

Advanced Kaldi Toolkit Features

Yuan-Fu Liao

National Taipei University of Technology

yfliao@ntutedu.tw



Staying Ahead in the STT Game

- STT technology is advancing very rapidly
 - Amazon, Apple, Baidu, Facebook, Google, Microsoft
- Kaldi leads and keeps up with major innovations
 - From SGMMs to DNN(2012)
 - From “English” to low-resource languages(2013)
 - From CPUs to GPUs(2014)
 - From close-talking to far-field microphones(2015)
 - Chain models for better STT, faster decoding (2017)



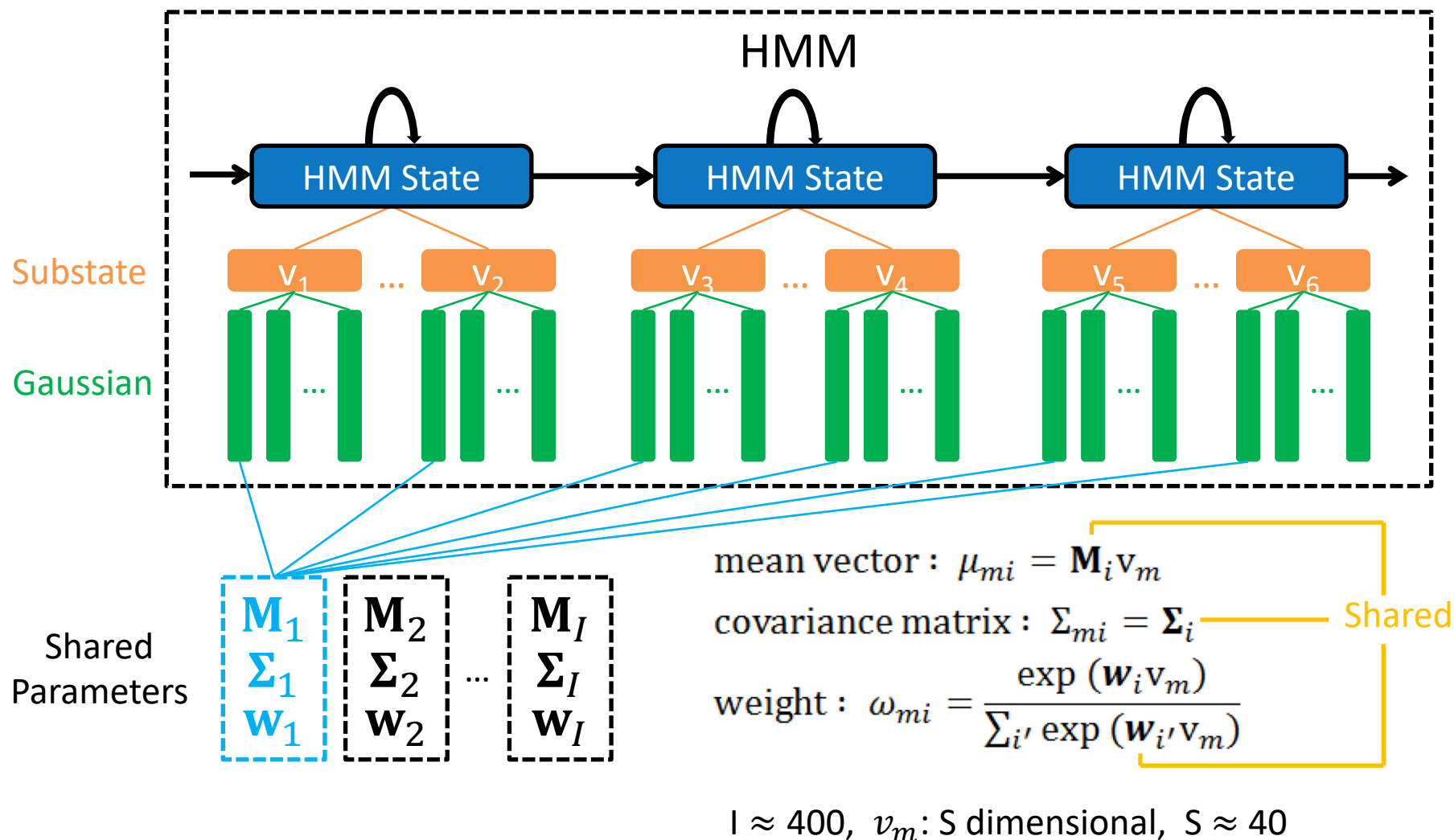
Staying Ahead in the STT Game

- STT technology is advancing very rapidly
 - Amazon, Apple, Baidu, Facebook, Google, Microsoft
- Kaldi leads and keeps up with major innovations
 - **From SGMMs to DNN(2012)**
 - From “English” to low-resource languages (2013)
 - From CPUs to GPUs (2014)
 - From close-talking to far-field microphones (2015)
 - Chain models for better STT, faster decoding (2017)

Subspace Gaussian Mixture Model

A Triphone HMM in Subspace GMM

"The Subspace Gaussian Mixture Model– a Structured Model for Speech Recognition", D. Povey, Lukas Burget et. al Computer Speech and Language, 2011



Deep Neural Networks for STT

Context-Dependent Pre-trained Deep Neural Networks for Large Vocabulary Speech Recognition,

