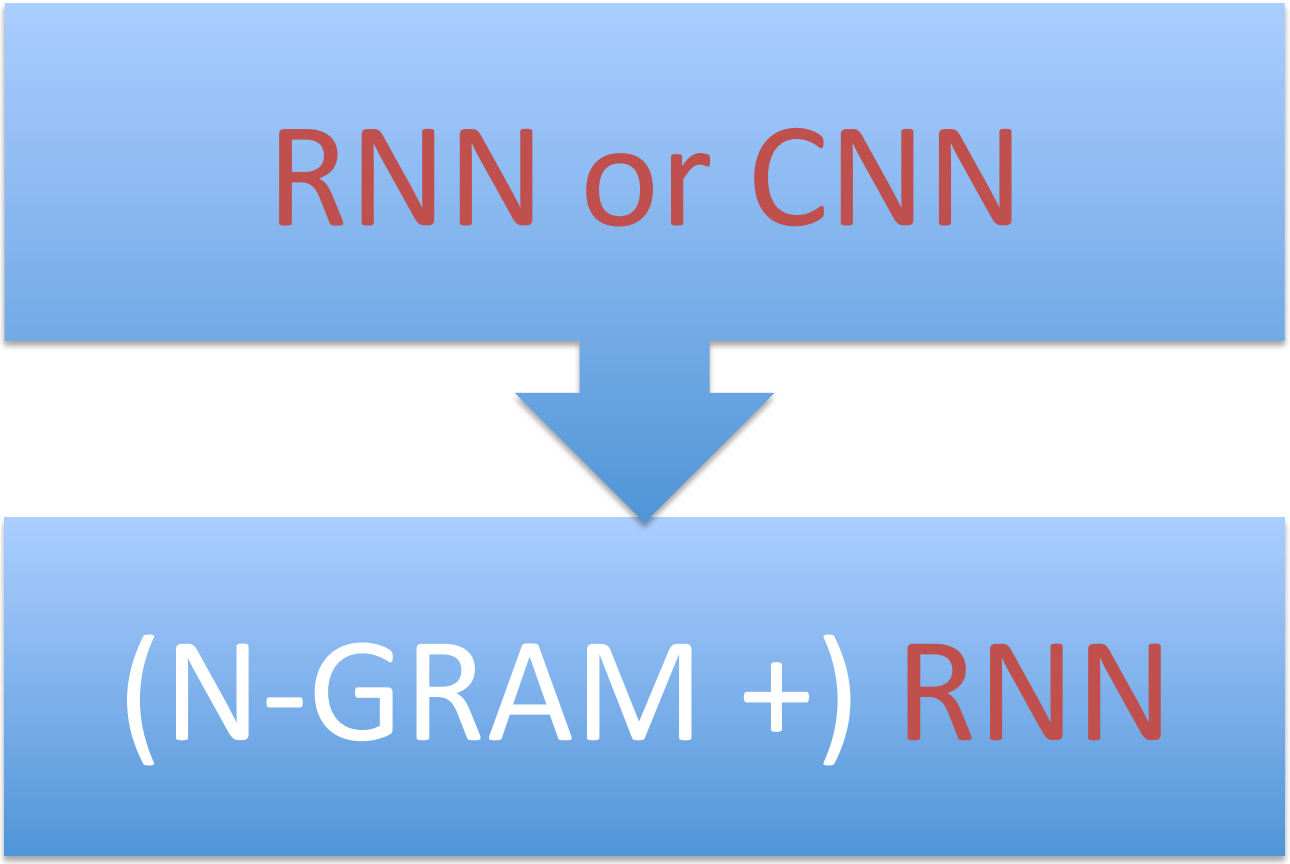


Acoustic Model:

Audio Input:



ALL NEURAL



Acoustic Model

Pronunciation Lexicon

Language Model

Deep Speech (Baidu)

