Speech Recognition

Andrew Senior
(DeepMind London)
Many thanks for slides to Vincent Vanhoucke, Heiga Zen, Jun Song & Andrew Zisserman
February 21st, 2017. Oxford University

Outline

Speech recognition

- Acoustic representation
- Phonetic representation
- History
- Probabilistic speech recognition

Neural network speech recognition

- Hybrid neural networks
- Training losses
- Sequence discriminative training
- New architectures

Other topics

Speech recognition problem

Automatic speech recognition (ASR)

 → "OK Google, directions home"

Text-to-speech synthesis (TTS)

"Take the first left" → 

Speech problems

- Automatic speech recognition
  - Spontaneous vs read speech
  - Large vocabulary
  - In noise
  - Low resource
  - Far-field
  - Accent-independent
  - Speaker-adaptive
- Text to speech
  - Low resource
  - Realistic prosody
- Speaker identification
- Speech enhancement
- Speech separation

Outline

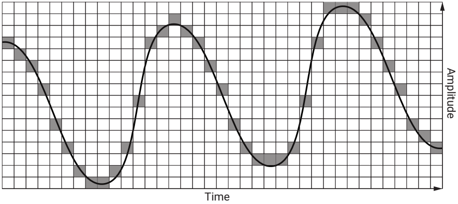
Speech recognition
- Acoustic representation
- Phonetic representation
- History
- Probabilistic speech recognition

Neural network speech recognition
- Hybrid neural networks
- Training losses
- Sequence discriminative training
- New architectures

Other topics

What is speech — physical realisation

- Waves of changing air pressure.
- Realised through excitation from the vocal cords
- Modulated by the vocal tract.
- Modulated by the articulators (tongue, teeth, lips).
- Vowels produced with an open vocal tract (stationary)
  - Can be parameterized by position of tongue.
- Consonants are constrictions of vocal tract.
- Converted to Voltage with a microphone.
- Sampled with an Analogue to Digital Converter



Sampling & Quantization

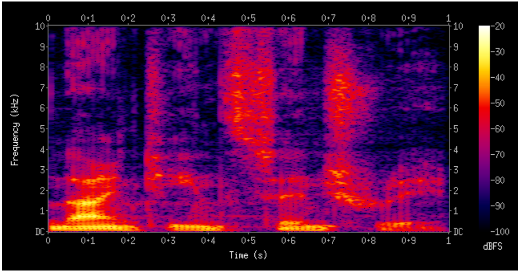
Speech representation

- Human hearing is ~50Hz-20kHz
- Human speech is ~85Hz–8kHz
- Telephone speech has 8kHz sampling: 300Hz–4kHz bandwidth
- 1 bit per sample can be intelligible
- CD is 44.1kHz 16 bits per sample
- Contemporary speech processing mostly around 16kHz 16bits/sample

Speech representation

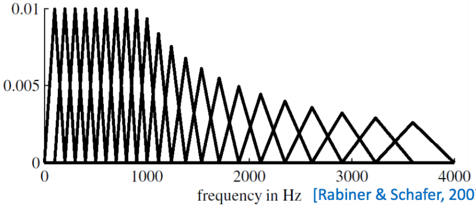
We want a low-dimensionality representation, invariant to speaker, background noise, rate of speaking etc.

- Fourier analysis shows energy in different frequency bands.
- windowed short-term fast Fourier transform
- e.g. FFT on overlapping 25ms windows (400 samples) taken every 10ms
  - Energy vs frequency [discrete] vs time [discrete]



Mel frequency representation

- FFT is still too high-dimensional.
- Downsample by local weighted averages on mel scale non-linear spacing, and take a log. $m = 1127 \ln(1 + \frac{f}{700})$
- Result in log-mel features (default for neural network speech modelling.)
- 40+ dimensional features per frame



[Rabiner & Schafer, 2007]

MFCCs

- Mel Frequency Cepstral Coefficients - MFCCs are the discrete cosine transformation of the mel filterbank energies. Whitened and low-dimensional.
- Similar to Principal Components of log spectra.
- GMM speech recognition systems may use 13 MFCCs
- Perceptual Linear Prediction – a common alternative representation.
- Frame stacking- it's common to concatenate several consecutive frames.
- e.g. 26 for fully-connected DNN. 8 for LSTM.
- GMMs used local differences (deltas) and second-order differences (delta-deltas) to capture dynamics. $(13 + 13 + 13$ dimensional$)$
- Ultimately use ~39 dimensional linear discriminant analysis (~class-aware PCA) projection of 9 stacked MFCC vectors.

Outline

Speech recognition

- Acoustic representation
- Phonetic representation
- History
- Probabilistic speech recognition

Neural network speech recognition

- Hybrid neural networks
- Training losses
- Sequence discriminative training
- New architectures

Other topics

Speech as communication

- Speech evolved as communication to convey information.
- Consists of sentences (in ASR we usually talk about "utterances")
- Sentences composed of words
- Minimal unit is a "phoneme"
  - Minimal unit that distinguishes one word from another.
  - Set of 40–60 distinct sounds.
  - Vary per language,
  - Universal representations.
    - IPA: international phonetic alphabet,
    - X-SAMPA (ASCII)
- Homophones
  - distinct words with the same pronunciation: "there" vs "their"
- Prosody
  - How something is said can convey meaning.

