Neural Networks for Distant Speech Recognition

Steve Renals
Centre for Speech Technology Research
University of Edinburgh
s.renals@ed.ac.uk

14 May 2014

Joint work with Pawel Świętojański

Significant contributions from Peter Bell & Arnab Ghoshal
Distant Speech Recognition

... so you have your energy source your user interface who’s controlling the chip ...

hmm

rustle

click
Why study meetings?

• Natural communication scenes
  • Multistream - multiple asynchronous streams of data
  • Multimodal - words, prosody, gesture, attention
  • Multiparty - social roles, individual and group behaviours

• Meetings offer realistic, complex behaviours but in a circumscribed setting

• Applications based on meeting capture, analysis, recognition and interpretation

• Great arena for interdisciplinary research
“ASR Complete” problem

- Transcription of conversational speech
- Distant speech recognition with microphone arrays
- Speech separation, multiple acoustic channels
- Reverberation
- Overlap detection
- Utterance and speaker segmentation
- Disfluency detection
Today’s Menu

- MDM corpora: ICSI and AMI meetings corpora
- MDM systems in 2010: GMMs, beamforming, and lots of adaptation
- MDM systems in 2014: Neural networks, less beamforming, and less adaptation
Corpora
ICSI Corpus
AMI Corpus

Mic Array

Headset mic

Lapel mic

http://corpus.amiproject.org
AMI Corpus Example
Meeting recording
(c. 2005)
Meeting recording (2010s)
GMM-based systems
(State-of-the-art 2010)
Basic system

- Speech/non-speech segmentation
- PLP/MFCC features
- ML trained HMM/GMM system (122k 39D Gaussians)
- 50k vocabulary
- Trigram language model (small: 26M words, PPL 78)
- Weighted FST decoder
Additional components

- Microphone array front end
- Speaker / channel adaptation
  - Vocal tract length normalisation (VTLN)
  - Maximum likelihood linear regression (MLLR)
- Input feature transform – LDA/STC
- Discriminative training
  - eg boosted maximum mutual information, BMMI
- Discriminative features
- Model combination
GMM results (WER)

ASR Word Error Rates for GMM/HMM Systems

<table>
<thead>
<tr>
<th>System</th>
<th>AMI WER (%)</th>
<th>ICSI WER (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SDM</td>
<td>63.2</td>
<td>56.1</td>
</tr>
<tr>
<td>MDM beamforming</td>
<td>54.8</td>
<td>46.8</td>
</tr>
<tr>
<td>IHM</td>
<td>29.6</td>
<td></td>
</tr>
</tbody>
</table>

GMM results (WER)
Microphone array processing for distant speech recognition

- Mic array processing in AMIDA ASR system (Hain et al, 2012)
  - Wiener noise filter
  - Filter-sum beamforming based on time-delay-of-arrival
  - Viterbi smoother post processing
  - Track direction of maximum energy
- Optimise beamforming for speech recognition
  - LIMABEAM (Seltzer et al, 2004, 2006) [explicit]
  - Simply concatenate feature vectors from multiple mics (Marino and Hain, 2011) [implicit]
(Deep) Neural Networks
The Perceptron
(Rosenblatt)
The Perceptron
(Rosenblatt)
The Perceptron (Rosenblatt)
MLPs and backprop (mid 1980s)
MLPs and backprop

- Train multiple layers of hidden units – nested nonlinear functions
- Powerful feature detectors
- Posterior probability estimation
- Theorem: any function can be approximated with a single hidden layer
“Hybrid” Neural network acoustic models (1990s)

DARPA RM 1992

Broadcast news 1998
20.8% WER
(best GMM-based system, 13.5%)

Cook, Christie, Ellis, Fosler-Lussier, Gotoh, Kingsbury, Morgan, Renals, Robinson, & Williams, DARPA, 1999

Bourlard & Morgan, 1994
Robinson, IEEE TNN 1994
Renals, Morgan, Cohen & Franco, ICASSP 1992
NN acoustic models
Limitations vs GMMs

- Computationally restricted to monophone outputs
  - CD-RNN factored over multiple networks – limited within-word context
- Training not easily parallelisable
  - experimental turnaround slower
  - systems less complex (fewer parameters)
    - RNN – <100k parameters
    - MLP – ~1M parameters
- Rapid adaptation hard (cf MLLR)
GMM

SVM

CRF

NN Winter #2
Discriminative long-term features – Tandem

- A neural network-based technique provided the biggest increase in accuracy in speech recognition during the 2000s
- **Tandem features** (Hermansky, Ellis & Sharma, 2000)
  - use (transformed) outputs or (bottleneck) hidden values as input features for a GMM
  - deep networks – e.g. 5 layer MLP to obtain bottleneck features (Grézl, Karafiát, Kontár & Černocký, 2007)
  - reduces errors by about 10% relative (Hain, Burget, Dines, Garner, Grezl, el Hannani, Huijbregts, Karafiat, Lincoln & Wan, 2012)
Deep Neural Networks (2010s)

