Learning representations of speech in neural network acoustic models

Steve Renals

Centre for Speech Technology Research
University of Edinburgh

s.renals@ed.ac.uk 3 December 2014
Introduction
Neural network acoustic models (1990s)
Neural network acoustic models (1990s)

- 9x39 MFCC inputs
- 1 hidden layer
- ~2000 hidden units
- ~40 CI phone outputs

Bourlard & Morgan, 1994
Neural network acoustic models (1990s)

~40 CI phone outputs

~2000 hidden units

1 hidden layer

9x39 MFCC inputs

Bourlard & Morgan, 1994

Renals, Morgan, Cohen & Franco, ICASSP 1992

Error (%) vs Million Parameters

DARPA RM 1992

CI-HMM

CI-MLP

CD-HMM

MIX
Neural network acoustic models (1990s)

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

Neural network
acoustic models (1990s)

~40 CI phone outputs
~2000 hidden units
1 hidden layer
9x39 MFCC inputs

Bourlard & Morgan, 1994
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)
NN acoustic models

Benefits

• Fewer limitations on inputs
  • Correlated features
  • Multi-frame windows
• Discriminative training criteria (frame level and sequence level)
• Can be used to generate ‘higher-level’ features
  • tandem, posteriorgrams
  • bottleneck features
(Deep) neural network acoustic models (2010s)
(Deep) neural network acoustic models (2010s)

- 9x39 MFCC inputs
- 3-8 hidden layers
- ~2000 hidden units
- ~6000 CD phone outputs

Dahl, Yu, Deng & Acero, IEEE TASLP 2012
Hinton, Deng, Yu, Dahl, Mohamed, Jaitly, Senior, Vanhoucke, Nguyen, Sainath & Kingsbury, IEEE SP Mag 2012
(Deep) neural network acoustic models (2010s)

- ~6000 CD phone outputs
- ~2000 hidden units
- 3-8 hidden layers
- 9x39 MFCC inputs

WIDE Softmax output layer

Dahl, Yu, Deng & Acero, IEEE TASLP 2012
Hinton, Deng, Yu, Dahl, Mohamed, Jaitly, Senior, Vanhoucke, Nguyen, Sainath & Kingsbury, IEEE SP Mag 2012
(Deep) neural network acoustic models (2010s)

- **WIDE**
  - Softmax output layer
  - ~6000 CD phone outputs
  - ~2000 hidden units

- **DEEP**
  - Automatically learned feature extraction
  - 3-8 hidden layers
  - 9x39 MFCC inputs

Dahl, Yu, Deng & Acero, IEEE TASLP 2012
Hinton, Deng, Yu, Dahl, Mohamed, Jaitly, Senior, Vanhoucke, Nguyen, Sainath & Kingsbury, IEEE SP Mag 2012
(Deep) neural network acoustic models (2010s)

- Approximately 6000 CD phone outputs
- Approximately 2000 hidden units
- 3-8 hidden layers
- Softmax output layer
- Automatically learned feature extraction

Acoustic Input:
- Spectral?
- Cepstral?
- Derived features?

Sources:
- Dahl, Yu, Deng & Acero, IEEE TASLP 2012
- Hinton, Deng, Yu, Dahl, Mohamed, Jaitly, Senior, Vanhoucke, Nguyen, Sainath & Kingsbury, IEEE SP Mag 2012
(Deep) neural network acoustic models (2010s)

- **ACOUSTIC INPUT**: Spectral? Cepstral? Derived features?
- **DEEP**: 3-8 hidden layers
  - Automatically learned feature extraction
- **WIDE**: Softmax output layer
  - ~2000 hidden units
  - ~6000 CD phone outputs
- **ACTIVATION FUNCTIONS**: pooling, RELU, gated units
(Deep) neural network acoustic models (2010s)

- DEEP: 3-8 hidden layers, ~2000 hidden units, ~6000 CD phone outputs
- WIDE: Softmax output layer

- Automatically learned feature extraction
- WEIGHT SHARING: adaptation, CNNs

- ACOUSTIC INPUT: Spectral? Cepstral? Derived features?
- ACTIVATION FUNCTIONS: pooling, RELU, gated units
(Deep) neural network acoustic models (2010s)

- 3-8 hidden layers
- ~2000 hidden units
- 9x39 MFCC inputs
- ~6000 CD phone outputs

