Towards Speaker Adaptive Training of Deep Neural Network Acoustic Models

Yajie Miao    Hao Zhang    Florian Metze
Language Technologies Institute
School of Computer Science    Carnegie Mellon University
Outline

1 Motivation

2 Speaker-normalized feature space with i-vectors
   - AdaptNN -- Bottom adaptation layers
   - iVecNN -- Linear feature shift

3 Speaker adaptive training of DNNs

4 Experiments

5 Summary & Future Work
Motivation

- Deep neural networks become the state of the art for acoustic modeling.

- For GMM models, speaker adaptive training has been a standard technique for improving WERs.

- Various methods [1,2,3,4,5] have been proposed to perform speaker adaptation for DNNs. However, how we can do SAT for DNN is not clear.

- In this work, we aim to achieve complete speaker adaptive training for DNN acoustic models.
SAT for HMM/GMM

- Starts with an initial GMM model and estimates fMLLR affine transforms
- Updates model parameters with fMLLR applied and then re-estimates fMLLR transforms. Repeats until convergence.
SAT for HMM/GMM

- Starts with an initial GMM model and estimates fMLLR affine transforms
- Updates model parameters with fMLLR applied and then re-estimates fMLLR transforms. Repeats until convergence.

We want to do the similar thing for DNNs!!
Basic Idea for SAT-DNN

- Start with an initial DNN model which is the regular DNN we train for hybrid systems
- Learn a function which takes advantage of i-vectors and projects DNN inputs into a speaker-normalized feature space
- Update the DNN model in the new feature space
Bottom Adaptation Layers

- Insert a smaller adaptation network between the initial DNN and the inputs
- I-vectors are appended to the outputs of each hidden layer
- By using i-vectors, this network AdaptNN transforms the original DNN inputs into a speaker-normalized space
The output layer of AdaptNN has the same dimension as the original input features.

The output layer adopts the linear activation function, while others use sigmoid.

Parameters of AdaptNN can be estimated by the standard error back-propagation by fixing initial DNN.
iVecNN takes speaker i-vectors as inputs and generates a linear feature shift for each speaker.

This feature shift is added to the original DNN inputs, and the resulting features become more speaker-normalized.
The output layer of iVecNN has the same dimension as DNN inputs and takes linear activation function.

Parameters of iVecNN can be estimated by the standard error back-propagation.

More flexible. It can be applied both to DNNs and also to convolutional neural nets (CNNs) [6].
Procedures of SAT-DNN -- Training

**Step 1** Train the initial DNN model. This DNN can be trained on SI (e.g., fbank) or SA features (e.g., fMLLR)
Procedures of SAT-DNN -- Training

**Step 1** Train the initial DNN model. This DNN can be trained on SI (e.g., fbank) or SA features (e.g., fMLLR)

**Step 2** Learn the feature function (AdaptNN or iVecNN) by keeping the initial DNN fixed. This step requires speaker i-vectors as the side information for feature transformation.
Procedures of SAT-DNN -- Training

**Step 1** Train the initial DNN model. This DNN can be trained on SI (e.g., fbank) or SA features (e.g., fMLLR)

**Step 2** Learn the feature function (AdaptNN or iVecNN) by keeping the initial DNN fixed. This step requires speaker i-vectors as the side information for feature transforming

**Step 3** Re-finetune the DNN parameters in the new feature space while keeping the feature function fixed. This finally gives us SAT-DNN
Procedures of SAT-DNN -- Decoding

**Step 1** Given a testing speaker, just extract the i-vector for adaptation. I-vector extraction is **totally unsupervised**

**Step 2** Input the speech features and the i-vectors into this architecture for decoding. This projects the input features into the speaker-normalized space and adapts the SAT-DNN model automatically to this testing speaker
Procedures of SAT-DNN -- Decoding

- Since i-vector extraction is totally unsupervised, **no initial decoding pass and no fine-tuning on the adaptation data**

- Only **one single pass of decoding**, although we are doing unsupervised adaptation

**Very Efficient Unsupervised Adaptation**
Comparison to related work


- Concatenate i-vectors with the original features directly and train the whole network from scratch

- We failed to get obvious gains from this proposal, most likely due to normalization of i-vectors. The i-vectors should be normalized very carefully, which is also observed by:

