BLHUC: BAYESIAN LEARNING OF HIDDEN UNIT CONTRIBUTIONS FOR DEEP NEURAL NETWORK SPEAKER ADAPTATION

Xurong Xie\textsuperscript{1,2}, Xunying Liu\textsuperscript{1,2}, Tan Lee\textsuperscript{1}, Shoukang Hu\textsuperscript{1}, Lan Wang\textsuperscript{2}

\textsuperscript{1}Chinese University of Hong Kong
\textsuperscript{2}Shenzhen Institutes of Advanced Technology,
Chinese Academy of Sciences
Introduction

• DNN based speaker adaptation
  • Feature based: i-vector, speaker code, LDA
  • Model based: linear transform, CAT (basis interpolation), LHUC

• Learning hidden unit contributions (LHUC) learns
  • Contributions of DNN hidden outputs using speaker-dependent (SD) scaling vectors
  • Deterministic parameters
  • Limited amount of adaptation data leads to over-fitting and poor generalization
Contributions of the work

• **Bayesian Learning** of hidden unit contributions (BLHUC)
  • Addressing SD parameter **uncertainty** in standard LHUC
  • Posterior distribution over the LHUC scaling vector is used
  • Variational inference and sampling based approach for estimating posterior parameters

• Two experiment setups to evaluate BLHUC
  • Unsupervised test time speaker adaptation
  • Speaker adaptive training (SAT)

• To the best of our knowledge, this is the first work on using Bayesian learning for DNN speaker adaptation
Learning hidden unit contributions (LHUC)

- Scaling vectors used in element-wise multiplication to modify the DNN hidden node outputs for each speaker
  \[ h^{l,s} = \xi(r^s) \otimes \psi(W^T h^{l-1,s} + b) \]

- where \( \xi(r^s) \) is the scaling vector parameterized by \( r^s \)
- \( r^s \) encodes speaker information
- \( \xi(\cdot) = 2\text{sigmoid}(\cdot) \)
Learning hidden unit contributions (LHUC)

• By using LHUC technique, the inference for input feature $o_t^s$ given adaptation data $o^s$ and its alignment $c^s$ is

$$P(c_t^s|o_t^s, o^s, c^s) = \int P(c_t^s|o_t^s, r^s)p(r^s|o^s, c^s)dr^s$$

$$\approx P(c_t^s|o_t^s, \hat{r}^s)$$

• $\hat{r}^s = \arg \max_{r^s} P(r^s|o^s, c^s)$ is the deterministic parameter estimate of $r^s$

• Assuming we are very confident that this deterministic estimate is reliable

• $r^s$ is often of high dimension in practice, and adaptation data is limited

• Parameter uncertainty leads to overfitting and poor generalization
Bayesian learning of hidden unit contributions (BLHUC)

- From deterministic to **probabilistic** estimate of SD parameter $r^s$
- Parameter posterior handles uncertainty

$$P(c_t^s | o_t^s, o^s, c^s) = \int P(c_t^s | o_t^s, r^s)p(r^s | o^s, c^s)dr^s$$

- Parameter posterior to be learnt
- Integral non-trivial to compute
- Back-propagation algorithm not directly usable

- Two tricks:
  - Variational lower bound
  - Parameter sampling
Variational estimation for BLHUC parameters

• The lower bound of cross entropy loss on adaptation data is

\[
\text{Loss} = - \log P(c^s|o^s) \\
= - \log \int P(c^s|o^s, r^s) p(r^s) dr^s \\
\leq -\int q_s(r^s) \log P(c^s|o^s, r^s) dr^s + KL(q_s || p)
\]

• where \( KL(q_s || p) = \int q_s(r^s) \log \frac{q_s(r^s)}{p(r^s)} dr^s \) is the KL divergence

• Variational distribution \( q_s(r^s) \) approximates posterior \( p(r^s|o^s, c^s) \)
• Assumed to be Gaussian – to be learnt
Variational estimation for BLHUC parameters

- Both $q_s(r^s)$ and prior $p(r^s)$ are assumed to be Gaussian for simplification
  - $q_s(r^s_d) = N(r^s_d; \mu_{s,d}, \sigma_{s,d}^2)$
  - $p(r^s_d) = N(r^s_d; \mu_{0,d}, \sigma_{0,d}^2)$

