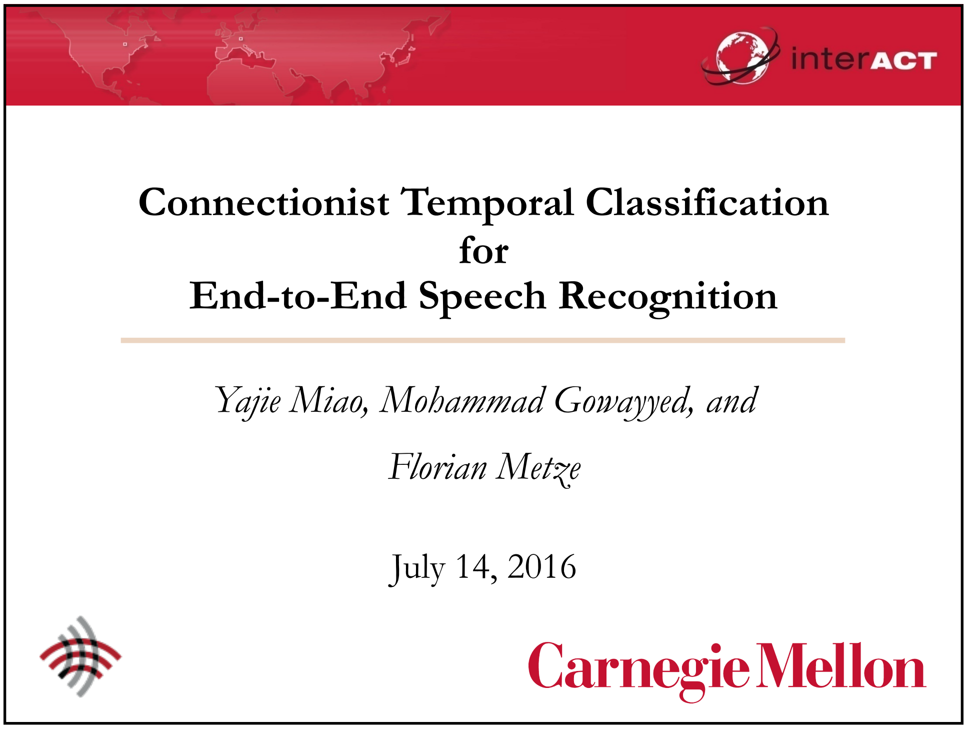


interACT

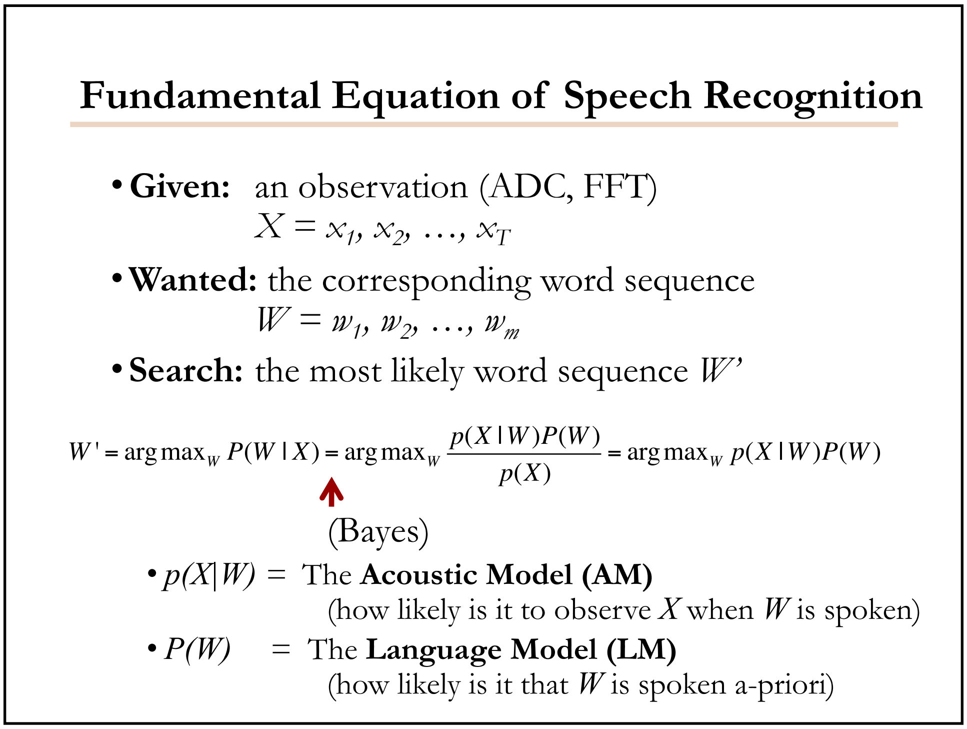
# Connectionist Temporal Classification for End-to-End Speech Recognition