**WEIGHT SHARING**
- adaptation, CNNs

**ACTIVATION FUNCTIONS**
- pooling, RELU, gated units

**ACOUSTIC INPUT**
- Spectral? Cepstral? Derived features?

**DEEP**
- Automatically learned feature extraction

**WIDE**
- Softmax output layer
Representation learning for acoustic models

- DISTANT SPEECH RECOGNITION
  - Combining input channels from multiple microphones
  - Convolutional Neural Networks

- SPEAKER ADAPTATION
  - Model-based adaptation of NN acoustic model – unsupervised, compact, efficient, accurate
  - Learning Hidden Unit Contributions (LHUC)

- DOMAIN ADAPTATION
  - Limited in-domain data
  - Multi-Level Adaptive Networks (MLAN)
Part One

Distant Speech Recognition
Distant speech recognition

- 11x120 FBANK inputs
- 6 hidden layers
- 2048 hidden units
- ~4000 tied state outputs

Wiener filter noise cancellation
Smoothed tdoa estimates
Delay-sum beamforming

mic array
Distant speech recognition

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

Mic array
Convolutional Neural Networks

Yann Le Cun, 1989 onwards
CNN acoustic model
CNN acoustic model

Statics, deltas, double-deltas for all acoustic context frames
CNN acoustic model

Statics, deltas, double-deltas for all acoustic context frames

128 convolutional filterbanks
CNN acoustic model

Statics, deltas, double-deltas for all acoustic context frames

128 convolutional filterbanks

Width 9, shift 1
CNN acoustic model

Statics, deltas, double-deltas for all acoustic context frames

maxpool size 3

128 convolutional filterbanks

width 9
shift 1
CNN acoustic model

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

Statics, deltas, double-deltas for all acoustic context frames

maxpool size 3

128 convolutional filterbanks

width 9
shift 1

v1, v2, v3, v4, v5, ..., v40
CNN – Multi-channel
CNN – Multi-channel
CNN – Multi-channel

- ~4000 tied state outputs
- 2048 hidden units
- 5 sigmoid layers
- Shared weights
- Convolutional bands
- Maxpool

Inputs
CNN – Multi-channel
Details?
Experimental Results?
Some Conclusions?
Details?  
Experimental Results?  
Some Conclusions?

Swietojanski & Renals,  
Convolutional Neural Networks for Distant Speech Recognition

Poster later today,  
Part Two
Speaker Adaptation
Speaker/Channel Adaptation
Speaker/Channel Adaptation

- GMM-based systems
  - Maximum likelihood linear transform (MLLT) family
    - MLLR, CMLLR, …
  - MAP, Eigenvoices, CAT, …
Speaker/Channel Adaptation

- GMM-based systems
  - Maximum likelihood linear transform (MLLT) family
    - MLLR, CMLLR, …
  - MAP, Eigenvoices, CAT, …
- Adapting DNN acoustic models
  - Feature space transforms (e.g. CMLLR, LIN)
  - Auxiliary features (e.g. iVectors)
  - Model-based
Speaker/Channel Adaptation

- GMM-based systems
  - Maximum likelihood linear transform (MLLT) family
    - MLLR, CMLLR, …
  - MAP, Eigenvoices, CAT, …
- Adapting DNN acoustic models
  - Feature space transforms (e.g. CMLLR, LIN)
  - Auxiliary features (e.g. iVectors)
  - Model-based
- Ideally: COMPACT / UNSUPERVISED / EFFICIENT
Model-based NN adaptation
Model-based NN adaptation

- Adaptation of different weight subsets (Liao 2013)
  - 5% relative decrease in WER when all 60M weights adapted
Model-based NN adaptation

- Adaptation of different weight subsets (Liao 2013)
  - 5% relative decrease in WER when all 60M weights adapted
- Adaptation cost based on KL divergence between SI and SA output distributions (Yu et al 2013)
  - 3% relative decrease in WER on Switchboard
Model-based NN adaptation

- Adaptation of different weight subsets (Liao 2013)
  - 5% relative decrease in WER when all 60M weights adapted
- Adaptation cost based on KL divergence between SI and SA output distributions (Yu et al 2013)
  - 3% relative decrease in WER on Switchboard
- Increase compactness by SVD factorisation of weight matrix (Xue et al 2014)
Model-based NN adaptation

- Adaptation of different weight subsets (Liao 2013)
  - 5% relative decrease in WER when all 60M weights adapted
- Adaptation cost based on KL divergence between SI and SA output distributions (Yu et al 2013)
  - 3% relative decrease in WER on Switchboard
- Increase compactness by SVD factorisation of weight matrix (Xue et al 2014)
- Automatically adapt specific parameter subsets – output biases (Yao et al 2012), slope and bias of hidden units (Siniscalchi et al 2013)
Model-based NN adaptation