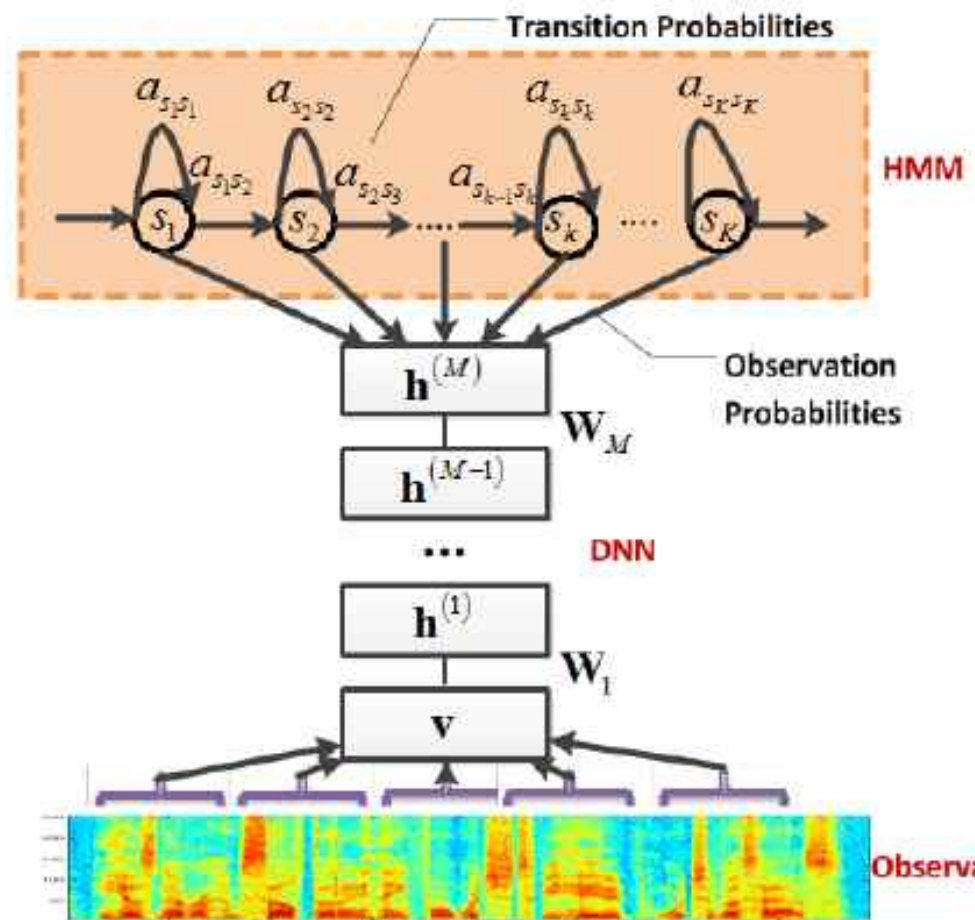
George E. Dahl, Dong Yu, Deng Li, and Alex Acero, IEEE Trans. on Audio, Speech and Language Processing, Jan, 2012

DNN as triphone state classifier

- input: acoustic features, e.g. MFCC
- output layer of DNN representing triphone states
- fine tuning the DNN by back propagation using labelled data

Hybrid System

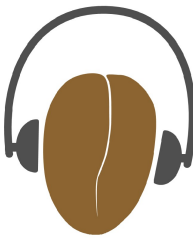
- normalized output of DNN as posterior of states $p(s|x)$
- state transition remaining unchanged, modeled by transition probabilities of HMM



DNN Acoustic Models for the Masses

- Nontrivial to get the DNN models to work well
 - **Design decisions**: # layers, # nodes, # outputs, type of nonlinearity, training criterion
 - **Training art**: learning rates, regularization, update stability (max change), data randomization, # epochs
 - **Computational art**: matrix libraries, memory mgmt
- Kaldi recipes provide a robust starting point

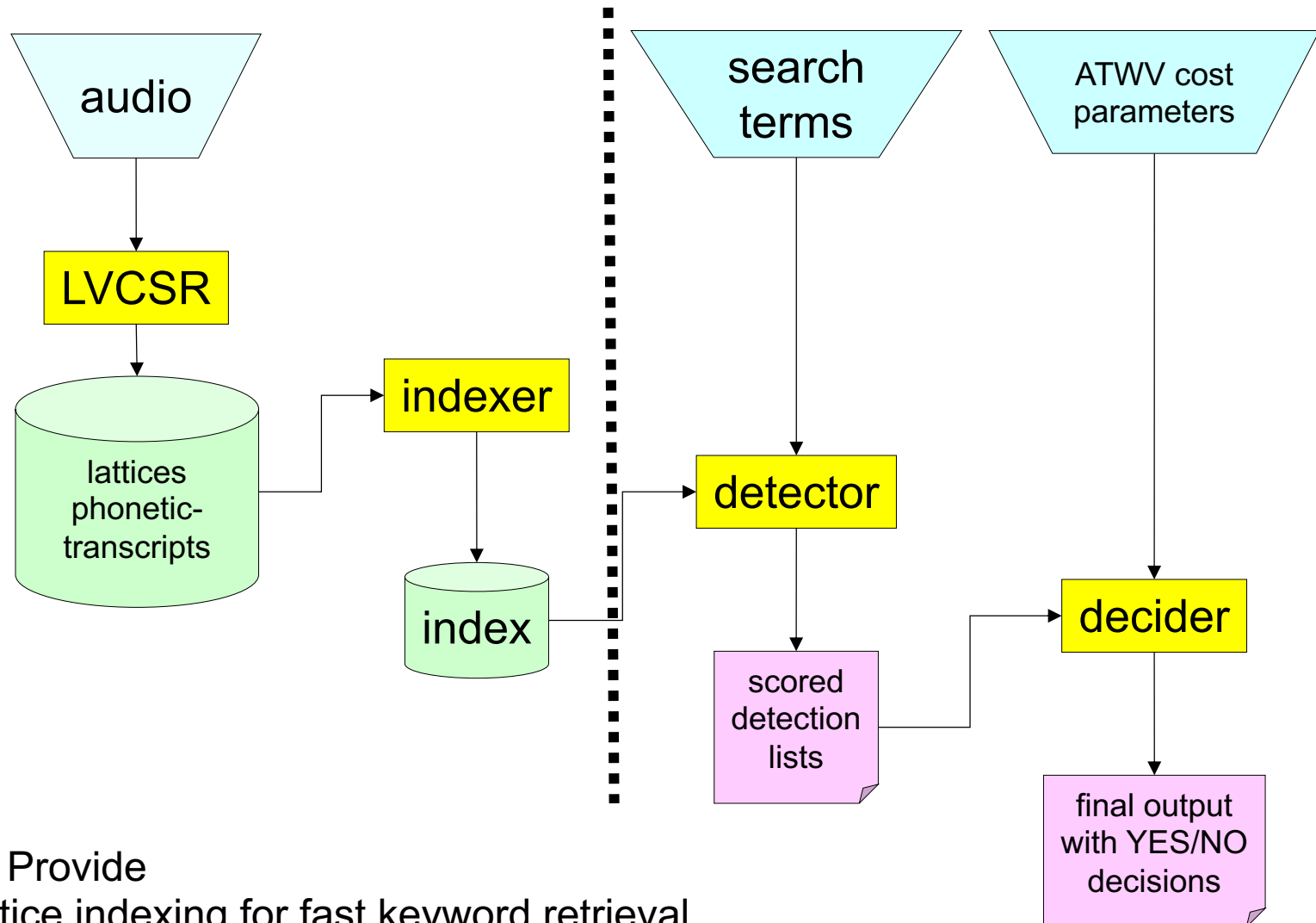
Corpus	Training Speech	SGMM WER	DNN WER
BABEL Pashto	10 hours	69.2%	67.6%
BABEL Pashto	80 hours	50.2%	42.3%
Fisher English	2000 hours	15.4%	10.3%



Staying Ahead in the STT Game

- STT technology is advancing very rapidly
 - Amazon, Apple, Baidu, Facebook, Google, Microsoft
- Kaldi leads and keeps up with major innovations
 - From SGMMs to DNN (2012)
 - **From “English” to low-resource languages (2013)**
 - From CPUs to GPUs (2014)
 - From close-talking to far-field microphones (2015)
 - New chain models (2017)

Keyword Search (KWS) System



Kaldi Provide

- lattice indexing for fast keyword retrieval.
- Proxy keywords to handle out-of-vocabulary (OOV) problem.