Datasets

- TIMIT
  - Hand-marked phone boundaries given
  - 630 speakers $\times$ 10 utterances
- Wall Street Journal (WSJ) 1986 Read speech. WSJ0 1991, 30k vocab
- Broadcast News (BN) 1996 104 hours
- Switchboard (SWB) 1992. 2000 hours spontaneous telephone speech 500 speakers
- Google voice search
  - anonymized live traffic 3M utterances 2000 hours hand-transcribed 4M vocabulary. Constantly refreshed, synthetic reverberation + additive noise
- DeepSpeech 5000h read (Lombard) speech + SWB with additive noise.
- YouTube 125,000 hours aligned captions (Soltau et al., 2016)

Outline

Speech recognition

- Acoustic representation
- Phonetic representation
- History
- Probabilistic speech recognition

Neural network speech recognition

- Hybrid neural networks
- Training losses
- Sequence discriminative training
- New architectures

Other topics

Rough History

- 1960s Dynamic Time Warping
- 1970s Hidden Markov Models
- Multi-layer perceptron 1986
- Speech recognition with neural networks 1987–1995
- Superseded by GMMs 1995–2009
- Neural network features 2002–
- Deep networks 2006– (Hinton, 2002)
- Deep networks for speech recognition
  - Good results on TIMIT (Mohamed et al., 2009)
  - Results on large vocabulary systems 2010 (Dahl et al., 2011)
  - Google launches DNN ASR product 2011
  - Dominant paradigm for ASR 2012 (Hinton et al., 2012)
- Recurrent networks for speech recognition 1990, 2012–
  - New models (attention, LAS, neural transducer)

Outline

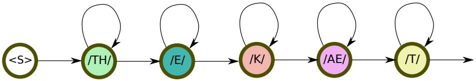
Speech recognition
- Acoustic representation
- Phonetic representation
- History
- Probabilistic speech recognition

Neural network speech recognition
- Hybrid neural networks
- Training losses
- Sequence discriminative training
- New architectures

Other topics

Probabilistic speech recognition

- Speech signal represented as an observation sequence $o = \{o_t\}$.
- We want to find the most likely word sequence $\hat{w}$
- We model this with a Hidden Markov Model.
  - The system has a set of discrete states,
  - transitions from state to state according to transition probabilities (Markovian: memoryless)
  - Acoustic observation when making a transition is conditioned on state alone. $P(o_t|c_t)$
  - We seek to recover the state sequence and consequently the word sequence.



Fundamental equation of speech recognition

We choose the decoder output as the most likely sequence $\hat{w}$ from all possible sequences, $\Sigma*$, for an observation sequence o:

$$\hat{w} = \arg\max_{w \in \Sigma*} P(w|o) \tag{1}$$

$$= \arg\max_{w \in \Sigma*} P(o|w)P(w) \tag{2}$$

A product of *Acoustic model* and *Language model* scores.

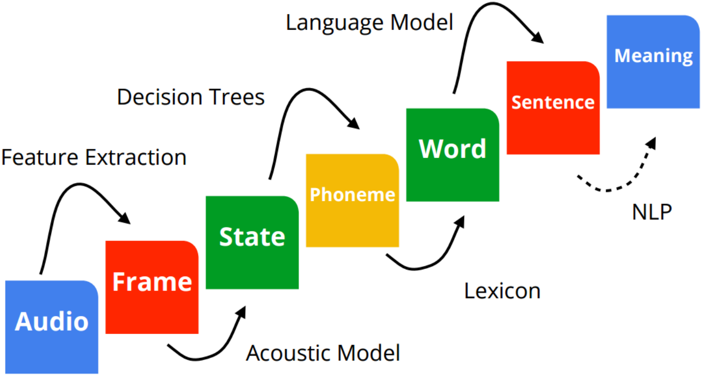
$$P(o|w) = \sum_{d,c,p} P(o|c)P(c|p)P(p|w) \tag{3}$$

Where p is the phone sequence and c is the state sequence.

- We can model word sequences with a language model.

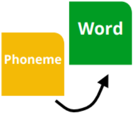
$$P(w_1, w_2, \ldots, w_N) = P(w_0) \prod P(w_i | w_0, \ldots, w_{i-1})$$

Speech recognition as transduction
From signal to language.



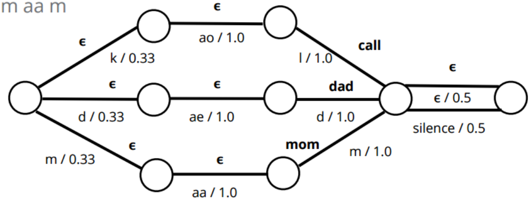
Speech recognition as transduction – lexicon

Construct graph using Weighted Finite State Transducers (WFST)