*Yajie Miao, Mohammad Gowayyed, and*

*Florian Metze*

July 14, 2016

Carnegie Mellon



## Fundamental Equation of Speech Recognition

- **Given:** an observation (ADC, FFT) $X = x_1, x_2, \ldots, x_T$
- **Wanted:** the corresponding word sequence $W = w_1, w_2, \ldots, w_m$
- **Search:** the most likely word sequence *W'*

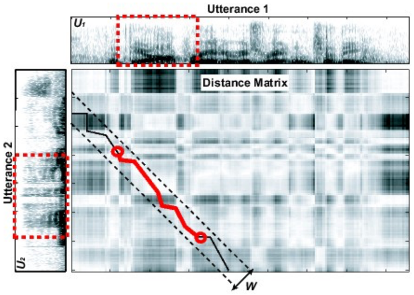
$$W' = \arg\max_W P(W \mid X) \underset{\uparrow}{=} \arg\max_W \frac{p(X \mid W)P(W)}{p(X)} = \arg\max_W p(X \mid W)P(W)$$

(Bayes)

- $p(X|W)$ = The **Acoustic Model (AM)** (how likely is it to observe *X* when *W* is spoken)
- $P(W)$ = The **Language Model (LM)** (how likely is it that *W* is spoken a-priori)

## Recognition Conceptually: AM and LM

- Let’s be pragmatic and keep AM and LM separate
- Simply count to get *P(W)*
- How to get an estimate for *p(X|W)*?
  - Take “spectrograms” and compare the recordings of two utterances using DTW
  - Accumulate cost along best path, using Hidden Markov Model (instead of 2nd utterance)



*Word Acquisition Using Unsupervised Acoustic Pattern Discovery*
Alex S. Park & James R. Glass. 2006.

## Hidden Markov Models

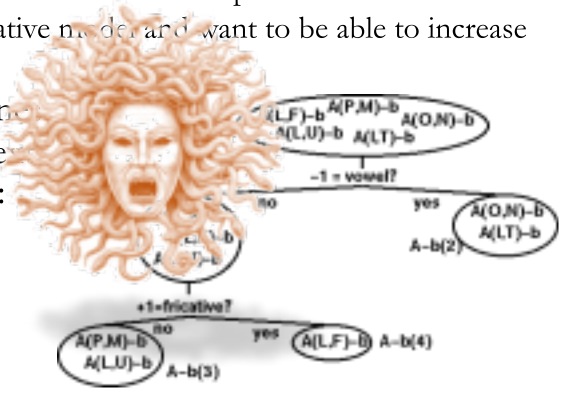
A “Hidden Markov Model” is a 5-tupel consisting of:

- *S* The set of **states** $S=\{s_1,s_2,\ldots,s_n\}$, n is the number of states
- π The **initial probability distribution** , $\pi(s_i) = P(q_1 = s_i)$ probability of $s_i$ being the first state of a sequence
- *A* The matrix of **state transition probabilities:** $1 \leq i, j \leq n$ $A=(a_{ij})$ with $a_{ij} = P(q_{t+1}= s_j|q_t = s_i)$ going from state $s_i$ to $s_j$
- *B* The set of **emission probability distributions/ densities,** $B=\{b_1,b_2,\ldots,b_n\}$ where $b_i(x)=P(o_t = x|q_t = s_i)$ is the probability of observing *x* when the system is in state $s_i$
- *V* Set of symbols, *v* is the number of distinct symbols. The observable **feature space** can be discrete: $V=\{x_1,x_2,\ldots,x_v\}$, or continuous $V=\boldsymbol{R}^d$