- Adaptation of different weight subsets (Liao 2013)
  - 5% relative decrease in WER when all 60M weights adapted

- Adaptation cost based on KL divergence between SI and SA output distributions (Yu et al 2013)
  - 3% relative decrease in WER on Switchboard

- Increase compactness by SVD factorisation of weight matrix (Xue et al 2014)

- Automatically adapt specific parameter subsets – output biases (Yao et al 2012), slope and bias of hidden units (Siniscalchi et al 2013)

- Speaker codes (Bridle & Cox 1990; Abdel-Hamid & Jiang 2013) – model-based adaptation using auxiliary features
LHUC
Learning Hidden Unit Contributions
Learning Hidden Unit Contributions

Key idea: add a learnable *speaker-dependent amplitude* to each hidden unit
Learning Hidden Unit Contributions

Key idea: add a learnable *speaker-dependent amplitude* to each hidden unit

\[ h^l_m = a(r^l_m) \circ \sigma^l(W^{l\top}h^{l-1}_m) \]
Learning Hidden Unit Contributions

Key idea: add a learnable **speaker-dependent amplitude** to each hidden unit

\[
 h^l_m = a(r^l_m) \circ \sigma^l(W^l h^{l-1}_m)
\]

Amplitude is a sigmoid in range 0–2, parameterised by \( r \)

SI Model: set amplitudes to 1
SD Model: learn amplitudes from data, per speaker

Use first pass decoding to obtain adaptation targets
LHUC

~6000 CD phone outputs

~2000 hidden units

3-8 hidden layers

~2000 hidden units

inputs
LHUC

Compact: 1 SD parameter per hidden unit

~6000 CD phone outputs

~2000 hidden units

3-8 hidden layers

~2000 hidden units

inputs
Compact: 1 SD parameter per hidden unit

Unsupervised: No speaker-adaptive training

~6000 CD phone outputs

~2000 hidden units

3-8 hidden layers

~2000 hidden units

inputs
LHUC

**Compact:** 1 SD parameter per hidden unit

**Unsupervised:** No speaker-adaptive training

**Efficient:** Few iterations of gradient descent on test/adapt data

- ~6000 CD phone outputs
- ~2000 hidden units
- ~2000 hidden units
- 3-8 hidden layers
LHUC

**Compact:** 1 SD parameter per hidden unit

**Unsupervised:** No speaker-adaptive training

**Efficient:** Few iterations of gradient descent on test/adapt data

**Flexible:** Applicable to different hidden unit transfer functions
Speaker-dependent amplitudes

Mean/Max/Min of adapted amplitudes in each layer
LHUC experiment I TED Talks (IWSLT)

Training: 143 hrs – 50M frames
12,000 tied states
(sigmoid) 6 x 2000 unit hidden layers
(maxout) 6 x 1500 maxpool[2] hidden layers
13-D PLP features + Δ + ΔΔ
TED Talks

TED Talks - IWSLT tst2011

WER/%

Sigmoid

Maxout

DNN

+LHUC

+CMLLR

+CMLLR+LHUC
LHUC experiment 1
TED Talks

TED Talks – IWSLT tst2011

WER/%

15.2 15.2 14.3 13.7 13.5 13.9 13.6 12.9 12.7 12.5 12.9 12.7 11.9

Sigmoid

DNN +LHUC +CMLLR +CMLLR+LHUC

6-7%
LHUC experiment 1
TED Talks

TED Talks – IWSLT tst2011

WER/%

DNN  +LHUC  +CMLLR  +CMLLR+LHUC

Sigmoid  RELU  Maxout

15.2  15.2  14.3  13.7  13.5  13.9  13.6  12.8  12.5  12.9  12.7  11.9

10%
TED Talks

TED Talks – IWSLT tst2011

<table>
<thead>
<tr>
<th></th>
<th>DNN</th>
<th>+LHUC</th>
<th>+CMLLR</th>
<th>+CMLLR+LHUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sigmoid</td>
<td>15.2</td>
<td>15.2</td>
<td>13.7</td>
<td>13.7</td>
</tr>
<tr>
<td>RELU</td>
<td>14.3</td>
<td>13.5</td>
<td>12.8</td>
<td>12.5</td>
</tr>
<tr>
<td>Maxout</td>
<td>15-17%</td>
<td>15-17%</td>
<td>15-17%</td>
<td>15-17%</td>
</tr>
<tr>
<td>WER/%</td>
<td>15.2</td>
<td>15.2</td>
<td>13.7</td>
<td>13.7</td>
</tr>
<tr>
<td></td>
<td>14.3</td>
<td>13.5</td>
<td>12.8</td>
<td>12.5</td>
</tr>
<tr>
<td></td>
<td>15-17%</td>
<td>15-17%</td>
<td>15-17%</td>
<td>15-17%</td>
</tr>
</tbody>
</table>
TED Talks – IWSLT tst2011