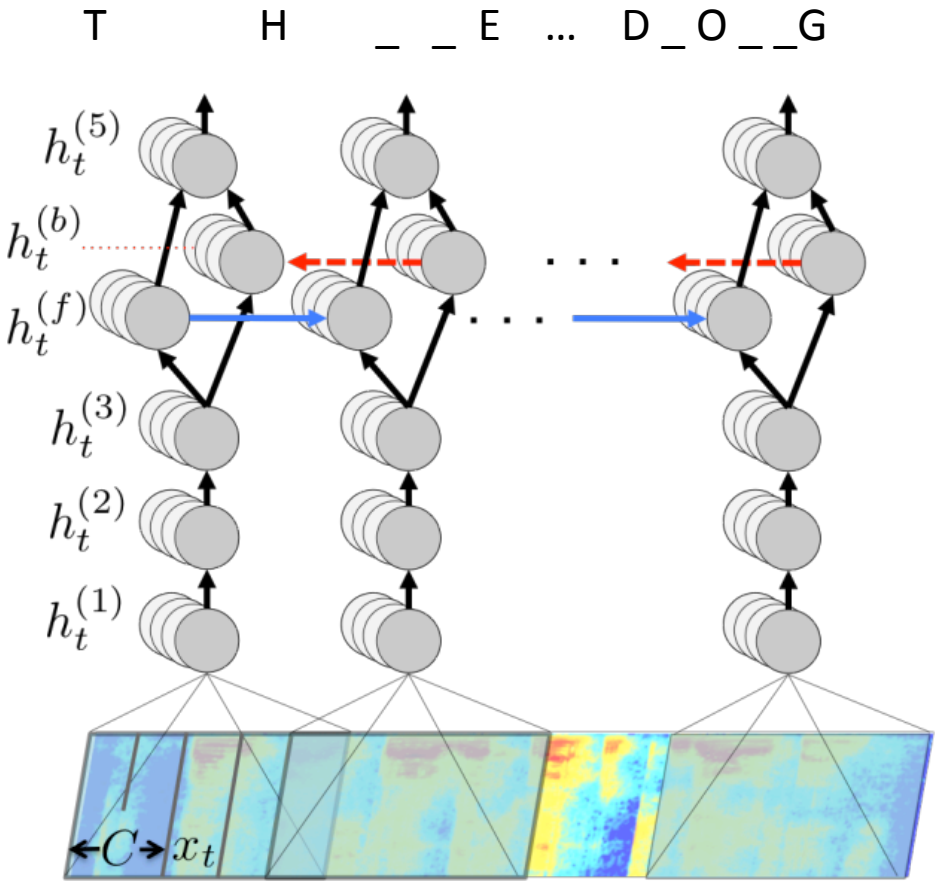
- No perceptual features (MFCC). No feature transformation. (No phonetic inventory. No transcription dictionary) No HMM.
- The output of the RNN are characters including space, apostrophe, (not CD phones)
- Connectionist Temporal Classification (No fixed aligment speech/character)
- Data augmentation. 5,000 hours (9600 speakers) + noise = 100,000 hours. Optimizations: data parallelism, model parallelism
- Good results in noisy conditions