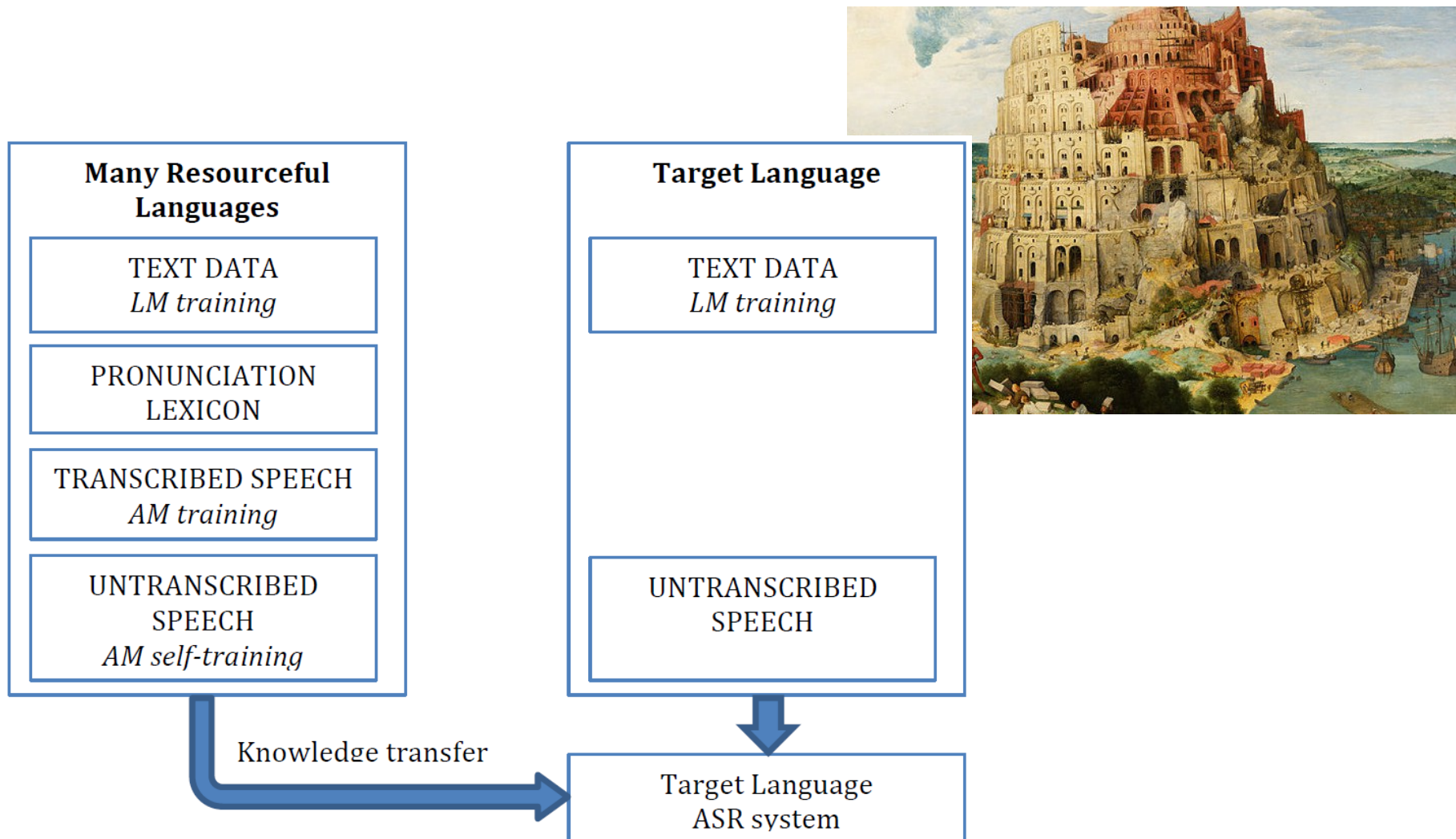
Low-Resource STT for the Masses

- Kaldi provides language-independent recipes
 - Typical BABEL Full language pair (LP) condition
 - 80 hours of transcribed speech, 800K words of LM text, 20K word pronunciation lexicon
 - Typical BABEL Limited LP condition
 - 10 hours of transcribed speech, 100K words of LM text, 6K word pronunciation lexicon

Language	Cantonese		Tagalog		Pashto		Turkish	
Speech	80h	10h	80h	10h	80h	10h	80h	10h
CER/WER	48.5%	61.2%	46.3%	61.9%	50.7%	63.0%	51.3%	65.3%
ATWV	0.47	0.26	0.56	0.28	0.46	0.25	0.52	0.25

- Babel program: “to rapidly develop speech recognition capability with limited amounts of transcription.”
- “actual term weighted value” (ATWV), $TWV(kw) \equiv 1 - P_{Miss}(kw) - \beta * P_{FA}(kw)$