Then, the KL divergence can be exactly calculated by

$$KL(q_s || p) = \frac{1}{2} \sum_{d=1}^{D} \left\{ \frac{\left(\mu_{s,d} - \mu_{0,d}\right)^2 + \sigma_{s,d}^2}{\sigma_{0,d}^2} - \log \frac{\sigma_{s,d}^2}{\sigma_{0,d}^2} - 1 \right\}$$

- Hyper parameters of both $p$ and $q_s$ are updatable

- But non-trivial to compute $\int q_s(r^s) \log P(c^s | o^s, r^s) dr^s$ – parameter sampling
Variational estimation for BLHUC parameters

• The BLHUC scaling vector posterior can be parameterized by

\[ \theta_s^B = \{ \mu_s, \gamma_s \} \]

• where \( \sigma_s = \exp \gamma_s \)

• \( \theta_s^B \) in the integral term of CE is not directly differentiable and updatable

• Re-parameterization used in sampling over \( \theta_s^B \)

\[
\int q_s(r^s) \log P(c^s | o^s, r^s)dr^s \\
= \int \mathcal{N}(\epsilon; 0, I) \log P(c^s | o^s, \mu_s + \exp(\gamma_s) \otimes \epsilon) d\epsilon \\
\approx \frac{1}{J} \sum_{j=1}^{J} \log P(c^s | o^s, \theta_s^B, \epsilon_j)
\]

• where \( \epsilon_j \) is the \( j \)th Monte Carlo sample drawn from \( N(0,1) \)
Variational estimation for BLHUC parameters

• Then, the gradient of $\theta_s^B$ in one data batch can be computed by

$$
\frac{\partial \text{Loss}_m}{\partial \theta_s^B} \approx \alpha \left\{ - \frac{1}{J} \sum_{j=1}^{J} \frac{\partial \log P(c_m^s|o_m^s, \theta_s^B, \epsilon_j)}{\partial \theta_s^B} + \frac{N_{m,s}}{N_s} \frac{\partial \text{KL}(q_s||p)}{\partial \theta_s^B} \right\}
$$

• To be used in back-propagation for estimation of $\theta_s^B$

• $\alpha = \frac{N_s}{N_{m,s}}$ can be absorbed by the learning rate

• The coefficient $\frac{N_{m,s}}{N_s}$ adjusts the weight of KL regularization term
Variational estimation for BLHUC parameters

• We set the sampling number by $J = 1$ during adaptation for efficiency

• Then, the resulting gradient is closely related to DNN adaptation using KL-divergence regularization (Yu, Yao, Su, Li & Seide 2013, “KL-divergence regularized deep neural network adaptation for improved large vocabulary speech recognition”)

• But with additional parameter uncertainty modeled in first term of variational lower bound

$$\frac{\partial \text{Loss}_m}{\partial \theta^B_s} \approx \alpha \left \{ -\frac{1}{J} \sum_{j=1}^{J} \frac{\partial \log P(c^S_m | o^S_m, \theta^B_s, \epsilon_j)}{\partial \theta^B_s} + \frac{N_{m,s}}{N_s} \frac{\partial K L(q_s || p)}{\partial \theta^B_s} \right \}$$

* The gradient form of standard LHUC using $\epsilon_j$
Inference for BLHUC in decoding

• Inference can be directly approximated by Monte Carlo sampling in the test stage

\[ p(c^s_t | o^s_t, o^s, c^s) = \int P(c^s_t | o^s_t, r^s) p(r^s | o^s, c^s) dr^s \approx \frac{1}{j} \sum_{j=1}^{J} P(c^s_t | o^s_t, r^s_j) \]

• where \( r^s_j \sim p(r^s | o^s, c^s) \approx q_s(r^s) \)

• A more efficient approximation (used in the paper) is using the mean of the posterior (Normal distribution)

\[ \int P(c^s_t | o^s_t, r^s) p(r^s | o^s, c^s) dr^s \approx P(c^s_t | o^s_t, \mathbb{E}[r^s | o^s, c^s]) = P(c^s_t | o^s_t, \mu_s) \]
Different adaptation setups

• Test time adaptation only
  • Standard LHUC estimation
    • Deterministic estimation on adaptation data (Swietojanski & Renals 2016 “Learning hidden unit contributions for unsupervised acoustic model adaptation”)
  • BLHUC estimation
    • SI prior can be separately estimated by training data
    • SI prior can also be zero mean and unit variance for convenience (used in the paper)

• Speaker adaptive training (SAT)
  • Standard LHUC training + standard LHUC test time adaptation
  • Standard LHUC training + BLHUC test time adaptation
    • SI prior mean and variance are computed over training speakers’ LHUC vectors
  • BLHUC training + BLHUC test time adaptation
    • SI prior is updated during training
Experiment setup