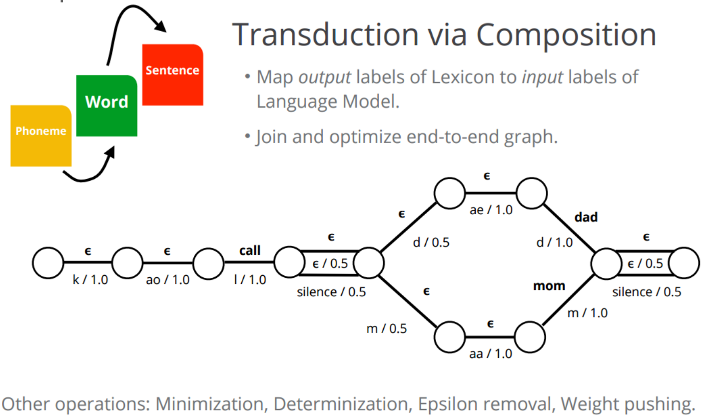


Lexicon

- call: k ao l
- dad: d ae d
- mom: m aa m



Compose Lexicon FST with Grammar FST $L \circ G$



Phonetic units

- Phonemes: "cat" $\rightarrow$ /K/, /AE/, /T/
- Context independent HMM states $k_1, k_2, ae_1 \ldots$
  - Model onset / middle / end separately.
- Context dependent states $k_{1.17}, \ldots$
- Context dependent phones
- Diphones (pairs of half-phones)
- Syllables
- Word-parts cf Machine translation (Wu et al., 2016)
- Characters (graphemes)
- Whole words Sak et al. (2014a, 2015); Soltau et al. (2016)
  - Hard to generalize to rare words.

Choice depends on language, size of dataset, task, resources available.

Context dependent phonetic clustering

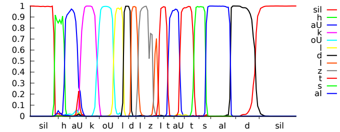
- A phone's realization depends on the preceding and following context
- Could improve discrimination if we model different contextual realizations separately:
  e.g AE preceded by K, followed by T: AE+T-K
- But, if we have 42 phones, and 3 states per phone, there are 3×42^3 context-dependent phones.
- Most of these won't be observed
- So cluster – group together similar distributions and train a joint model.
- Have a "back-off" rule to determine which model to use for unobserved contexts.
- Usually a decision tree.

Gaussian Mixture Models

- Dominant paradigm for ASR from 1990 to 2010
- Model the probability distribution of the acoustic features for each state.
  $P(o_t|c_i) = \sum_j w_{ij} N(o_t; \mu_{ij}, \sigma_{ij})$
- Often use diagonal covariance Gaussians to keep number of parameters under control.
- Train by the E-M algorithm (Dempster et al., 1977) alternating:
  - M: forced alignment computing the maximum-likelihood state sequence for each utterance
  - E: parameter (μ, σ) estimation
- Complex training procedures to incrementally fit increasing numbers of components per mixture.
  - More components, better fit. 79 parameters / component.
- Given an alignment mapping audio frames to states, this is parallelizable by state.
- Hard to share parameters / data across states.

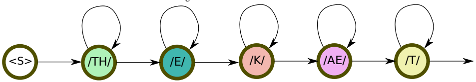
Forced alignment

- Forced alignment uses a model to compute the maximum likelihood alignment between speech features and phonetic states.
- For each training utterance, construct the set of phonetic states for the ground truth transcription.
- Use Viterbi algorithm to find ML monotonic state sequence
- Under constraints such as at least one frame per state.
- Results in a phonetic label for each frame.
- Can give hard or soft segmentation.



Forced alignment

With a transducer with states c_i:



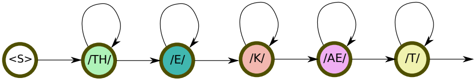
Compute state likelihoods at time t

$$P(o_{1,\dots,t}|c_i) = \sum_j P(o_t|c_j)P(o_{1,\dots,t}|c_j)P(c_i|c_j)$$

With transition probabilities: $P(c_i|c_j)$
To find best path;

$$P(o_{1,\dots,t}|c_i) = \max_j P(o_t|c_j)P(o_{1,\dots,t}|c_j)P(c_i|c_j)$$

Forced alignment $t = 0$



Observation likelihoods $P(o_t|c_i)$

...															
...															
...															
/t/	0.1	0.1	0.1	0.1	0.2	0.1									
/ae/	0.1	0.1	0.3	0.3	0.1	0.4									
/k/	0.1	0.1	0.1	0.2	0.5	0.1									
/e/	0.1	0.2	0.3	0.2	0.1	0.3									
/th/	0.6	0.5	0.1	0.1	0.2	0.1									

t->

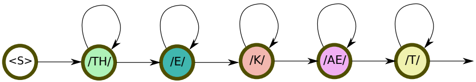
Start distribution $P_{t=0}(c_i)$

...

...															
...															
/t/															
/ae/	0														
/k/	0														
/e/	0														
/th/	0														
<s>	1.0														
	0														

t->

Forced alignment $t = 1$



Observation likelihoods $P(o_t|c_i)$

...															
...															
...															
/t/	0.1	0.1	0.1	0.1	0.2	0.1									
/ae/	0.1	0.1	0.3	0.3	0.1	0.4									
/k/	0.1	0.1	0.1	0.2	0.5	0.1									
/e/	0.1	0.2	0.3	0.2	0.1	0.3									
/th/	0.6	0.5	0.1	0.1	0.2	0.1									

t->

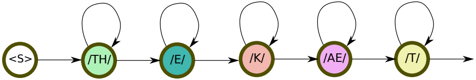
State likelihoods $P(o_{1,\ldots,t}|c_i)$

...