Building Speech Recognition System from Untranscribed Data





Staying Ahead in the STT Game

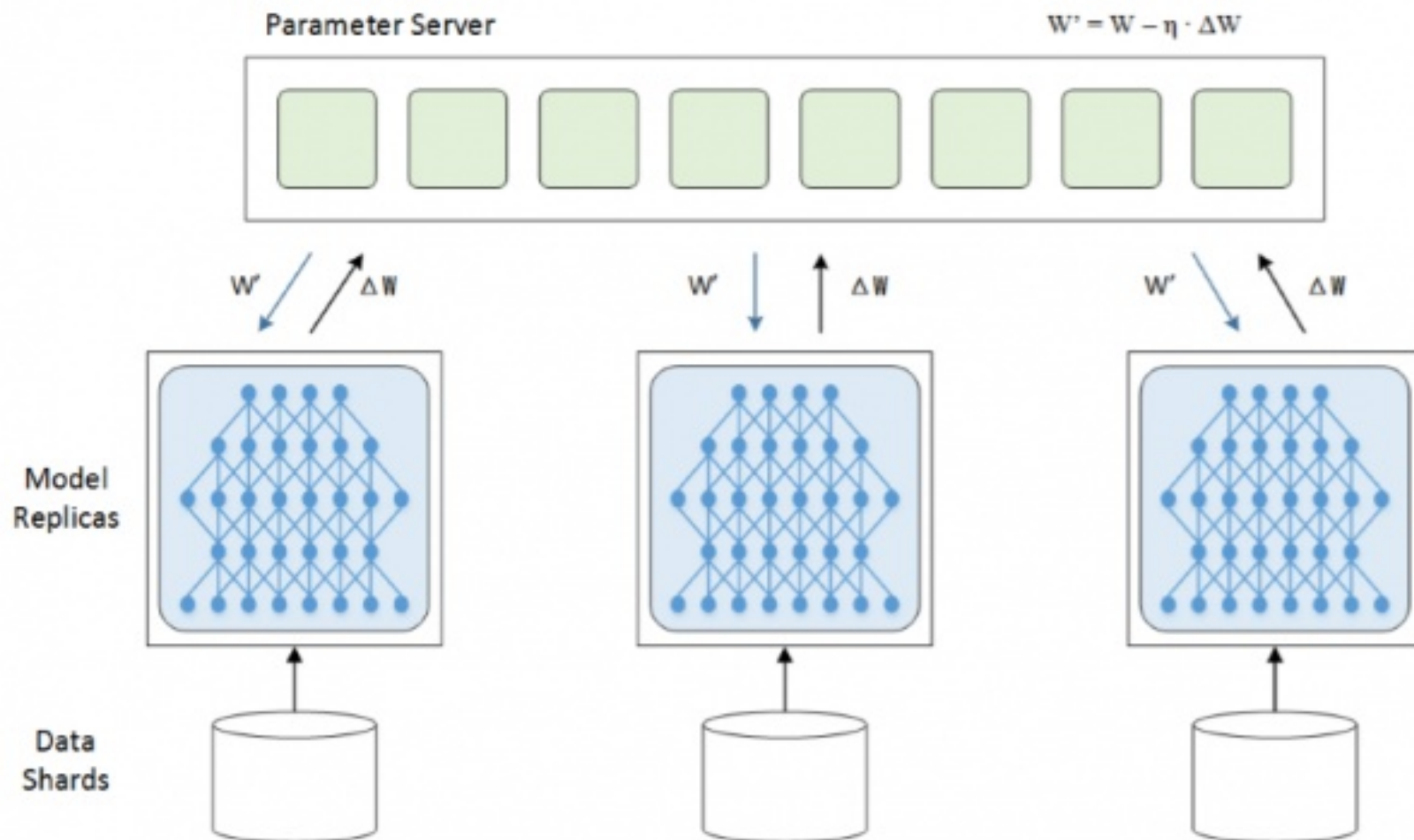
- STT technology is advancing very rapidly
 - Amazon, Apple, Baidu, Facebook, Google, Microsoft
- Kaldi leads and keeps up with major innovations
 - From SGMMs to DNN (2012)
 - From “English” to low-resource languages (2013)
 - **From CPUs to GPUs (2014)**
 - From close-talking to far-field microphones (2015)
 - Chain models for better STT, faster decoding (2017)

Parallel (GPU-based) Training

- Original neural network training algorithms were inherently sequential (e.g. SGD)
- Scaling up to “big data” becomes a challenge
- Several solutions have emerged recently
 - 2009: Delayed SGD (Yahoo!)
 - 2011: Lock-free SGD (Hogwild! U Wisconsin)
 - 2012: Gradient averaging (DistBelief, Google)
 - 2014: Model averaging (NG-SGD, Kaldi)

PARALLEL TRAINING OF DNNs WITH NATURAL GRADIENT AND PARAMETER AVERAGING

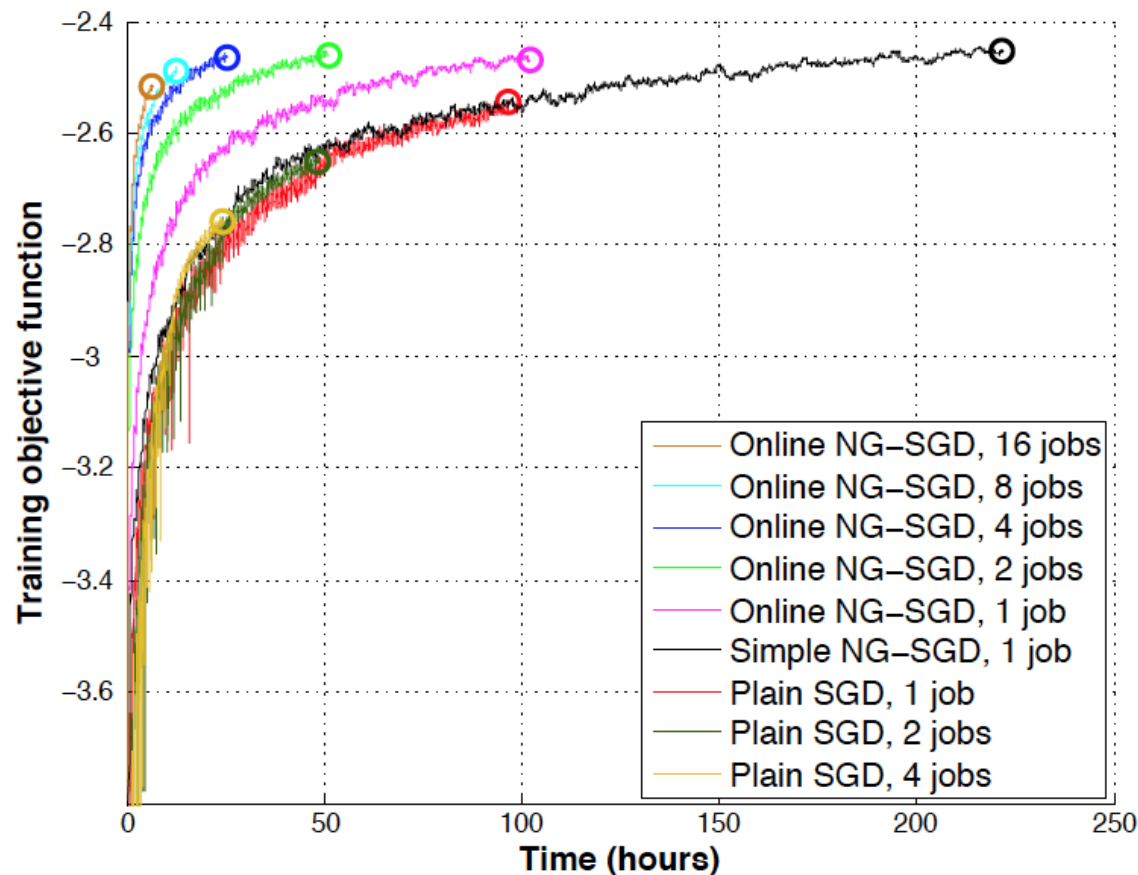
Daniel Povey, Xiaohui Zhang & Sanjeev Khudanpur, ICLR 2015