• 300 hrs SWBD setup
• Hub5’ 00 for test (SWBD test set + CallHome test set)

• HMMs: 8929 states
• DNN setting
  • Input: 9 successive frames
  • Hidden layer: 2000 nodes, 6 layers, sigmoid
  • Output: 8929 nodes, softmax

• LM: 4-gram, 30,000 words, Fisher + SWBD training

• Features: 80 dimensional f-bank + delta

• Implemented on modified version of Kaldi toolkit and HTK
Result of test time adaptation

• Using all test data as adaptation data
• The BLHUC adapted systems significantly outperformed both the SI baseline system and standard LHUC adapted CE and MPE systems

<table>
<thead>
<tr>
<th>DNN criterion</th>
<th>Test adapt</th>
<th>WER (%)</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>SWBD</td>
<td>CallHome</td>
<td></td>
</tr>
<tr>
<td>CE</td>
<td>-</td>
<td>15.3</td>
<td>27.6</td>
<td></td>
</tr>
<tr>
<td></td>
<td>LHUC</td>
<td>14.6</td>
<td>25.8</td>
<td></td>
</tr>
<tr>
<td></td>
<td>BLHUC</td>
<td>14.2</td>
<td>25.3</td>
<td></td>
</tr>
<tr>
<td>MPE</td>
<td>-</td>
<td>13.4</td>
<td>26.8</td>
<td></td>
</tr>
<tr>
<td></td>
<td>LHUC</td>
<td>12.8</td>
<td>24.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>BLHUC</td>
<td>12.4</td>
<td>23.1</td>
<td></td>
</tr>
</tbody>
</table>

△ 0.5
△ 0.9
Using different amount of adaptation data for CE systems

- On SWBD test set
- BLHUC adapted systems consistently achieved the best performance using different adaptation data

![Graph showing WER with respect to adaptation data amount on SWBD test set. The graph indicates a decreasing trend in WER as the amount of adaptation data increases. The BLHUC adapted system consistently outperforms other systems, with a range of improvement from 0.3 to 0.5.]
Using different amount of adaptation data for CE systems

• on CallHome test set
• BLHUC adapted systems consistently achieved the best performance using different adaptation data

![Graph showing WER w.r.t adaptation data amount on CallHome test set.](image)
Using different amount of adaptation data for MPE systems

• On SWBD test set
• BLHUC adapted systems consistently achieved the best performance using different adaptation data
Using different amount of adaptation data for MPE systems

- On the harder CallHome test set, BLHUC adaptation obtained significantly improvement by even using only one utterance (2 seconds on average).

![Graph showing WER w.r.t adaptation data amount on CallHome test set](image)

- DNN MPE +BLHUC
- DNN MPE +LHUC
- DNN MPE

WER (%)

- 27.0
- 26.0
- 25.0
- 24.0
- 23.0

Adaptation data amount

- 1 utt (2 sec.)
- 10% (15 sec.)
- 25% (33 sec.)
- 50% (70 sec.)
- 100% (140 sec.)

0.9~1.1
Result of SAT

- Using **all test data** as adaptation data for CE systems
- Using BLHUC for both training and testing achieved the best performance

<table>
<thead>
<tr>
<th>DNN criterion</th>
<th>SAT</th>
<th>Test adapt</th>
<th>WER (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>SWBD</td>
</tr>
<tr>
<td>CE</td>
<td>-</td>
<td>-</td>
<td>15.3</td>
</tr>
<tr>
<td></td>
<td>LHUC</td>
<td>LHUC</td>
<td>13.2</td>
</tr>
<tr>
<td></td>
<td>LHUC</td>
<td>BLHUC</td>
<td>13.0</td>
</tr>
<tr>
<td></td>
<td>BLHUC</td>
<td>BLHUC</td>
<td><strong>12.8</strong></td>
</tr>
</tbody>
</table>

0.6
Conclusion

• **Bayesian learning** of the hidden unit contribution for DNN based speaker adaptation is proposed in the work

• An **efficient variational approximation** for learning LHUC parameter posterior

• BLHUC adaptation consistently outperformed the standard LHUC adaptation, especially on the harder CallHome data set and using limited amount of adaptation data (as minimum as 2 sec of speech)

• To the best of our knowledge, this is the first work on using Bayesian learning for DNN speaker adaptation

• Future work: Bayesian learning of other adaptation techniques