Hybrid

CD Phone Outputs

3–8 hidden layers

Bottleneck layer 26

Hidden units 2000

MFCC Inputs (39*9=351)

Dahl, Yu, Deng & Acero, IEEE TASLP 2012

Hinton, Deng, Yu, Dahl, Mohamed, Jaitly, Senior, Vanhoucke, Nguyen, Sainath & Kingsbury, IEEE SP Mag 2012
Deep neural networks

What’s new?
Deep neural networks

1. **Unsupervised pretraining** *(Hinton, Osindero & Teh, 2006)*
   - Train a stacked RBM generative model, then finetune
   - Good initialisation
   - Regularisation

2. **Deep** – many hidden layers
   - Deeper models more accurate
   - GPUs gave us the computational power

3. **Wide** output layer (context dependent phone classes) rather than factorised into multiple nets
   - More accurate phone models
   - GPUs gave us the computational power
Deep neural networks

1. **Unsupervised pretraining** *(Hinton, Osindero & Teh, 2006)*
   - Train a stacked RBM generative model, then finetune
   - Good initialisation
   - Regularisation

2. **Deep** – many hidden layers
   - Deeper models more accurate
   - GPUs gave us the computational power

3. **Wide** output layer (context dependent phone classes) rather than factorised into multiple nets
   - More accurate phone models
   - GPUs gave us the computational power

Switchboard Hub5 '00 test set
300 hour training set

---GMM/BMMI---

---DNN/CE---

---DNN/sMBR---

<table>
<thead>
<tr>
<th>Method</th>
<th>Hub5 '00 Test Set</th>
<th>300 Hour Training Set</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHE</td>
<td>33.0</td>
<td>18.4</td>
</tr>
<tr>
<td>AVE</td>
<td>25.8</td>
<td>12.6</td>
</tr>
<tr>
<td>CHE</td>
<td>33.0</td>
<td>18.4</td>
</tr>
</tbody>
</table>

KALDI

http://kaldi.sf.net/
Neural network acoustic models

Softmax output layer

Automatically learned feature extraction

Aim to learn representations for distant speech recognition based on multiple mic channels

~6000 CD phone outputs

~2000 hidden units

3-8 hidden layers

9x39 MFCC inputs

Neural network

Acoustic models
Neural network acoustic models

- **~6000 CD phone outputs**
- **~2000 hidden units**
- **3-8 hidden layers**
- **Softmax output layer**
- **Automatically learned feature extraction**
- **Multi-channel input**
- **Spectral domain?**

Aim to learn representations for distant speech recognition based on multiple mic channels.
Neural network acoustic models for distant speech recognition

- NNs have proven to result in accurate systems for a variety of tasks – TIMIT, WSJ, Switchboard, Broadcast News, Lectures, Aurora4, …
- NNs can integrate information from multiple frames of data (in comparison with GMMs)
- NNs can construct feature representations, from multiple sources of data
- NNs are well suited to learning multiple modules with a common objective function
Baseline DNN system

50,000 word pronunciation dictionary

Small trigram LM
(PPL 78, trained on 26M words)

mic array

Wiener filter noise cancellation
Smoothed tdoa estimates
Delay-sum beamforming

~4000 tied state outputs

2048 hidden units

6 hidden layers

11x120 FBANK inputs

11x120 FBANK inputs
Baseline GMM results

ASR Word Error Rates for GMM/HMM Systems

- **AMI**: 63.2%
- **ICSI**: 56.1%
- **AMI**: 54.8%
- **ICSI**: 46.8%
- **AMI**: 29.6%
Baseline DNN results

ASR Word Error Rates for baseline DNN/HMM Systems

<table>
<thead>
<tr>
<th>System</th>
<th>WER/%</th>
</tr>
</thead>
<tbody>
<tr>
<td>SDM AMI</td>
<td>53.1</td>
</tr>
<tr>
<td>ICSI</td>
<td>47.8</td>
</tr>
<tr>
<td>MDM beamforming AMI</td>
<td>49.5</td>
</tr>
<tr>
<td>ICSI</td>
<td>41.0</td>
</tr>
<tr>
<td>IHM</td>
<td>26.6</td>
</tr>
</tbody>
</table>
Baseline DNN results