...															
...															
/t/															
/ae/	0														
/k/	0														
/e/	0														
/th/	0	0.6													
<s>	1.0														
	0	1													

t->

Forced alignment $t = 1$



Observation likelihoods $P(o_t|c_i)$

...						
...						
...						
/t/	0.1	0.1	0.1	0.1	0.2	0.1
/ae/	0.1	0.1	0.3	0.3	0.1	0.4
/k/	0.1	0.1	0.1	0.2	0.5	0.1
/e/	0.1	0.2	0.3	0.2	0.1	0.3
/th/	0.6	0.5	0.1	0.1	0.2	0.1

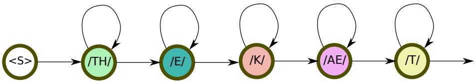
t->

State likelihoods $P(o_{1,\ldots,t}|c_i)$

...			
...			
...			
/t/			
/ae/	0		
/k/	0		
/e/	0		.03
/th/	0	0.6	.15
<s>	1.0		
	0	1	2

t->

Forced alignment $t = T$

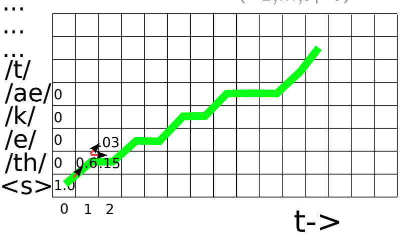


Observation likelihoods $P(o_t|c_i)$

...															
...															
...															
/t/	0.1	0.1	0.1	0.1	0.2	0.1									
/ae/	0.1	0.1	0.3	0.3	0.1	0.4									
/k/	0.1	0.1	0.1	0.2	0.5	0.1									
/e/	0.1	0.2	0.3	0.2	0.1	0.3									
/th/	0.6	0.5	0.1	0.1	0.2	0.1									

t->

State likelihoods $P(o_{1,\ldots,t}|c_i)$



Decoding

Speech recognition unfolds in much the same way.

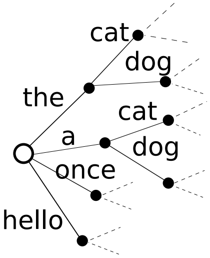
Now we have a graph instead of a straight-through path.

Optional silences between words

Alternative pronunciation paths.

Typically use max probability, and work in the $\log$ domain.

Hypothesis space is huge, so we only keep a "beam" of the best paths, and can lose what would end up being the true best path.

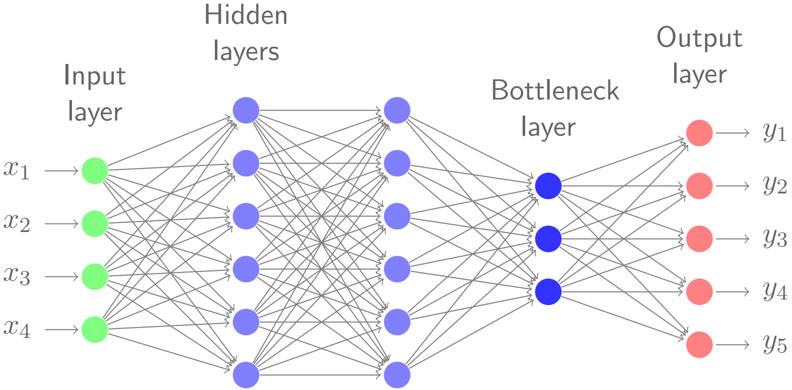


Two main paradigms for neural networks for speech

- Use neural networks to compute nonlinear feature representations.
  - "Bottleneck" or "tandem" features (Hermansky et al., 2000)
  - Low-dimensional representation is modelled conventionally with GMMs.
  - Allows all the GMM machinery and tricks to be exploited.
- Use neural networks to estimate phonetic unit probabilities.

Neural network features

Train a neural network to discriminate classes.
Use output or a low-dimensional *bottleneck* layer representation as features.



Neural network features

- TRAP: Concatenate PLP-HLDA features and NN features.
- Bottleneck outperforms posterior features (Grezl et al., 2007)
- Generally DNN features + GMMs reach about the same performance as hybrid DNN-HMM systems, but are much more complex.

Outline

Speech recognition

- Acoustic representation
- Phonetic representation
- History
- Probabilistic speech recognition

Neural network speech recognition

- Hybrid neural networks
- Training losses
- Sequence discriminative training
- New architectures

Other topics

Hybrid networks

- Train the network as a classifier with a softmax across the phonetic units.
- Train with cross-entropy.
- Softmax

$$y(i) = \frac{\exp(a(i,\theta))}{\sum_{j=1}^{N} \exp(a(j,\theta))}$$

will converge to posterior across phonetic states: $P(c_i|o_t)$

Hybrid Neural network decoding

Now we model $P(o|c)$ with a Neural network instead of a Gaussian Mixture model. Everything else stays the same.

$$P(o|c) = \prod_t P(o_t|c_t) \tag{4}$$

$$P(o_t|c_t) = \frac{P(c_t|o_t)P(o_t)}{P(c_t)} \tag{5}$$

$$\propto \frac{P(c_t|o_t)}{P(c_t)} \tag{6}$$

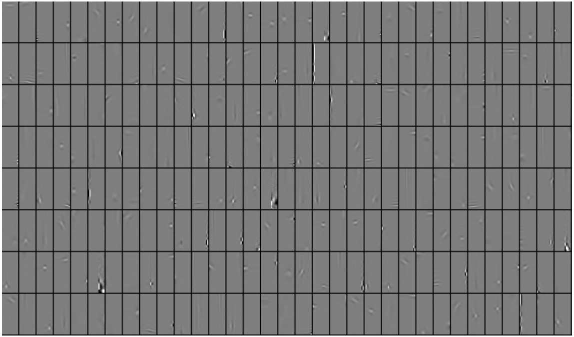
For observations o_t at time t and a CD state sequence c_t.
We can ignore $P(o_t)$ since it is the same for all decoding paths.
The last term is called the "scaled posterior":

$$\log P(o_t|c_t) = \log P(c_t|o_t) - \alpha \log P(c_t) \tag{7}$$

Empirically (by cross validation) we actually find better results with a "prior smoothing" term $\alpha \approx 0.8$.