## Context-Dependent States

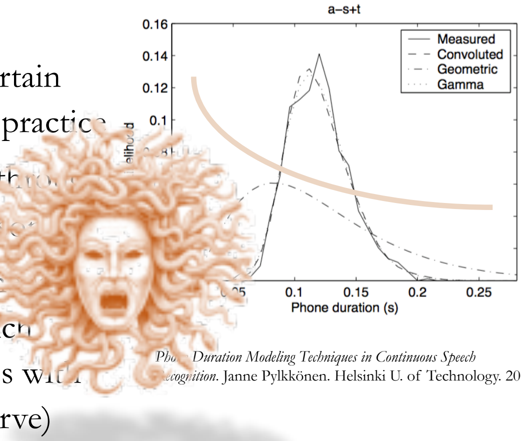
- Not complicated enough?
  - No – co-articulation influences the pronunciation
  - We have a generative model and want to be able to increase the model size
- Cluster ~50 phon
  into ~5000 conte
  dependent states:

  A(F,L)-b, etc.
- It's ugly!



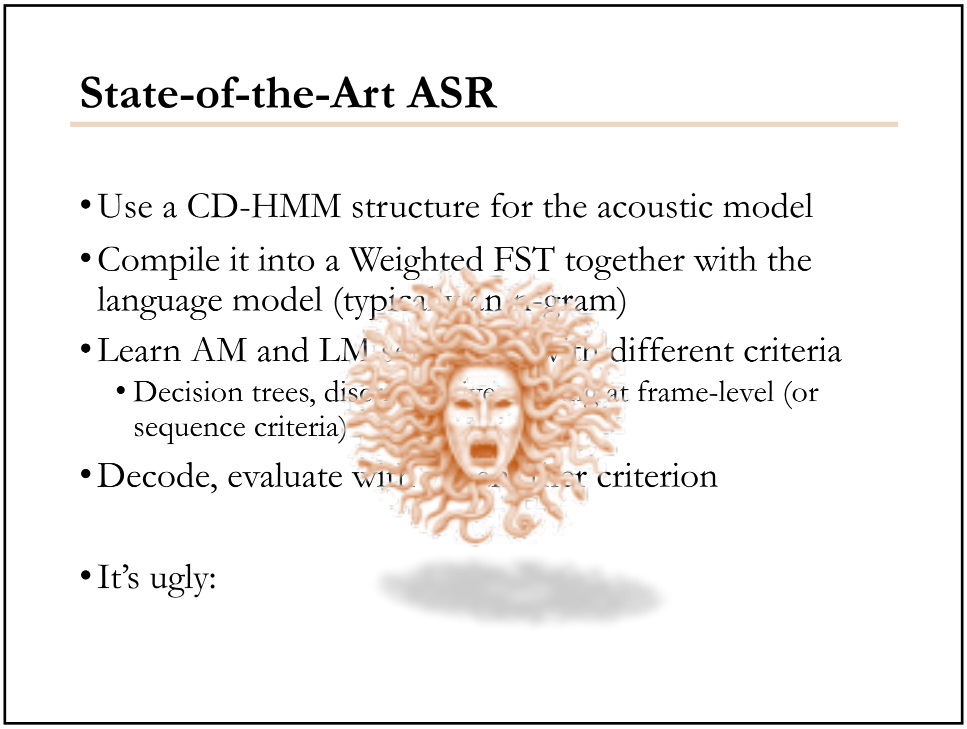
## Duration Modeling

- Phonemes have a certain minimal duration in practice
- We could fit curves th
  them (Gamma, Pois
- But we use an expo
  decay for states (which
  approximates phones wi
  the "convoluted" curve)
- It's ugly!