ASR Word Error Rates for baseline DNN/HMM Systems

DNN has 10–15% WER reduction over GMM

Baseline DNN results

- AMI: 53.1
- ICSI: 47.8
- AMI: 49.5
- ICSI: 41.0
- AMI: 26.6

DNN has 10–15% WER reduction over GMM
Concatenating input features

- 8 x 11x120 FBANK inputs
- 6 hidden layers
- 2048 hidden units
- ~4000 tied state outputs
DNN results

beamforming vs concatenation

ASR Word Error Rates for AMI corpus MDM

<table>
<thead>
<tr>
<th>Method</th>
<th>WER/%</th>
</tr>
</thead>
<tbody>
<tr>
<td>GMM / Beamforming</td>
<td>54.8</td>
</tr>
<tr>
<td>DNN / Beamforming</td>
<td>49.5</td>
</tr>
<tr>
<td>DNN / Concatenation (4ch)</td>
<td>51.2</td>
</tr>
</tbody>
</table>
Convolutional Neural Networks (CNNs)
Convolutional Neural Networks

Yann Le Cun, 1989 onwards
CNN – Single channel

Inputs

maxpool

shared weights

convolitional bands

p1

h1

v1

v2

h2

v3

v4

h3

v5

h4

v40

...
CNN – Single channel

- 5 sigmoid layers
- 2048 hidden units
- ~4000 tied state outputs

Inputs

- Convolutional bands
- Maxpool
- Shared weights
CNN – Single channel

- 5 sigmoid layers
- 2048 hidden units
- ~4000 tied state outputs
- 128 convolutional filterbanks
- width 9, shift 1
- statics, deltas, double-deltas for all context frames of band
- maxpool size 2
- shared weights
- convolutional bands
- maxpool
CNN – Multi-channel
CNN – Multi-channel

- 5 sigmoid layers
- 2048 hidden units
- ~4000 tied state outputs
- Inputs
- Convolutional bands
- Maxpool
- Shared weights
- V1, V2, V3, V4, V5, V40
CNN – Cross-channel

Diagram showing a CNN model with layers such as convolutional bands, cross-channel maxpooling, cross-band maxpooling, and shared weights.
SDM systems

![Graph showing WER/% for different systems and datasets.](image)

- **AMI**: 63.2, 53.1, 51.3, 50.8
- **ICSI**: 56.1, 47.8, 46.5, 45.9
- Systems: GMM BF, DNN BF, CNN BF, DNN BF / maxout
MDM systems

CNN has 7–8% WER reduction over DNN (16–19% WER reduction over GMM)

<table>
<thead>
<tr>
<th>System</th>
<th>AMI</th>
<th>ICSI</th>
</tr>
</thead>
<tbody>
<tr>
<td>GMM BF</td>
<td>54.8</td>
<td>46.8</td>
</tr>
<tr>
<td>DNN BF</td>
<td>49.5</td>
<td>41.0</td>
</tr>
<tr>
<td>CNN BF</td>
<td>45.9</td>
<td>38.1</td>
</tr>
<tr>
<td>CNN x-chan</td>
<td>48.4</td>
<td>37.8</td>
</tr>
<tr>
<td>DNN BF / maxout</td>
<td>46.4</td>
<td>39.0</td>
</tr>
</tbody>
</table>
Discussion

- CNN with single distant mic has similar WER to DNN with 8 beamformed mics
- Channel-wise convolution followed by cross-channel max-pooling better than multi-channel convolution
- Cross-channel CNNs still work even if channel order changed at test... able to pick the most informative channel
- Invariances across time and frequency important in multi-channel case
- CNN improvements over DNN less for ‘piecewise’ hidden units (maxout, ReLU) [but only DSR data....]
Future data?

- Existing corpora: MC-WSJ, AMI, ICSI
- Desiderata for new datasets
  - Recorded in a variety of environments
  - Highly multimodal
  - Natural & conversational speech
  - Wide range of challenges
    - signal processing
    - language processing
  - Evaluation campaigns
Future data?

- Sheffield Wargames Corpus
- Many mic channels
- Arrays + headmount
- Tracking info
- Cameras

http://mini.dcs.shef.ac.uk/data-2/
Conclusions

• Improvements due to:
  • deep structures to learn feature representations,
  • wide context-dependent output,
  • temporal context at input, correlated features

• From these experiments and others
  • DNNs offer 10–30% relative improvement over GMMs
  • CNNs offer 5–10% relative improvement over DNNs
  • Beamforming still ~5% better than multichannel learning
  • Log spectral features give improvement over MFCCs
Practical details for DNNs

- Computing platform
  - High-end PC, with “gamers GPU” (e.g. GTX690)

- Open source software, eg:
  - Theano – [http://deeplearning.net/software/theano](http://deeplearning.net/software/theano)
  - Pylearn2 - [http://deeplearning.net/software/pylearn2/](http://deeplearning.net/software/pylearn2/)
  - Torch7 – [http://www.torch.ch](http://www.torch.ch)
  - Quicknet – [http://www.icsi.berkeley.edu/Speech/qn.html](http://www.icsi.berkeley.edu/Speech/qn.html)