Adam Coates

Bidirectional Recursive DNN

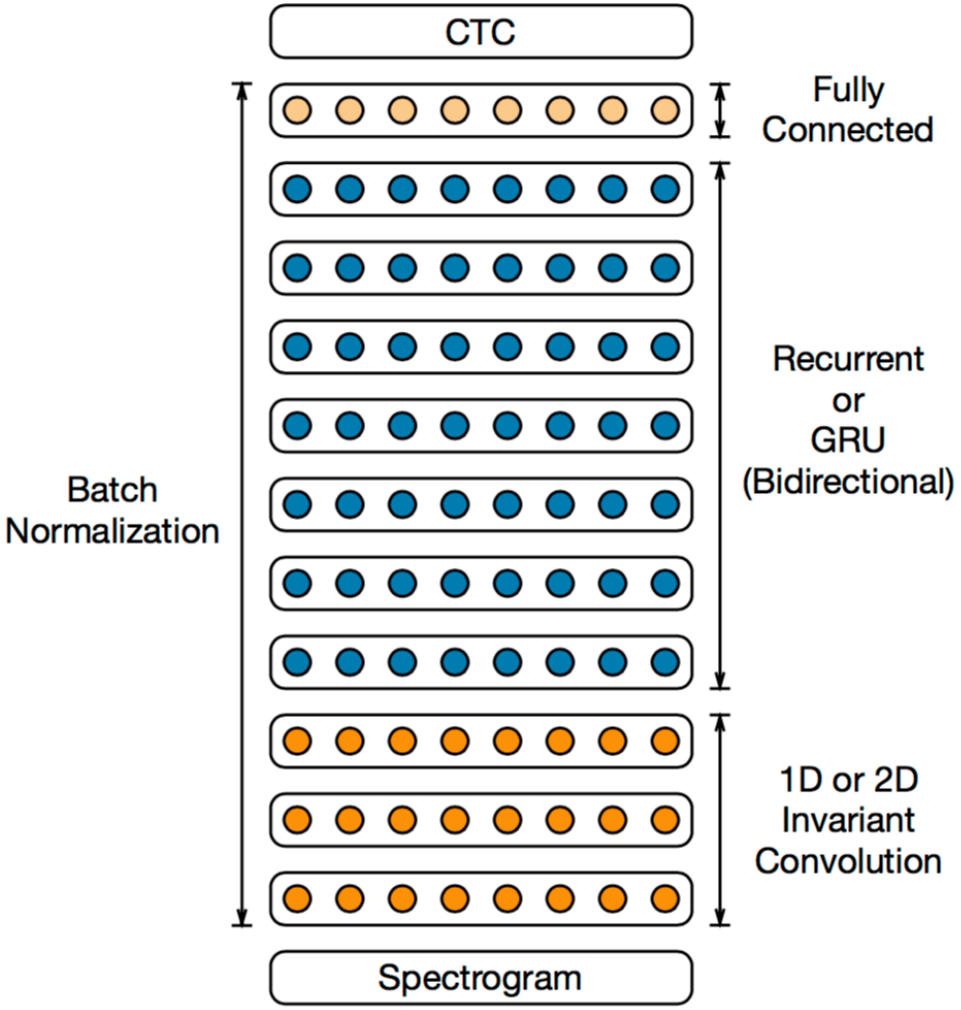
Unrolled RNN

Spectogram
Clipped ReLu
Accelerated gradient method
GPU friendly



$$h_t^{(f)} = g(W^{(4)} h_t^{(3)} + W_r^{(f)} h_{t-1}^{(f)} + b^{(4)})$$
$$h_t^{(b)} = g(W^{(4)} h_t^{(3)} + W_r^{(b)} h_{t+1}^{(b)} + b^{(4)})$$

Deep Speech II (Baidu)



Language Model

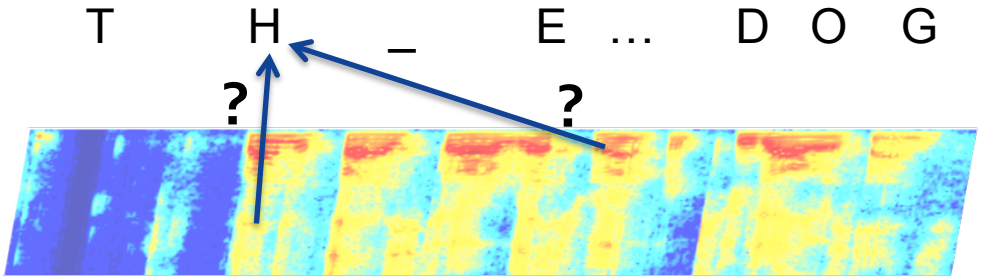
English: Kneser-Ney smoothed 5-gram model with pruning.
Vocabulary: 400,000 words from 250 million lines of text
Language model with 850 million n-grams.
Mandarin: Kneser-Ney smoothed character level 5-gram model with pruning
Training data: 8 billion lines of text.
Language model with about 2 billion n-grams.

Maximize $Q(y) = \log(p_{ctc}(y|x)) + \alpha \log(p_{lm}(y)) + \beta \text{ word\_count}(y)$

Language	Architecture	Dev no LM	Dev LM
English	5-layer, 1 RNN	27.79	14.39
English	9-layer, 7 RNN	14.93	9.52
Mandarin	5-layer, 1 RNN	9.80	7.13
Mandarin	9-layer, 7 RNN	7.55	5.81

Table 6: Comparison of WER for English and CER for Mandarin with and without a language model. These are simple RNN models with only one layer of 1D invariant convolution.