Input features

Neural networks can handle high-dimensional features with correlated features.
Use (26) stacked filterbank inputs. (40-dimensional mel-spaced filterbanks)
Example filters learned in the first layer of a fully-connected network:



(33 x 8 filters. Each subimage 40 frequency vs 26 time.)

Neural network architectures for speech recognition

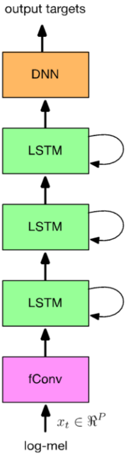
- Fully connected
- Convolutional networks (CNNs)
- Recurrent neural networks (RNNs)
  - LSTMs
  - GRUs

Convolutional neural networks

- Time delay neural networks
  - Waibel et al. (1989)
  - Dilated convolutions (Peddinti et al., 2015)
- CNNs in time or frequency domain. Abdel-Hamid et al. (2014); Sainath et al. (2013)
- Wavenet (van den Oord et al., 2016)

Recurrent neural networks

- RNNs
  - RNN (Robinson and Fallside, 1991)
  - LSTM Graves et al. (2013)
  - Deep LSTM-P Sak et al. (2014b)
  - CLDNN (right) (Sainath et al., 2015a)
  - GRU. DeepSpeech 1/2 (Amodei et al., 2015)
- Bidirectional (Schuster and Paliwal, 1997) helps, but introduces latency.
- Dependencies not long at speech frame rates (100Hz).
- Frame stacking and down-sampling help.



Human parity in speech recognition (Xiong et al., 2016)

- Ensemble of BLSTMs
- i-vectors for speaker normalization
  - i-vector is an embedding of audio trained to discriminate between speakers. (Speaker ID)
- Interpolated n-gram + LSTM language model.
- 5.8% WER on SWB (vs 5.9% for human).

Outline

Speech recognition

- Acoustic representation
- Phonetic representation
- History
- Probabilistic speech recognition

Neural network speech recognition

- Hybrid neural networks
- Training losses
- Sequence discriminative training
- New architectures

Other topics

Cross Entropy Training

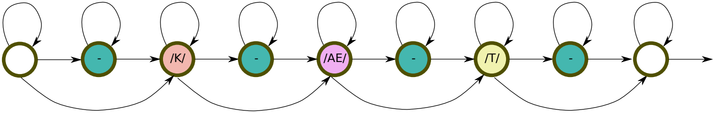
- GMMs were trained with *Maximum Likelihood*
- Conventional training uses *Cross-Entropy* loss.

$$\mathcal{L}_{XENT}\left(o_t, \theta\right) = \sum_{i=1}^{N} y_t\left(i\right) \log \frac{y_t\left(i\right)}{\hat{y}_t\left(i\right)}$$

- With large data we can use Viterbi (binary) targets: $y_t \in \{0, 1\}$
  - i.e. a *hard* alignment.
- Can also use a soft (Baum-Welch) alignment (Senior and Robinson, 1994)

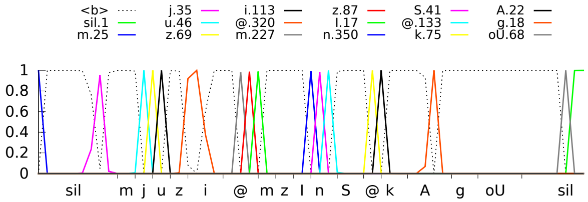
Connectionist Temporal Classification (Graves et al., 2006)

- CTC is a bundle of alternatives to conventional system:
  - CTC introduces an optional blank symbol between the "real" labels.
  - Simple to implement in the FST framework -an optional



  - Continuous realignment — no need for a bootstrap model
  - Always use soft targets.
  - Don't scale by posterior.
- Similar results to conventional training.

CTC alignments



Outline

Speech recognition

- Acoustic representation
- Phonetic representation
- History
- Probabilistic speech recognition

Neural network speech recognition

- Hybrid neural networks
- Training losses
- Sequence discriminative training
- New architectures

Other topics

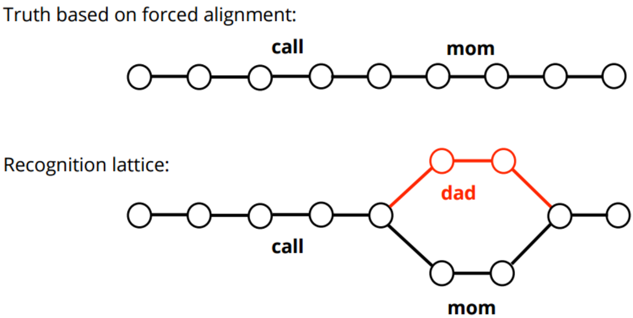
Sequence discriminative training

- Conventional training uses *Cross-Entropy* loss
  - Tries to maximize probability of the true state sequence given the data.
- We care about Word Error Rate of the complete system.
- Design a loss that's differentiable and closer to what we care about.
- Applied to neural networks (Kingsbury, 2009)
- Posterior scaling gets learnt by the network.
- Improves conventional training *and CTC* by ~15% relative.
- bMMI, sMBR(Povey et al., 2008)