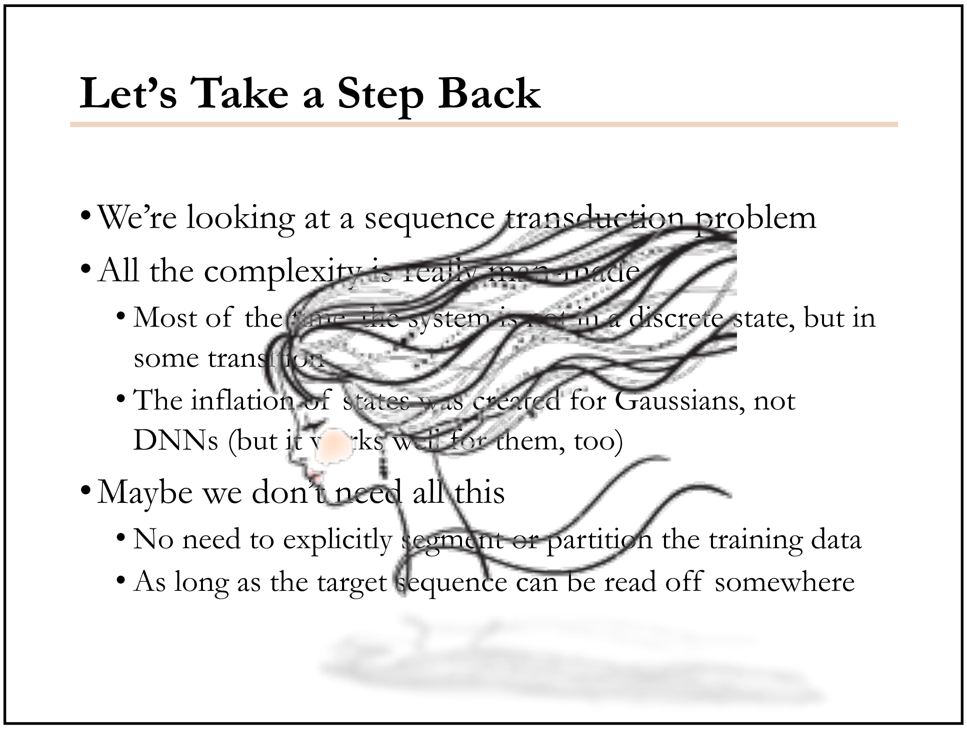
*Phone Duration Modeling Techniques in Continuous Speech Recognition.* Janne Pylkkönen. Helsinki U. of Technology. 2004.

```
SIL   { { 0  0.01 } { 1  0.0 } }
1     { { 0  0.01 } { 1  0.0 } }
3     { { 0  0.01 } { 1  0.0 } { 2  0.015 } }
```



## State-of-the-Art ASR

- Use a CD-HMM structure for the acoustic model
- Compile it into a Weighted FST together with the language model (typically an n-gram)
- Learn AM and LM separately with different criteria
  - Decision trees, discriminative training at frame-level (or sequence criteria)
- Decode, evaluate with yet another criterion
- It's ugly:



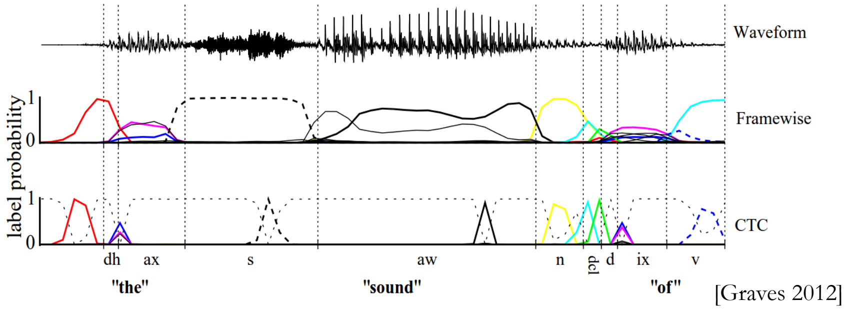
## Let's Take a Step Back

- We're looking at a sequence transduction problem
- All the complexity is really man-made
  - Most of the time, the system is not in a discrete state, but in some transition
  - The inflation of states was created for Gaussians, not DNNs (but it works well for them, too)
- Maybe we don't need all this
  - No need to explicitly segment or partition the training data
  - As long as the target sequence can be read off somewhere

# Connectionist Temporal Classification

- Alex Graves (2006) described the “CTC” loss function
  - Sum over all possible frame alignments permitted for output sequence using Forward-Backward
  - Plays well with RNN or LSTM neural network models
- CTC introduces a new symbol: blank (-)
  - “Cannot decide with confidence given the current information”
  - “No output”, but do not confuse with silence
- Most of the time, the network will output (-)
  - Class im-balance not a problem in a connectionist architecture
  - As long as the target symbols appear from time to time