Connectionist Temporal Classification

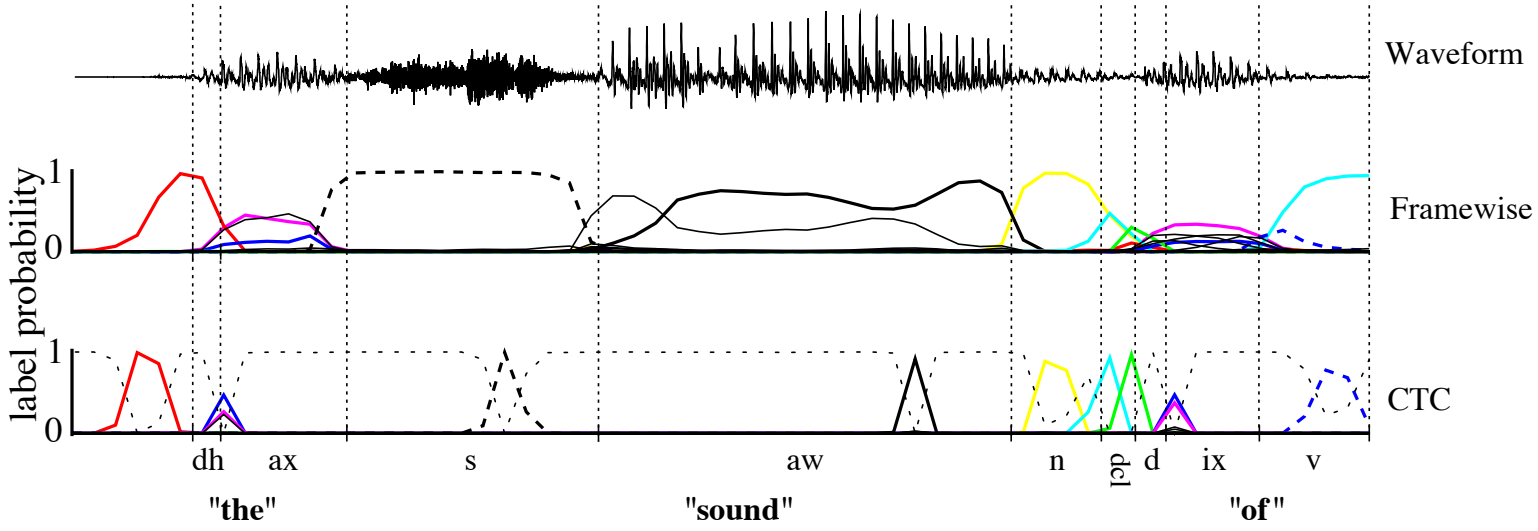


- How to connect speech data with transcription?
  - Transcription not labeled per millisecond
- Use CTC, from [Graves 06]
- Efficient dynamic programming of all possible alignments to compute error of {audio, transcription}

Bryan Catanzaro

Connectionist Temporal Classification

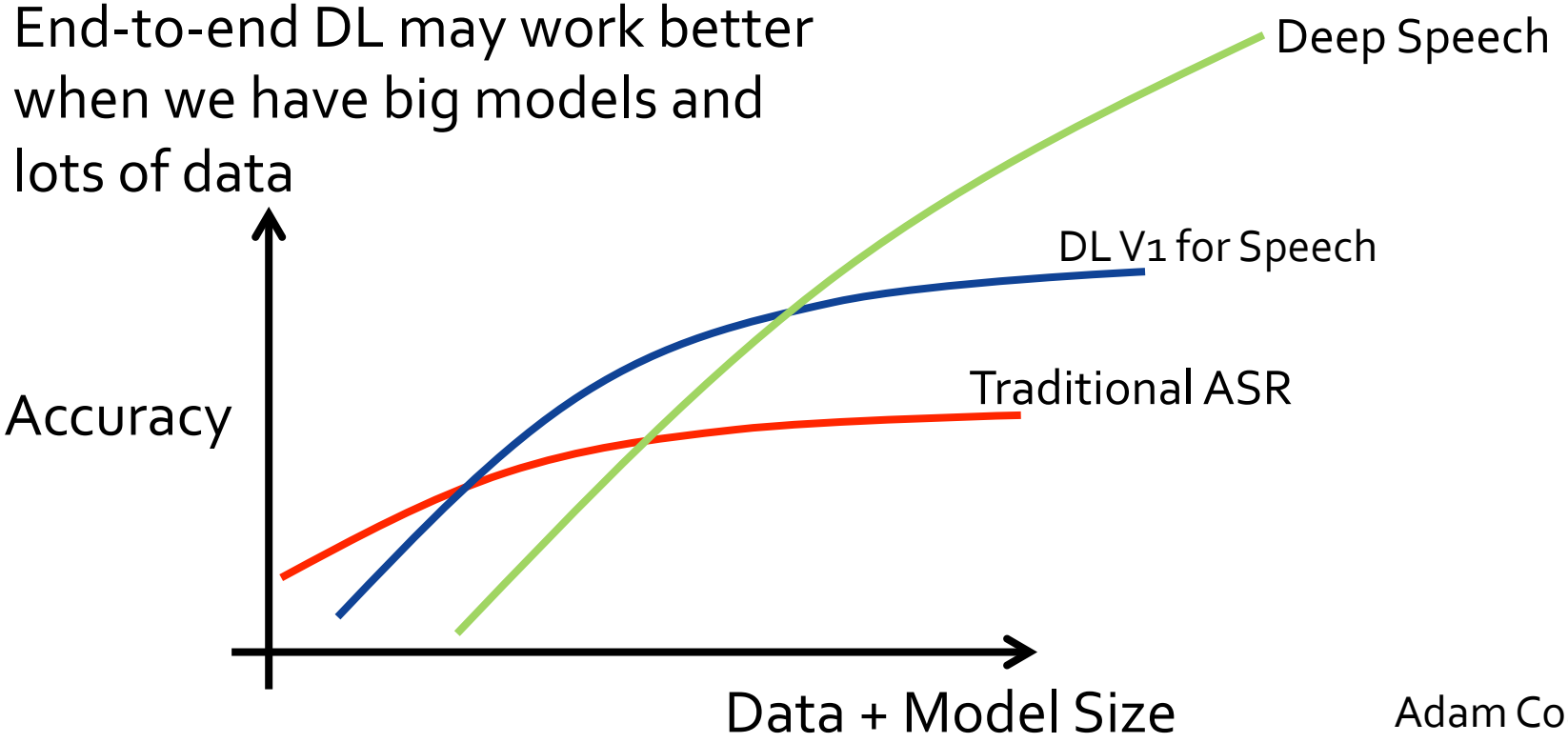
- The framewise network receives an error for misalignment
- The CTC network predicts the sequence of phonemes / characters (as spikes separated by ‘blanks’)
- No force alignment (initial model) required for training.