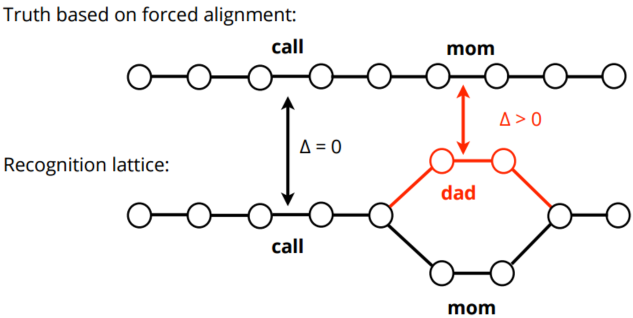
$$P\left(S_r|X_r\right) = \frac{p\left(\mathbf{X}_r, S_r\right)}{\sum_S p\left(\mathbf{X}_r, S\right)} = \frac{p\left(\mathbf{X}_r|S_r\right) P\left(S_r\right)}{\sum_S p\left(\mathbf{X}_r|S\right) P\left(S\right)}$$

$$\mathcal{L}_{mmi}\left(\theta\right) = -\sum_{r=1}^{R} \log P\left(S_r|\mathbf{X}_r\right)$$

Sequence discriminative training



Sequence discriminative training



Outline

Speech recognition
- Acoustic representation
- Phonetic representation
- History
- Probabilistic speech recognition

Neural network speech recognition
- Hybrid neural networks
- Training losses
- Sequence discriminative training
- New architectures

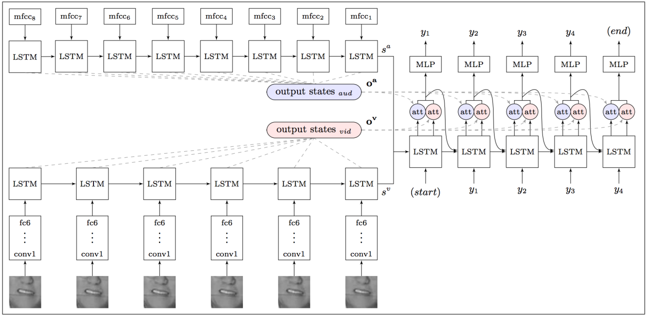
Other topics

Sequence2Sequence

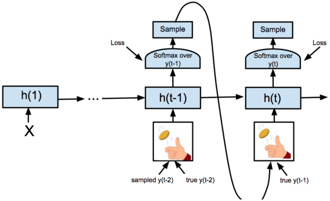
- Basic sequence2sequence not that good for speech
  - Utterances are too long to memorize
  - Monotonicity of audio (vs Machine Translation)
- Attention + seq2seq for speech (Chorowski et al., 2015)
- Listen, Attend and Spell (Chan et al., 2015)
- Output characters until EOS
- Incorporates language model of training set.
- Harder to incorporate a separately-trained language model. (e.g. trained on trillions of tokens)

Watch Listen, Attend and Spell (Chung et al., 2016)

Apply LAS to audio and video streams simultaneously.



Train with scheduled sampling (Bengio et al., 2015)



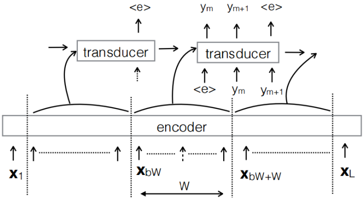
Watch Listen, Attend and Spell (Chung et al., 2016)

Method	SNR	CER	WER
Lips only			
Professional	-	58.7%	73.8%
WAS	-	42.1%	53.2%
Audio only			
LAS	clean	16.2%	26.9%
LAS	0dB	59.0%	74.5%
Audio and lips			
WLAS	clean	13.3%	22.8%
WLAS	0dB	35.8%	50.8%

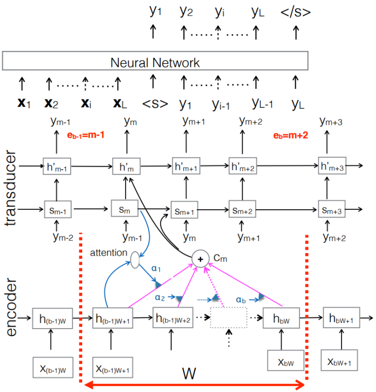
Methods	LRW [9]	GRID [11]
Lan *et al.* [23]	-	35.0%
Wand *et al.* [39]	-	20.4%
Chung and Zisserman [9]	38.9%	-
WAS (ours)	**15.5%**	**3.3%**

Neural transducer (Jaitly et al., 2015)

- Seq2seq models require the whole sequence to be available.
- Introduce latency compared to unidirectional.
- Solution: Transcribe monotonic chunks at a time with attention.