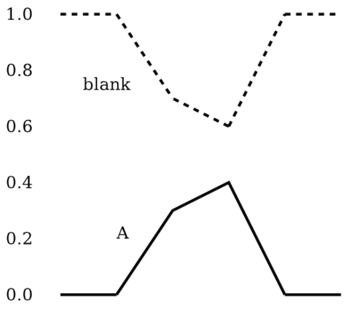
# Observations



[Graves 2012]

- Sparse representations (spikes) appear
- Any modeling unit can be used: phone, syllable, word

## Problem with Best Path Decoding



p(l=blank) = p(- -)
= 0.7*0.6
= 0.42

p(l=A) = p(AA)+p(A-)+p(-A)
= 0.3*0.4 + 0.3*0.6 + 0.7*0.4
= 0.58

**Fig. 7.5 Problem with best path decoding.** The single most probable path contains no labels, and best path decoding therefore outputs the labelling 'blank'. However the combined probabilities of the paths corresponding to the labelling 'A' is greater.

[Graves 2012]

## Enter WFST Decoding

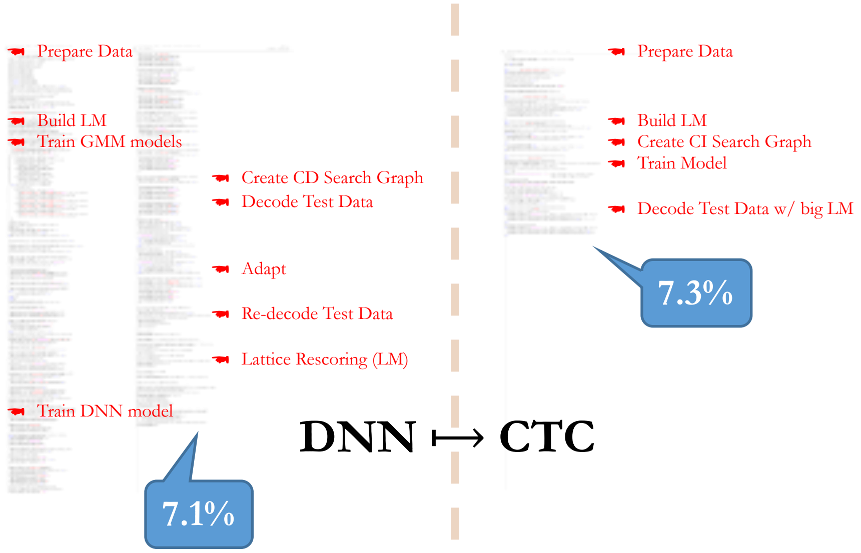
- Turns out WFSTs can decode CTC-AMs well
  - FSTs can map *aaaa*, *-aa-*, *---a* , … to "A"
- Resulting FSTs will typically be <u>much smaller</u>
  - S = T ∘ min(det(L ∘ G))
- Traditional HCLG
  - S = min(det(H ∘ min(det(C ∘ min(det(L ∘ G))))))
  - Don't need HMM and Context FST any more, can replace by extremely simple Token FST
- Need to do some work on normalization of posteriors
  - Our experiments show it is most reliable to simply count the phones – which is also the simplest solution

## Results on Read Speech

- WSJ: Baseline is unadapted Kaldi lMEL DNN
  - 20k vocabulary, quite optimized
  - Similar numbers for 5k vocab
  - No gains by going to CD models in our experiments (although Google reports gains)

| Task (WER) | Traditional | CTC | Remark |
|---|---|---|---|
| WSJ | 7.1% | 7.3% | CI Phones |
| | | 8.9% | Characters |
| | | 9.3% | Syllables |