Alex Graves 2006

GMM-HMM / DNN-HMM / RNN

- End-to-end DL may work better when we have big models and lots of data



Adam Coates

Results
2000 HUB5 (LDC2002S09)

System	AM training data	SWB	CH
Vesely et al. (2013)	SWB	12.6	24.1
Seide et al. (2014)	SWB+Fisher+other	13.1	–
Hannun et al. (2014) Baidu	SWB+Fisher	12.6	19.3
Zhou et al. (2014)	SWB	14.2	–
Maas et al. (2014)	SWB+Fisher	15.0	23.0
Soltau et al. (2014)	SWB	10.4	19.1
Saon et al (2015) IBM	SWB+Fisher+CH	8.0	14.1
Saon et al (2016) IBM	SWB+Fisher +..	6.6	12.2
Xiong et al (2016) Microsoft	SWB+Fisher	5.9	11.1
Saon et al (2017) IBM	SWB+Fisher +..	5.5	10.3
Xiong et al (2017) Microsoft	SWB+Fisher +..	5.1	
Human performance		5.9/5.1	11.3/6.8

Hybrid(IBM) versus DS1(Baidu)

	IBM 2015	**Baidu 2014**
Features	VTL-PLP, MVN, LDA, STC, fMMLR, i-Vector	80 log filter banks
Alignment	GMM-HMM 300K Gaussians	-
DNN	DNN(5x2048) + CNN(128x9x9+5x2048) + +RNN 32000 outputs	4RNN (5 x 2304) 28 outputs
DNN Training	CE + MBR Discriminative Training (ST)	CTC
HMM	32K states (DNN outputs) pentaphone acoustic context	-
Language Model	37M 4-gram + model M (class based exponential model) + 2 NNLM	4-gram (Transcripts)

DS1 versus DS2 (Baidu)

	Deep Speech 1 (Baidu 2014)	**DS2 (Baidu 2015)**
Features	80 log filter banks	?
Alignment	-	-
DNN	4RNN (5 x 2304) 28 outputs	9-layer, 7RNN, BatchNorm, Conv. Layers. (Time/Freq)
DNN Training	CTC	CTC
HMM	-	-
Language Model	4-gram	5-gram

Deep Speech 2 Versus DS1

Test set	DS1	DS2
Baidu Test	24.01	13.59

Read Speech			
Test set	DS1	DS2	Human
WSJ eval'92	4.94	3.60	5.03
WSJ eval'93	6.94	4.98	8.08
LibriSpeech test-clean	7.89	5.33	5.83
LibriSpeech test-other	21.74	13.25	12.69

Deep Speech 2 Versus DS1

Accented Speech			
Test set	DS1	DS2	Human
VoxForge American-Canadian	15.01	7.55	4.85
VoxForge Commonwealth	28.46	13.56	8.15
VoxForge European	31.20	17.55	12.76
VoxForge Indian	45.35	22.44	22.15

Table 14: Comparing WER of the DS1 system to the DS2 system on accented speech.