  A. Senior, I. Lopez-Moreno. Improving DNN speaker independence with i-vector inputs. ICASSP 2014.

- When using our SAT-DNN, no need to worry about i-vector normalization. The feature function will do this job!
Experiments -- Switchboard

- A 110-Hour training setup [7] = 100k utterances
  - Kaldi for GMM: mono $\rightarrow$ delta $\rightarrow$ lda+mllt $\rightarrow$ sat
  - Kaldi+PDNN: [http://www.cs.cmu.edu/~ymiao/kaldipdnn.html](http://www.cs.cmu.edu/~ymiao/kaldipdnn.html)
  - Two types of DNN inputs: SI filterbanks and SA fMLLRs
- Tested on the SWBD part of HUB’00

I-Vector Extractor Building

- Open-source ALIZE toolkit [8]
- A **100-dimensional** i-vector is extracted for each training and testing speaker
## Experiments -- Switchboard

<table>
<thead>
<tr>
<th>Models</th>
<th>Filterbank</th>
<th>fMLLR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline (initial) DNN</td>
<td>21.4</td>
<td>19.9</td>
</tr>
<tr>
<td>SAT-DNN + AdaptNN</td>
<td>19.8 ↓7.5%</td>
<td>18.7 ↓6.0%</td>
</tr>
<tr>
<td>SAT-DNN + iVecNN</td>
<td>19.9 ↓7.0%</td>
<td>19.0 ↓4.8%</td>
</tr>
<tr>
<td>Initial DNN + AdaptNN</td>
<td>20.8 (2.8%)</td>
<td>19.2 (3.5%)</td>
</tr>
<tr>
<td>Initial DNN + iVecNN</td>
<td>21.2 (0.9%)</td>
<td>19.7 (1.0%)</td>
</tr>
</tbody>
</table>
## Experiments -- Switchboard

<table>
<thead>
<tr>
<th>Models</th>
<th>Filterbank</th>
<th>fMLLR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline (initial) DNN</td>
<td>21.4</td>
<td>19.9</td>
</tr>
<tr>
<td>SAT-DNN + AdaptNN</td>
<td>19.8 ↓7.5%</td>
<td>18.7 ↓6.0%</td>
</tr>
<tr>
<td>SAT-DNN + iVecNN</td>
<td>19.9 ↓7.0%</td>
<td>19.0 ↓4.8%</td>
</tr>
<tr>
<td>Initial DNN + AdaptNN</td>
<td>20.8 (2.8%)</td>
<td>19.2 (3.5%)</td>
</tr>
<tr>
<td>Initial DNN + iVecNN</td>
<td>21.2 (0.9%)</td>
<td>19.7 (1.0%)</td>
</tr>
</tbody>
</table>

Our recent work enlarges the improvement to **11.1% and 6.8%** relatively on Filterbank and fMLLR respectively
More challenging BABEL dataset

- Conversational telephone speech from low-resource languages
- 80 hours of training data for each language
- Tagalog (IARPA-babel106-v0.2f) and Turkish (IARPA-babel105b-v0.4); only on the SI filterbank features

<table>
<thead>
<tr>
<th>Models</th>
<th>Tagalog</th>
<th>Turkish</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline (initial) DNN</td>
<td>49.3</td>
<td>51.3</td>
</tr>
<tr>
<td>SAT-DNN + AdaptNN</td>
<td>47.1</td>
<td>48.6</td>
</tr>
<tr>
<td>SAT-DNN + iVecNN</td>
<td>47.3</td>
<td>49.3</td>
</tr>
</tbody>
</table>
Summary & Future Work

Summary

- We can do SAT for DNNs! To achieve this, we propose two feature learning approaches to get the speaker-normalized space.
- We get nice improvement! Our experiments show SAT-DNN outperforms DNNs regardless of the feature types of the DNN inputs.
- Our code is open source! You can check out the code and run the experiments.

Future Work

- Comparison with speaker adaptation methods; perform sequence training [9] over the resulting SAT-DNN.
- Extend the SAT framework to other architectures, e.g., to bottleneck feature extraction [10] and convolutional neural networks [6].
References

Acknowledgements. This work was supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Defense U.S. Army Research Laboratory (DoD / ARL) contract number W911NF-12-C-0015. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoD/ARL, or the U.S. Government.