Neural transducer

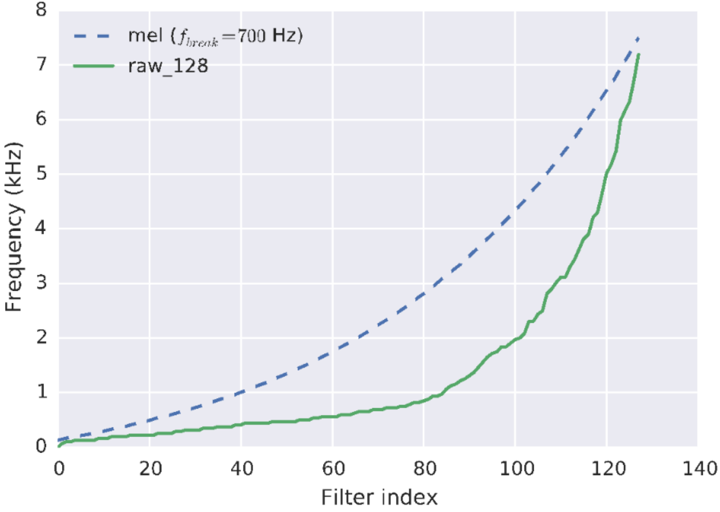


Raw waveform speech recognition

- We typically train on a much-reduced dimensional signal.
- Would like to train end-to-end.
- Learn filterbanks, instead of hand-crafting.
- A conventional RNN at audio sample rate can't learn long-enough dependencies.
  - Add a convolutional filter to a conventional system e.g. CLDNN (Sainath et al., 2015b)
  - WaveNet-style architecture. [See TTS talk on Thursday]
  - Clockwork RNN (Koutník et al., 2014) Run a hierarchical RNN at multiple rates.

Raw waveform speech recognition

Frequency distribution of learned filters differs from hand-initialization:



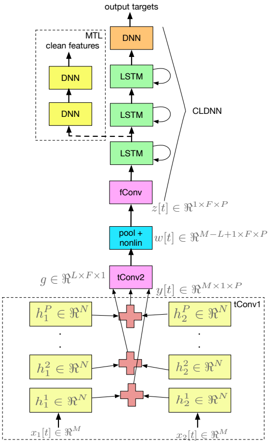
Speech recognition in noise

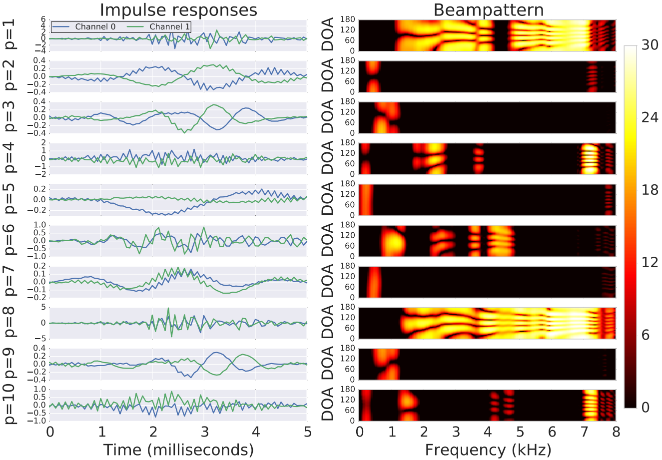
- Multi-style training ("MTS")
  - Collect noisy data.
  - Or, add realistic but randomized noise to utterances during training.
  - e.g. Through a "room simulator" to add reverberation.
  - Optionally add a clean-reconstruction loss in training.
- Train a denoiser.
- NB *Lombard* effect – voice changes in noise.

Multi-microphone speech recognition

- Multiple microphones give a richer representation
- "Closest to the speaker" has better SNR
- Beamforming
  - Given geometry of microphone array and speed of sound
  - Compute Time Delay of Arrival at each microphone
  - *Delay-and-sum*: Constructive interference of signal in chosen direction.
  - Destructive interference depends on direction / frequency of noise.
- More features for a neural network to exploit.
  - Important to preserve phase information to enable beam-forming

Factored multichannel raw waveform CLDNN (Sainath et al., 2016)





References I

Abdel-Hamid, O., Mohamed, A.-R., Jiang, H., Deng, L., Penn, G., and Yu, D. (2014). Convolutional neural networks for speech recognition. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, 22(10):1533–1545.

Amodei, D., Anubhai, R., Battenberg, E., Case, C., Casper, J., Catanzaro, B., Chen, J., Chrzanowski, M., Coates, A., Diamos, G., Elsen, E., Engel, J., Fan, L., Fougner, C., Han, T., Hannun, A. Y., Jun, B., LeGresley, P., Lin, L., Narang, S., Ng, A. Y., Ozair, S., Prenger, R., Raiman, J., Satheesh, S., Seetapun, D., Sengupta, S., Wang, Y., Wang, Z., Wang, C., Xiao, B., Yogatama, D., Zhan, J., and Zhu, Z. (2015). Deep speech 2: End-to-end speech recognition in english and mandarin. *CoRR*, abs/1512.02595.

Bengio, S., Vinyals, O., Jaitly, N., and Shazeer, N. (2015). Scheduled sampling for sequence prediction with recurrent neural networks. *CoRR*, abs/1506.03099.

Chan, W., Jaitly, N., Le, Q. V., and Vinyals, O. (2015). Listen, attend and spell. *CoRR*, abs/1508.01211.

Chorowski, J., Bahdanau, D., Serdyuk, D., Cho, K., and Bengio, Y. (2015). Attention-based models for speech recognition. *CoRR*, abs/1506.07503.