Noisy Speech			
Test set	DS1	DS2	Human
CHiME eval clean	6.30	3.34	3.46
CHiME eval real	67.94	21.79	11.84
CHiME eval sim	80.27	45.05	31.33

DS2 Training data

Dataset	Speech Type	Hours
WSJ	read	80
Switchboard	conversational	300
Fisher	conversational	2000
LibriSpeech	read	960
Baidu	read	5000
Baidu	mixed	3600
Total		11940

END-TO-END

Acoustic Model

Pronunciation Lexicon

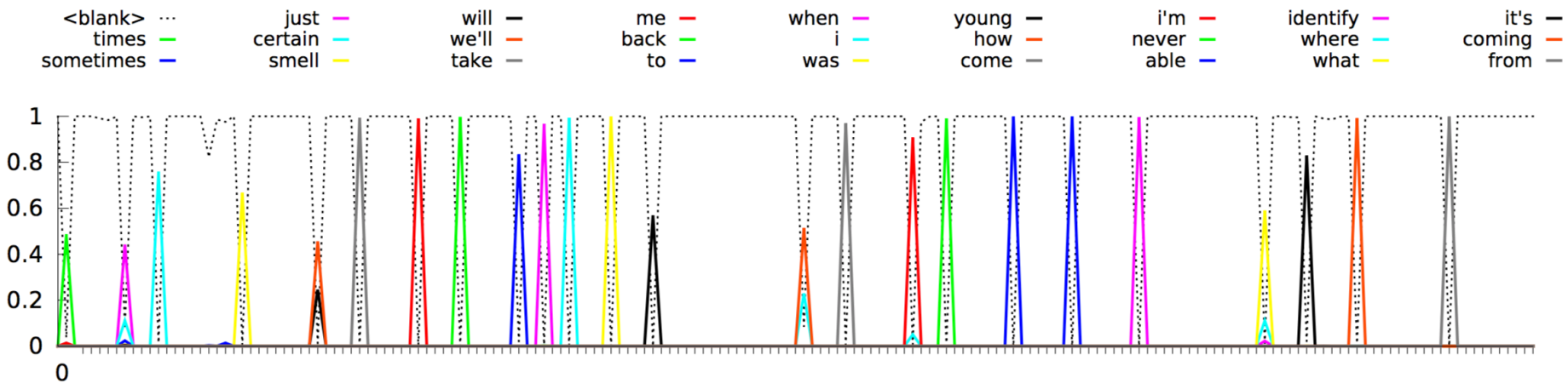
Language Model