## Detox of (Kaldi WSJ) Training Scripts



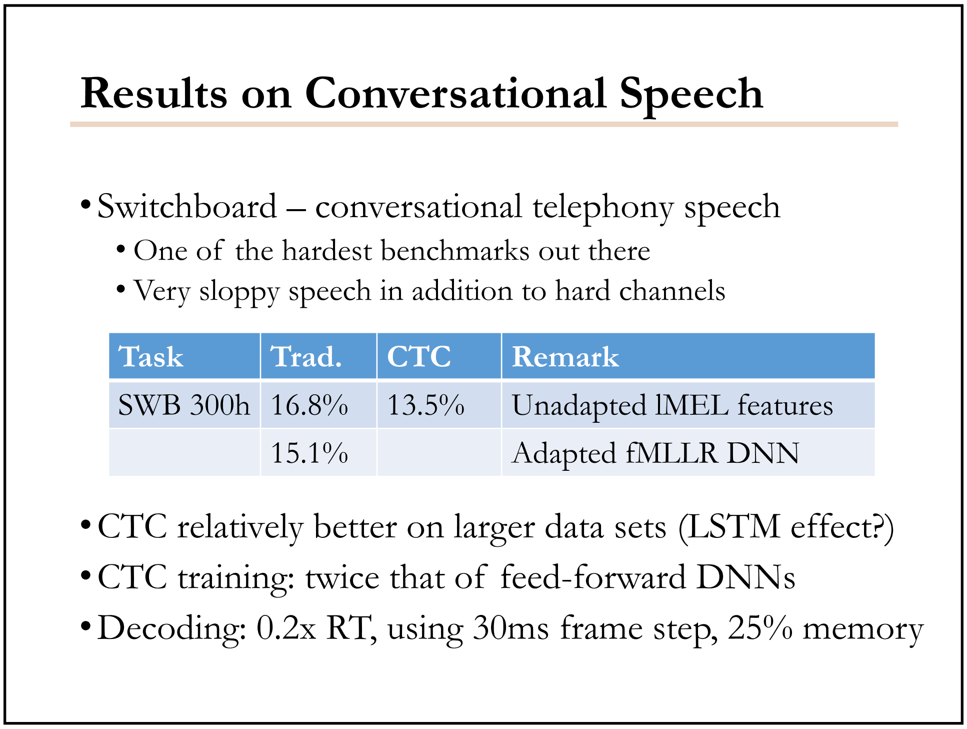
- Prepare Data
- Build LM
- Train GMM models
- Create CD Search Graph
- Decode Test Data
- Adapt
- Re-decode Test Data
- Lattice Rescoring (LM)
- Train DNN model

7.1%

- Prepare Data
- Build LM
- Create CI Search Graph
- Train Model
- Decode Test Data w/ big LM