More Details at SLT!
LHUC experiment 2
Aurora-4
LHUC experiment 2
Aurora-4

Standard Aurora-4 protocol

**DNN/Sigmoid** – 5x2000 unit hidden layers

**CNN/Sigmoid** – additional conv layer (300 filters)

**CNN/Maxout** - 300 filters, 6x1000 maxpool[2] hidden layers

**CNN/Maxout (Annealed Dropout)** (Rennie et al, SLT 2014)
LHUC experiment 2
Aurora-4

Aurora-4 (average of all conditions)

- SI: 13.9
- LHUC: 12.4
- DNN/Sigmoid: 13.6
- CNN/Sigmoid: 11.9
- CNN/Maxout: 11.8
- CNN/Maxout (Annealed dropout): 10.8
- CNN/Maxout (Annealed dropout): 8.6
LHUC experiment 2
Aurora-4

Aurora-4 (average of all conditions)

<table>
<thead>
<tr>
<th>Model</th>
<th>WER (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>DNN/Sigmoid</td>
<td>13.9</td>
</tr>
<tr>
<td>CNN/Sigmoid</td>
<td>13.6</td>
</tr>
<tr>
<td>CNN/Maxout</td>
<td>11.8</td>
</tr>
<tr>
<td>CNN/Maxout (Annealed dropout)</td>
<td>10.8</td>
</tr>
</tbody>
</table>

13% improvement
LHUC experiment 2
Aurora-4

Aurora-4 (average of all conditions)

<table>
<thead>
<tr>
<th>Method</th>
<th>WER (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SI</td>
<td>13.9</td>
</tr>
<tr>
<td>DNN/Sigmoid</td>
<td>12.4</td>
</tr>
<tr>
<td>CNN/Sigmoid</td>
<td>11.9</td>
</tr>
<tr>
<td>CNN/Maxout (Annealed dropout)</td>
<td>9.5</td>
</tr>
<tr>
<td>CNN/Maxout</td>
<td>10.8</td>
</tr>
</tbody>
</table>

10-12% LHUC
LHUC experiment 2
Aurora-4

Aurora-4 (average of all conditions)

<table>
<thead>
<tr>
<th>Model Configuration</th>
<th>SI</th>
<th>LHUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>DNN/Sigmoid</td>
<td>13.9</td>
<td>12.4</td>
</tr>
<tr>
<td>CNN/Sigmoid</td>
<td>13.6</td>
<td>11.9</td>
</tr>
<tr>
<td>CNN/Maxout</td>
<td>11.8</td>
<td>9.5</td>
</tr>
<tr>
<td>CNN/Maxout (Annealed dropout)</td>
<td>10.8</td>
<td>8.6</td>
</tr>
</tbody>
</table>

WER/\%
LHUC experiment 2
Aurora-4

Aurora-4 (average of all conditions)

<table>
<thead>
<tr>
<th>Architecture</th>
<th>WER/%</th>
</tr>
</thead>
<tbody>
<tr>
<td>DNN/Sigmoid</td>
<td>13.9</td>
</tr>
<tr>
<td>CNN/Sigmoid</td>
<td>12.4</td>
</tr>
<tr>
<td>CNN/Maxout</td>
<td>13.6</td>
</tr>
<tr>
<td>CNN/Maxout (Annealed dropout)</td>
<td>11.9</td>
</tr>
<tr>
<td>CNN/Maxout</td>
<td>11.8</td>
</tr>
<tr>
<td>CNN/Maxout</td>
<td>9.5</td>
</tr>
<tr>
<td>CNN/Maxout</td>
<td>10.8</td>
</tr>
<tr>
<td>CNN/Maxout</td>
<td>8.6</td>
</tr>
</tbody>
</table>

New!
LHUC

- Speaker-dependent hidden unit amplitudes
- Compact, unsupervised adaptation scheme
- Applicable to different hidden unit functions
- Doesn’t require adaptive training
- Experiments on TED/Aurora-4
  - 10-20% relative WER decrease (few minutes of speech)
  - 3% rel WER decrease from 10s speech
  - Complementary to CMLLR transform
  - (need to combine with i-vector features)
Part Three
Domain Adaptation
Conventional DNN