Model Averaging with NG-SGD

- Train DNNs with large amount of data
 - Utilize a cluster of CPUs or GPUs
 - Minimize network traffic (esp. for CPUs)
- Solution: exploit data parallelization
 - Update model in parallel over many mini-batches
 - Infrequently average models (parameters)
- Use “Natural-Gradient” SGD for model updating
 - Approximates conditioning via inverse Fisher matrix
 - Improves convergence even without parallelization

Parallelization Matters!



- Typically, a GPU is 10x faster than a 16 core CPU
- Linear speed-up till 4 GPUs, then diminishing



Staying Ahead in the STT Game

- STT technology is advancing very rapidly
 - Amazon, Apple, Baidu, Facebook, Google, Microsoft
- Kaldi leads and keeps up with major innovations
 - From SGMMs to DNN (2012)
 - From “English” to low-resource languages (2013)
 - From CPUs to GPUs (2014)
 - **From close-talking to far-field microphones (2015)**
 - Chain models for better STT, faster decoding (2017)

IARPA's Automatic Speech recognition In Reverberant Environments Challenge

Automatic speech recognition that works in a variety of acoustic environments and recording scenarios is a holy grail of the speech research community

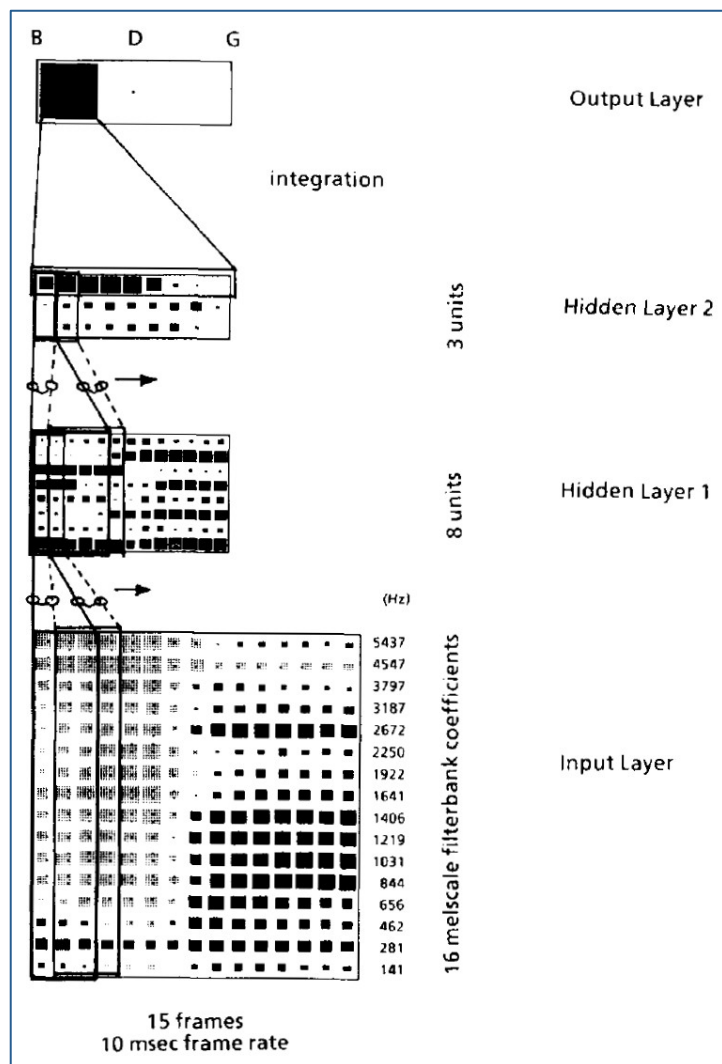


Voice Enabled Smart-Home

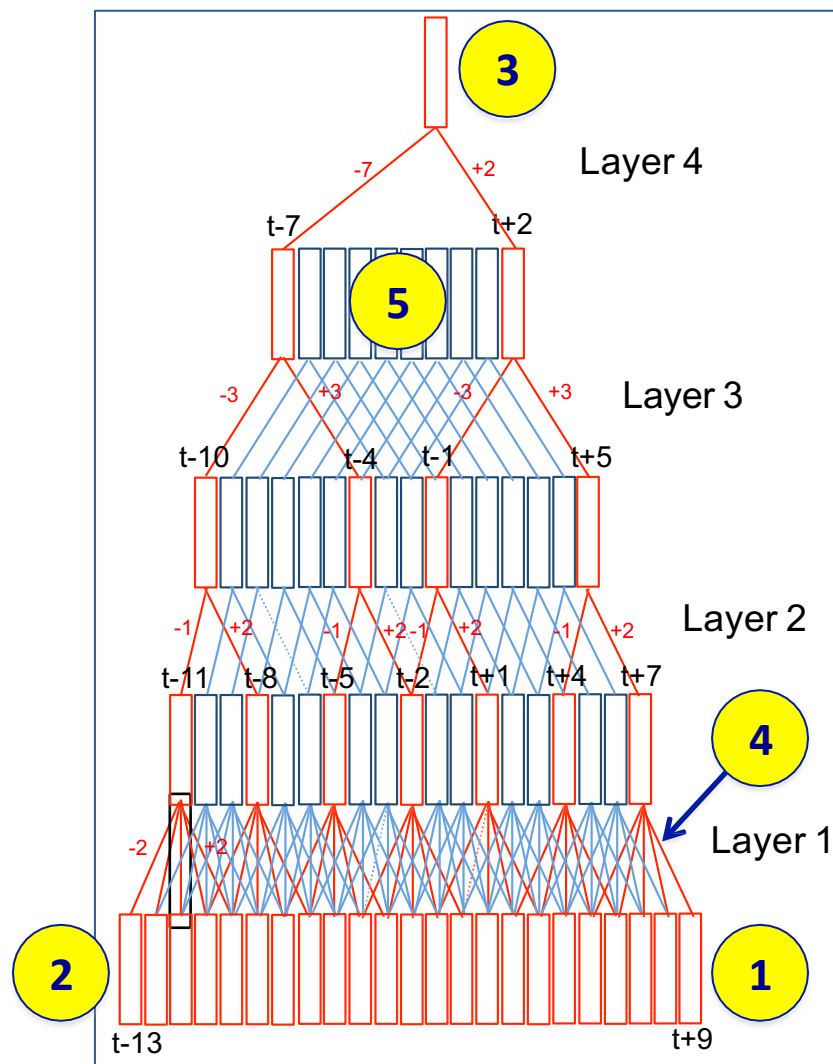
Kaldi ASR Improvements for ASplRE

- **Time delay neural networks (TDNN)**
 - A way to deal with **long acoustic-phonetic context**
 - A structured alternative to deep/recurrent neural nets
- **Data augmentation** with simulated reverberations
 - A way to mitigate **channel distortions not seen in training**
 - A form of multi-condition training of ASR models
- **i-vector based speaker & environment adaptation**
 - A way to deal with **speaker & channel variability**