7.3%

DNN ⟼ CTC

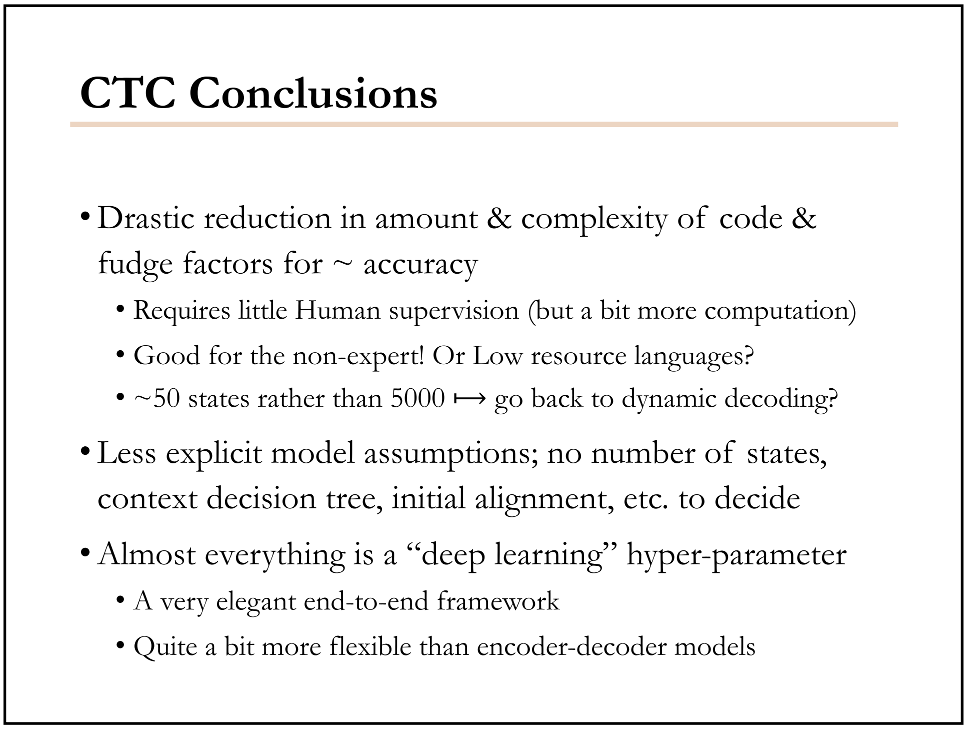


## Results on Conversational Speech

- Switchboard – conversational telephony speech
  - One of the hardest benchmarks out there
  - Very sloppy speech in addition to hard channels

| Task | Trad. | CTC | Remark |
|---|---|---|---|
| SWB 300h | 16.8% | 13.5% | Unadapted lMEL features |
| | 15.1% | | Adapted fMLLR DNN |

- CTC relatively better on larger data sets (LSTM effect?)
- CTC training: twice that of feed-forward DNNs
- Decoding: 0.2x RT, using 30ms frame step, 25% memory



## CTC Conclusions

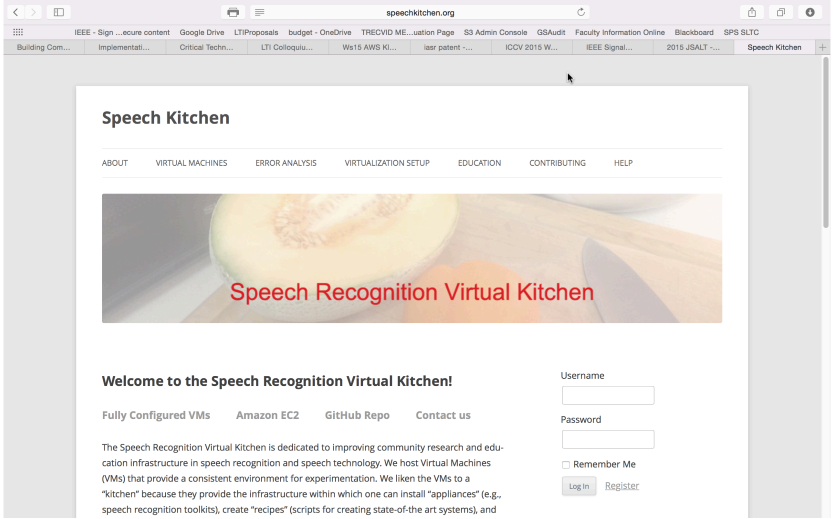
- Drastic reduction in amount & complexity of code & fudge factors for ~ accuracy
  - Requires little Human supervision (but a bit more computation)
  - Good for the non-expert! Or Low resource languages?
  - ~50 states rather than 5000 ⟼ go back to dynamic decoding?
- Less explicit model assumptions; no number of states, context decision tree, initial alignment, etc. to decide
- Almost everything is a "deep learning" hyper-parameter
  - A very elegant end-to-end framework
  - Quite a bit more flexible than encoder-decoder models

# Thank You!

Questions? ⟼ fmetze@cs.cmu.edu

Y. Miao, M. Gowayyed, and F. Metze: EESEN - END-TO-END SPEECH RECOGNITION USING DEEP RNN MODELS AND WFST-BASED DECODING. In *Proc. ASRU*, Scottsdale, AZ; U.S.A., Dec 2015. IEEE. https://github.com/srvk/eesen.