Chung, J. S., Senior, A. W., Vinyals, O., and Zisserman, A. (2016). Lip reading sentences in the wild. *CoRR*, abs/1611.05358.

Dahl, G., Yu, D., Li, D., and Acero, A. (2011). Large vocabulary continuous speech recognition with context-dependent dbn-hmms. In *ICASSP*.

Dempster, A., Laird, N., and Rubin, D. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39(B):1 – 38.

Graves, A., Fernández, S., Gomez, F., and Schmidhuber, J. (2006). Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd International Conference on Machine Learning*, pages 369–376. ACM.

Graves, A., Jaitly, N., and Mohamed, A. (2013). Hybrid speech recognition with deep bidirectional LSTM. In *ASRU*.

Grezl, Karafiat, and Cernocky (2007). Neural network topologies and bottleneck features. *Speech Recognition*.

Hermansky, H., Ellis, D., and Sharma, S. (2000). Tandem connectionist feature extraction for conventional HMM systems. In *ICASSP*.

References II

Hinton, G., Deng, L., Yu, D., Dahl, G., Mohamed, A., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T. N., and Kingsbury, B. (2012). Deep Neural Networks for Acoustic Modeling in Speech Recognition. *IEEE Signal Processing Magazine*, 29(6):82–97.

Hinton, G. E. (2002). Training products of experts by minimizing contrastive divergence. *Neural Computation*.

Jaitly, N., Le, Q. V., Vinyals, O., Sutskever, I., and Bengio, S. (2015). An online sequence-to-sequence model using partial conditioning. *CoRR*, abs/1511.04868.

Kingsbury, B. (2009). Lattice-based optimization of sequence classification criteria for neural-network acoustic modeling. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 3761–3764, Taipei, Taiwan.

Koutník, J., Greff, K., Gomez, F. J., and Schmidhuber, J. (2014). A clockwork RNN. *CoRR*, abs/1402.3511.

Mohamed, A., Dahl, G., and Hinton, G. (2009). Deep belief networks for phone recognition. In *NIPS*.

Peddinti, V., Povey, D., and Khudanpur, S. (2015). A time delay neural network architecture for efficient modeling of long temporal contexts. In *Interspeech*.

Povey, D., Kanevsky, D., Kingsbury, B., Ramabhadran, B., Saon, G., and Visweswariah, K. (2008). Boosted MMI for model and feature-space discriminative training. In *Proc. ICASSP*.

Robinson, A. and Fallside, F. (1991). A recurrent error propagation network speech recognition system. *Computer Speech and Language*, 5(3):259–274.

Sainath, T. N., Mohamed, A.-r., Kingsbury, B., and Ramabhadran, B. (2013). Deep convolutional neural networks for lvcsr. In *IEEE International Conference on Acoustics, Speech and Signal Processing*.

Sainath, T. N., Vinyals, O., Senior, A., and Sak, H. (2015a). Convolutional, long short-term memory, fully connected deep neural networks. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*.

Sainath, T. N., Weiss, R., Senior, A., Wilson, K., and Vinyals, O. (2015b). Raw waveform CLDNNs. In *Submitted to Interspeech*.

Sainath, T. N., Weiss, R. J., Wilson, K. W., Narayanan, A., and Bacchiani, M. (2016). Factored Spatial and Spectral Multichannel Raw Waveform CLDNNs. In *to appear in Proc. ICASSP*.

References III

Sak, H., Senior, A., and Beaufays, F. (2014a). Long Short-Term Memory Based Recurrent Neural Network Architectures for Large Vocabulary Speech Recognition. *ArXiv e-prints*.

Sak, H., Senior, A., and Beaufays, F. (2014b). Long Short-Term Memory Recurrent Neural Network Architectures for Large Scale Acoustic Modeling. In *INTERSPEECH 2014*.

Sak, H., Senior, A., Rao, K., Irsoy, O., Graves, A., Beaufays, F., and Schalkwyk, J. (2015). Learning acoustic frame labeling for speech recognition with recurrent neural networks. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*.

Schuster, M. and Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *Signal Processing, IEEE Transactions on*, 45(11):2673–2681.

Senior, A. and Robinson, A. (1994). Forward-backward retraining of recurrent neural networks. In *NIPS*.

Soltau, H., Liao, H., and Sak, H. (2016). Neural speech recognizer: Acoustic-to-word LSTM model for large vocabulary speech recognition. *CoRR*, abs/1610.09975.

van den Oord, A., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A. W., and Kavukcuoglu, K. (2016). Wavenet: A generative model for raw audio. *CoRR*, abs/1609.03499.

Waibel, A., Hanazawa, T., Hinton, G., Shikano, K., and Lang, K. (1989). Phoneme recognition using time-delay neural networks. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 37(3).

Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., Klingner, J., Shah, A., Johnson, M., Liu, X., Kaiser, L., Gouws, S., Kato, Y., Kudo, T., Kazawa, H., Stevens, K., Kurian, G., Patil, N., Wang, W., Young, C., Smith, J., Riesa, J., Rudnick, A., Vinyals, O., Corrado, G., Hughes, M., and Dean, J. (2016). Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, abs/1609.08144.

Xiong, W., Droppo, J., Huang, X., Seide, F., Seltzer, M., Stolcke, A., Yu, D., and Zweig, G. (2016). Achieving human parity in conversational speech recognition. *CoRR*, abs/1610.05256.