Acoustic-to-Word LSTM

Google, 2016, **125.000 hours** of training data

BLSTM RNN with a CTC loss that predicts words

100K words including numeric entities



ATTENTION

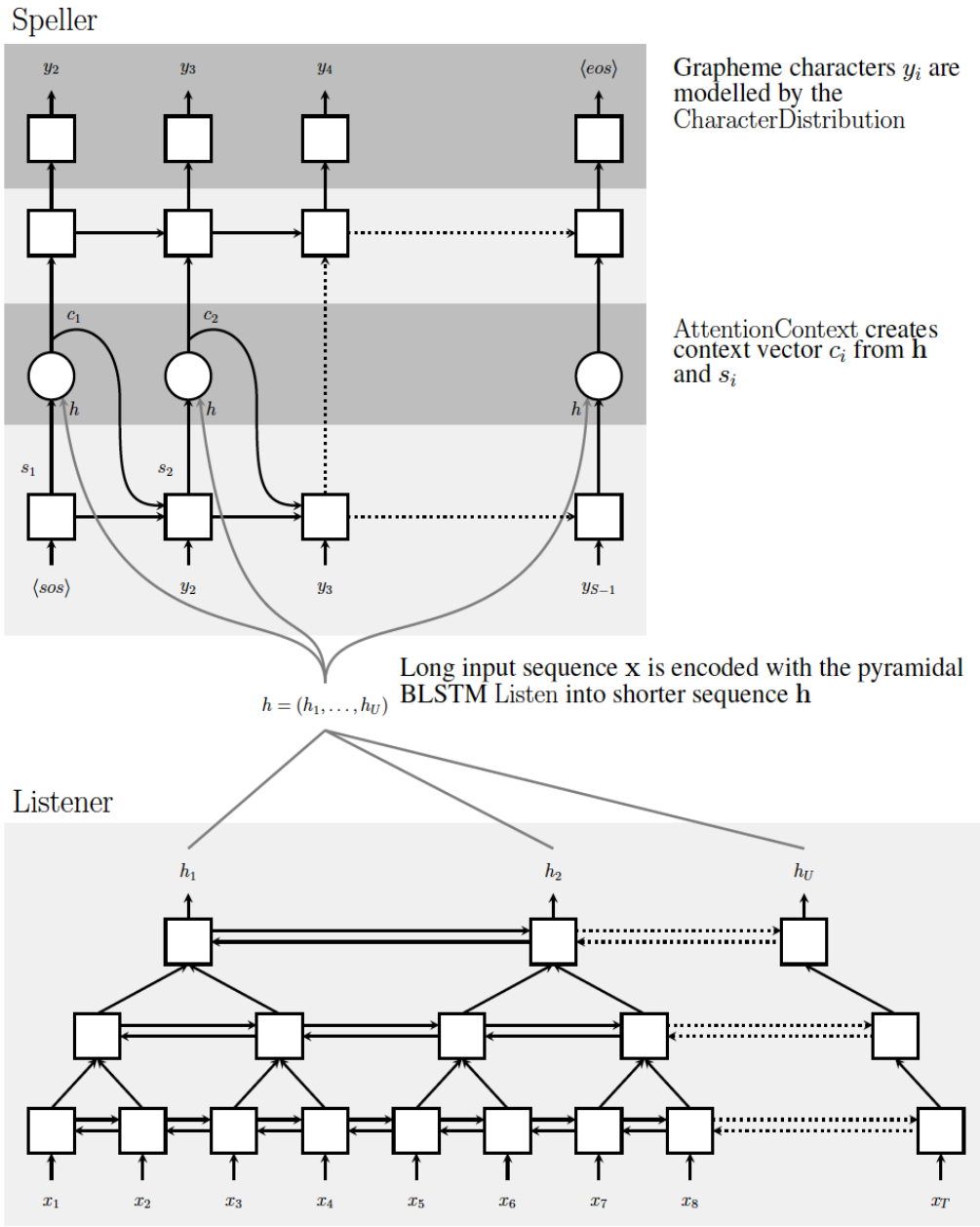
1) Listen, Attend and Spell (LAS-2015)

No independence assumptions between the characters.

Key improvement of LAS over previous end-to-end CTC models

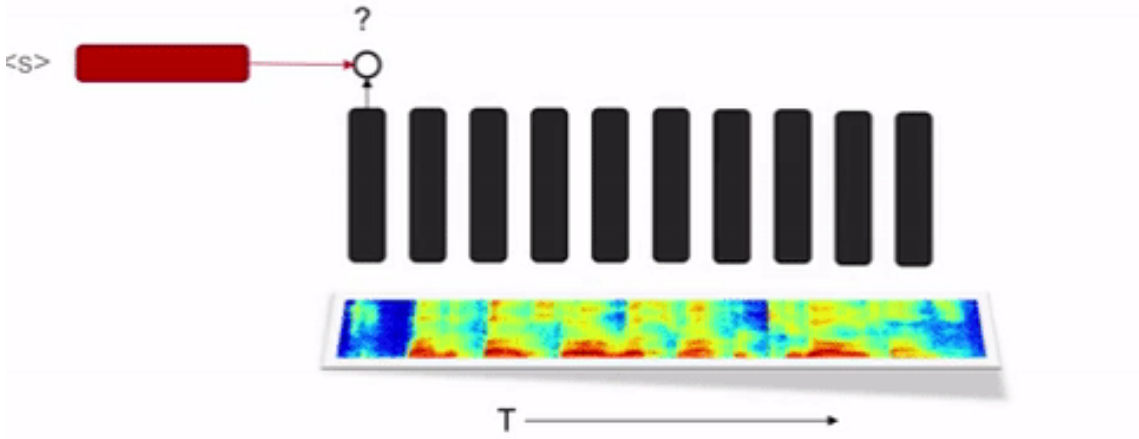
2) Improved version (LAS-2018)