- In-domain PLP features
  - Standard features
    - Baseline HMMs
    - Train in-domain DNNs on PLP features
      - Tandem features
        - Tandem HMMs
        - Hybrid HMMs
Out-of-domain NN features

- Train DNNs on OOD data
- OOD posterior features generated for in-domain data
- In-domain PLP features
- Tandem features
- OOD Tandem HMMs
MLAN
Multi-Layer Adaptive Networks

Train DNNs on OOD data

OOD posterior features generated for in-domain data

In-domain PLP features

Tandem features

OOD Tandem HMMs

Train in-domain DNNs on Tandem features

Tandem MLAN HMMs

MLAN features

Hybrid MLAN HMMs
MLAN Experiment
TED Talks
TED Talks – IWSLT test2011

WER/%
12.6
10.9
11.2

Tandem
SWB MLAN
AMI MLAN
TED Talks – IWSLT test2011

MLAN Experiment

Tandem SWB MLAN AMI MLAN

WER/%

12.6 10.9 11.2

11-13%
MLAN

- Use NNs trained on out-of-domain data to generate features with in-domain data
- Does not bias training to out-of-domain data
- Useful way to include tandem/bottleneck features in NN/HMM hybrids
- Can be used for cross-lingual ASR (SLT)
Conclusion
Conclusions
Conclusions

- Learning representations can give significant improvements in accuracy
- **CNNs for multichannel ASR**
  - beneficial for uncalibrated arrays
- **LHUC** – speaker-dependent hidden unit amplitudes
  - unsupervised, compact speaker adaptation
- **MLAN** – out-of-domain NN acoustic features
  - effective domain adaptation
Conclusions

• Learning representations can give significant improvements in accuracy

• **CNNs for multichannel ASR**
  • beneficial for uncalibrated arrays

• **LHUC – speaker-dependent hidden unit amplitudes**
  • unsupervised, compact speaker adaptation

• **MLAN – out-of-domain NN acoustic features**
  • effective domain adaptation

Thanks!
Conclusions

- Learning representations can give significant improvements in accuracy
- CNNs for multichannel ASR
  - beneficial for uncalibrated arrays
- LHUC – speaker-dependent hidden unit amplitudes
  - unsupervised, compact speaker adaptation
- MLAN – out-of-domain NN acoustic features
  - effective domain adaptation
Learning representations of speech in neural network acoustic models

Steve Renals
Centre for Speech Technology Research
University of Edinburgh

s.renals@ed.ac.uk

3 December 2014
# Aurora-4 results

<table>
<thead>
<tr>
<th>Method</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>Avg</th>
</tr>
</thead>
<tbody>
<tr>
<td>DNN/Sigmoid</td>
<td>5.1</td>
<td>9.3</td>
<td>9.3</td>
<td>20.8</td>
<td>13.9</td>
</tr>
<tr>
<td>DNN/Sigmoid +LHUC</td>
<td>4.5</td>
<td>8.6</td>
<td>7.4</td>
<td>18.3</td>
<td>12.4</td>
</tr>
<tr>
<td>CNN/Sigmoid</td>
<td>5.0</td>
<td>9.2</td>
<td>9.0</td>
<td>20.3</td>
<td>13.6</td>
</tr>
<tr>
<td>CNN/Sigmoid +LHUC</td>
<td>4.3</td>
<td>8.3</td>
<td>7.2</td>
<td>17.6</td>
<td>11.9</td>
</tr>
<tr>
<td>CNN/Maxout</td>
<td>4.2</td>
<td>7.7</td>
<td>8.2</td>
<td>17.7</td>
<td>11.8</td>
</tr>
<tr>
<td>CNN/Maxout +LHUC</td>
<td>3.8</td>
<td>6.5</td>
<td>5.5</td>
<td>14.3</td>
<td>9.5</td>
</tr>
<tr>
<td>CNN/Maxout (Annealed d/o)</td>
<td>4.0</td>
<td>5.9</td>
<td>7.0</td>
<td>15.8</td>
<td>10.8</td>
</tr>
<tr>
<td>CNN/Maxout +LHUC (Ann d/o)</td>
<td>3.6</td>
<td>6.7</td>
<td>5.2</td>
<td>13.5</td>
<td>8.6</td>
</tr>
</tbody>
</table>