# http://speechkitchen.org/



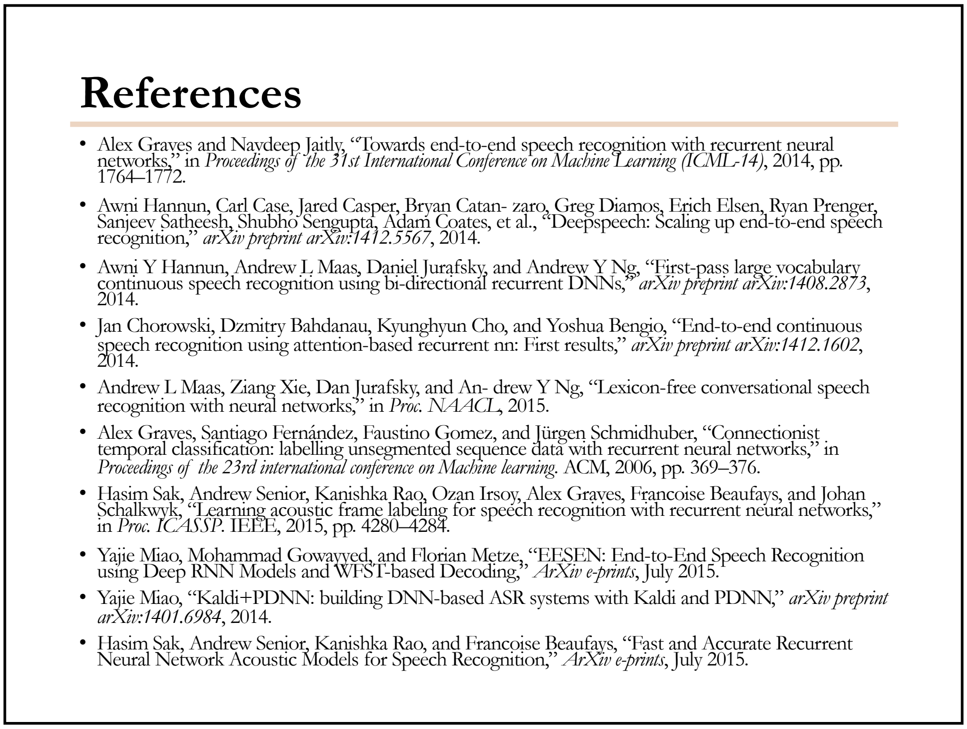
## Welcome to the Speech Recognition Virtual Kitchen!

The Speech Recognition Virtual Kitchen is dedicated to improving community research and education infrastructure in speech recognition and speech technology. We host Virtual Machines (VMs) that provide a consistent environment for experimentation. We liken the VMs to a "kitchen" because they provide the infrastructure within which one can install "appliances" (e.g., speech recognition toolkits), create "recipes" (scripts for creating state-of-the art systems), and

Username

Password

□ Remember Me



# References

- Alex Graves and Navdeep Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, 2014, pp. 1764–1772.
- Awni Hannun, Carl Case, Jared Casper, Bryan Catan- zaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, et al., "Deepspeech: Scaling up end-to-end speech recognition," *arXiv preprint arXiv:1412.5567*, 2014.
- Awni Y Hannun, Andrew L Maas, Daniel Jurafsky, and Andrew Y Ng, "First-pass large vocabulary continuous speech recognition using bi-directional recurrent DNNs," *arXiv preprint arXiv:1408.2873*, 2014.
- Jan Chorowski, Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio, "End-to-end continuous speech recognition using attention-based recurrent nn: First results," *arXiv preprint arXiv:1412.1602*, 2014.
- Andrew L Maas, Ziang Xie, Dan Jurafsky, and An- drew Y Ng, "Lexicon-free conversational speech recognition with neural networks," in *Proc. NAACL*, 2015.
- Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd international conference on Machine learning*. ACM, 2006, pp. 369–376.
- Hasim Sak, Andrew Senior, Kanishka Rao, Ozan Irsoy, Alex Graves, Francoise Beaufays, and Johan Schalkwyk, "Learning acoustic frame labeling for speech recognition with recurrent neural networks," in *Proc. ICASSP*. IEEE, 2015, pp. 4280–4284.
- Yajie Miao, Mohammad Gowayyed, and Florian Metze, "EESEN: End-to-End Speech Recognition using Deep RNN Models and WFST-based Decoding," *ArXiv e-prints*, July 2015.
- Yajie Miao, "Kaldi+PDNN: building DNN-based ASR systems with Kaldi and PDNN," *arXiv preprint arXiv:1401.6984*, 2014.
- Hasim Sak, Andrew Senior, Kanishka Rao, and Francoise Beaufays, "Fast and Accurate Recurrent Neural Network Acoustic Models for Speech Recognition," *ArXiv e-prints*, July 2015.