Word piece models
Unidirectional LSTM for streaming
Multi-headed attention
Minimum Word Error Rate training
WER 9.2% -> 5.6 % (LSTM HMM: 6.7)



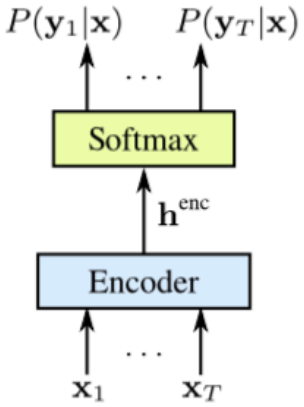
RNN Transducer

- The RNN-Transducer assumes the alignment between input and output tokens is local and monotonic.
- Better fit for speech recognition than attention-based Seq2Seq models by removing extra hacks applied to attentional models to encourage monotonicity.
- No external language model

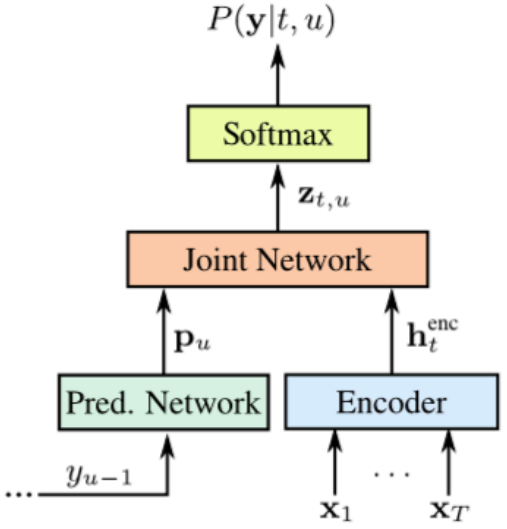


Exploring Neural Transducers for End-to-End Speech Recognition, Baidu, Jul 2017

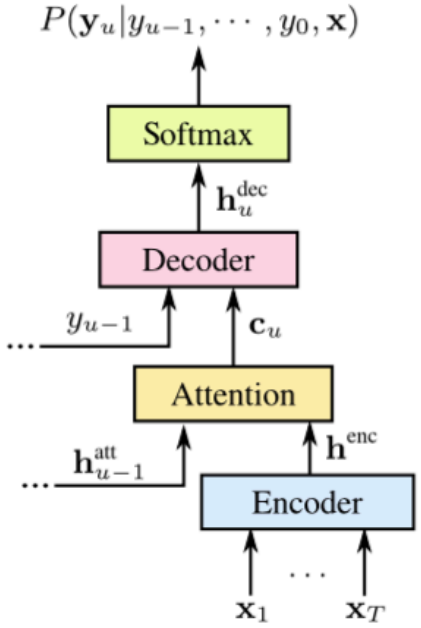
Transducer with Attention



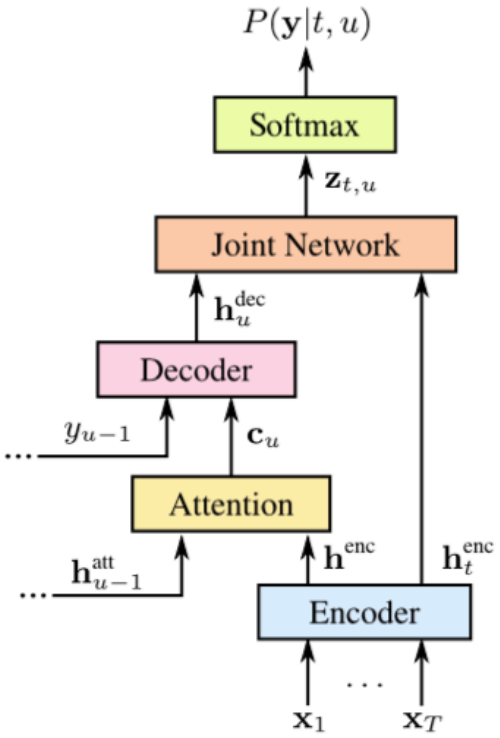
(a.) CTC



(b.) RNN-Transducer



(c.) Attention-based Model



(d.) RNN-Transducer with Attention

A Comparison of Sequence-to-Sequence Models for Speech Recognition, Google 2017