 - Adapted [with a twist] from Speaker ID systems
- **Recurrent neural network language models (RNNLM)**
 - A (known) way to model **long-range word dependencies**
 - Incorporated post-submission into JHU ASplRE system

Time Delay Neural Networks



Alex Waibel, Kevin Lang, et al (1987)




Our TDNN Architecture (2015)

Improved ASR on Several Data Sets

Standard ASR Test Sets	Size	DNN	TDNN	Rel. Δ
Wall Street Journal	80 hrs	6.6%	6.2%	5%
TED-LIUM	118 hrs	19.3%	17.9%	7%
Switchboard	300 hrs	15.5%	14.0%	10%
Libri Speech	960 hrs	5.2%	4.8%	7%
Fisher English	1800 hrs	22.2%	21.0%	5%

- Consistent 5-10% reduction in word error rate (**WER**) over DNNs on most datasets, including conversational speech.
- TDNN training speeds are on par with DNN, and nearly an order of magnitude faster than RNN

ASpIRE (Fisher Training)	1800 hrs	47.7%	47.6%	
--------------------------	----------	-------	-------	---

Data Augmentation for ASR Training

- Simulate noisy data from clean speech
- As in Image recognition Task

✓ Original



✓ Simulated



Simulating Reverberant Speech for Multi-condition (T)DNN Training

- Simulate ca 5500 hours of reverberant, noisy data from 1800 hours of the Fisher English CTS corpus
 - Replicate each of the ca 21,000 conversation sides 3 times
 - Randomly change the [sampling rate](#) [up to $\pm 10\%$]
 - Convolve each conversation side with one of 320 real-life [room impulse responses](#) (RIR) chosen at random
 - Add [noise](#) to the signal (when available with the RIR)
- Generate (T)DNN training labels from clean speech
 - Align “pre-reverb” speech to ca 7500 CD-HMM states
- Train DNN and TDNN acoustic models
 - [Cross-entropy](#) training followed by [sequence training](#)

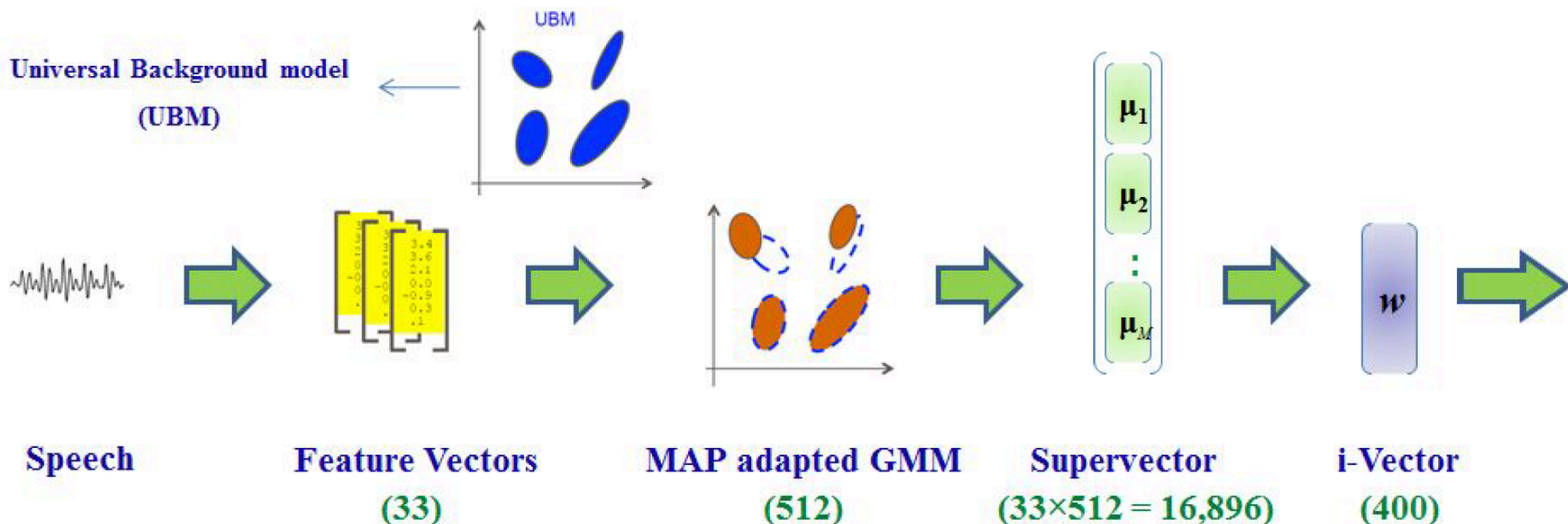
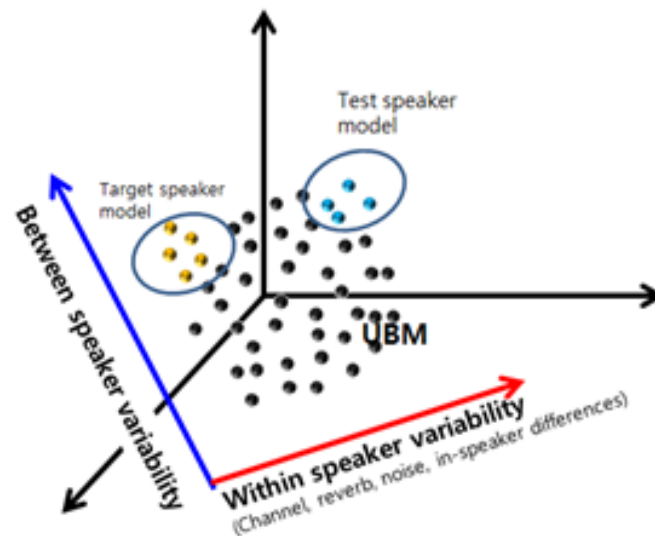
Result of Data Augmentation

Acoustic Model	Data Augmentation	Dev WER
TDNN A (230 ms)	None (1800h, clean speech)	47.6%
TDNN A (230 ms)	+ 3 x (reverberation + noise)	31.7%
TDNN B (290 ms)	+ 3 x (reverberation + noise)	30.8%
TDNN A (230 ms)	+ sampling rate perturbation	31.0%
TDNN B (290 ms)	+ sampling rate perturbation	31.1%

- Data augmentation with simulated reverberation is beneficial
 - Likely to be a very important reason for relatively good performance
- Sampling rate perturbation didn't help much on ASPIRE data
- Sequence training helped reduce WER on the dev set
 - Required modifying the sMBR training criterion to realize gains
 - But the **gains did not carry over** to dev-test set

i-vectors for Speaker Compensation

- **Problem: mismatch speaker, channel and session conditions**
- **i-vector constructs a low-dimensional subspace that could be used to compensate those variabilities**



Using i-vectors Instead of fMLLR

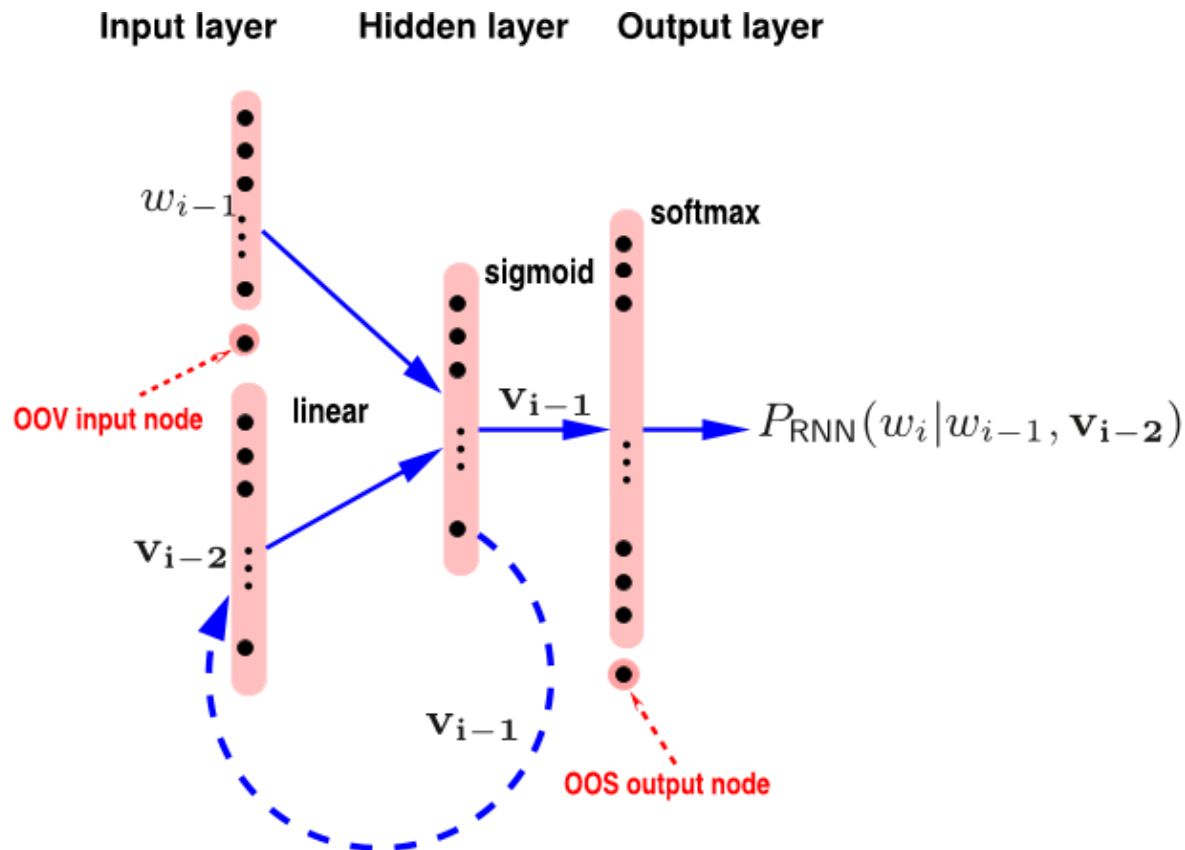
and using unnormalized MFCCs to compute i-vectors

- 100-dim i-vectors are appended to MFCC inputs of the TDNN
 - i-vectors are computed from raw MFCCs (i.e. no mean subtraction etc)
 - UBM posteriors however use MFCCs normalized over a 6 sec window
- i-vectors are computed for each training utterance
 - Increases speaker- and channel variability seen in training data
 - May model transient distortions? e.g. moving speakers, passing cars
- i-vectors are calculated for every ca 60 sec of test audio
 - UBM prior is weighted 10:1 to prevent overcompensation
 - Weight of test statistics is capped at 75:1 relative to UBM statistics

Speaker Compensation Method	Dev WER
TDNN without i-vectors	34.8%
+ i-vectors (from all frames)	33.8%
+ i-vectors (from reliable speech frames)	30.8%

Recurrent Neural Network based Language Models

- Significant outperformance in both perplexity and word error rate over standard n-gram



RNN LM on ASpIRE Data

Language Model and Rescoring Method	Dev WER
4-gram LM and lattice rescoring	30.8%
RNN-LM and 100-best rescoring	30.2%
RNN-LM and 1000-best rescoring	29.9%
RNN-LM (4-gram approximation) lattice rescoring	29.9%
RNN-LM (6-gram approximation) lattice rescoring	29.8%

- An RNN LM consistently outperforms the N-gram LM
- The Kaldi lattice rescoring appears to cause no loss in performance
 - Approximation entails not “expanding” the lattice to represent each unique history separately
 - When two paths merge in an N-gram lattice, only one $s(t)$ is chosen at random and propagated forward

Performance on Evaluation Data

Participant	Test WER	System Type
Kaldi	44.3%	Single System
BBN (and others)	44.3%	Combination
I ² R (Singapore)	44.8%	Combination

Acoustic Model	Language Model	Dev WER	Test WER	Eval WER
TDNN B (CE training)	4-gram	30.8%	27.7%	44.3%
TDNN B (sMBR training)	4-gram	29.1%	28.9%	43.9%
TDNN B (CE training)	RNN	29.8%	26.5%	43.4%
TDNN B (sMBR training)	RNN	28.3%	28.2%	43.4%



Staying Ahead in the STT Game

- STT technology is advancing very rapidly
 - Amazon, Apple, Baidu, Facebook, Google, Microsoft
- Kaldi leads and keeps up with major innovations
 - From SGMMs to DNN (2012)
 - From “English” to low-resource languages (2013)
 - From CPUs to GPUs (2014)
 - From close-talking to far-field microphones (2015)
 - Chain models for better STT, faster decoding (2017)

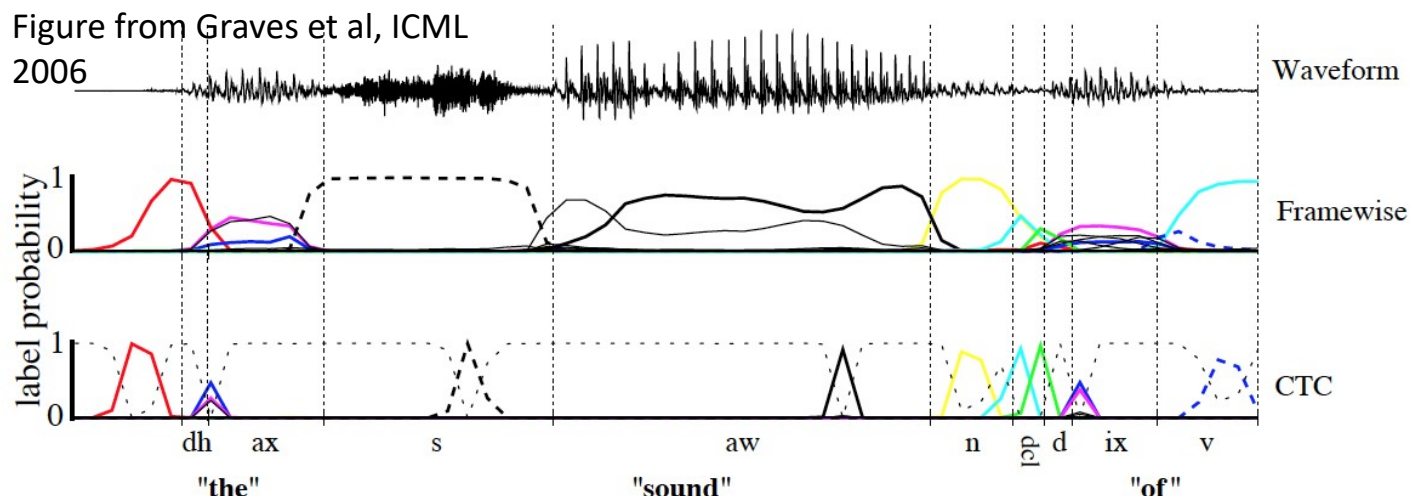
Chain Models

- A new class of acoustic models for hybrid STT
 - “1-state” HMM for context-dependent phones
 - LSTM-RNN acoustic models (TDNN also compatible)
- A new lattice-free MMI training method
 - Better suited to using GPUs for parallelization
 - Does not require CE training before MMI training
- Improved speed and STT performance
 - 6%-8% relative WER reduction over previous best
 - 5-10x improvement in training time; 3x decoding time

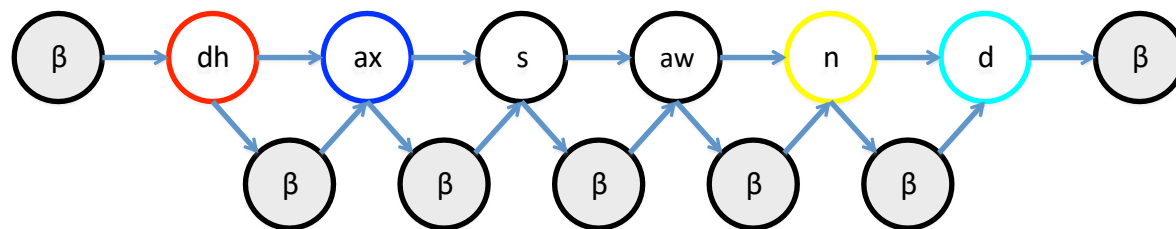
Chain Models vs. Connectionist Temporal Classification

- CTC

- ✓ NN output



- ✓ Boundary search path



- Chain Models

- ✓ $S_p \rightarrow$ CD phone (boundary)
- ✓ $S_b \rightarrow$ "Blank" phone (stable state)



STT Results for Chain Models

300 hours of SWBD Training Speech; Hub-5 '00 Evaluation Set

Training Objective	Model (Size)	Total WER	SWBD WER
Cross-Entropy	TDNN A (16.6M)	18.2%	12.5%
CE + sMBR	TDNN A (16.6M)	16.9%	11.4%
Lance-free MMI	TDNN A (9.8M)	16.1%	10.7%
	TDNN B (9.9M)	15.6%	10.4%
	TDNN C (11.2M)	15.5%	10.2%
LF-MMI + sMBR	TDNN C (11.2M)	15.1%	10.0%

- LF-MMI reduces WER by ca 10%-15% *relative*
- LF-MMI is better than standard CE + sMBR training (ca 8%)
- LF-MMI improves very slightly with additional sMBR training

Chain Models and LF-MMI Training

STT Performance on a Variety of Corpora

Corpus and Audio Type	Training Speech	CE + sMBR Error Rate	LF-MMI Error Rate
AMI IHM	80 hours	23.8%	22.4%
AMI SDM	80 hours	48.9%	46.1%
TED-LIUM	118 hours	11.3%	12.8%
Switchboard	300 hours	16.9%	15.5%
Fisher + SWBD	2100 hours	15.0%	13.3%

- Chain models with LF-MMI reduce WER by 6%-11% (*relative*)
- LF-MMI improves a bit further with additional sMBR training
- FL-MMI is 5x-10x faster to train, 3x faster to decode



Staying Ahead in the STT Game

- STT technology is advancing very rapidly
 - Amazon, Apple, Baidu, Facebook, Google, Microsoft
- Kaldi leads and keeps up with major innovations
 - From SGMMs to DNN(2012)
 - From “English” to low-resource languages(2013)
 - From CPUs to GPUs(2014)
 - From close-talking to far-field microphones(2015)
 - Chain models for better STT, faster decoding (2017)
- and the list goes on ...