The MERL/MELCO/TUM system for the REVERB Challenge using Deep Recurrent Neural Network Feature Enhancement

Felix Weninger$^{1,2}$, Shinji Watanabe$^1$, Jonathan Le Roux$^1$, John R. Hershey$^1$, Yuuki Tachioka$^3$, Jürgen Geiger$^2$, Björn Schuller$^2$, Gerhard Rigoll$^2$

$^1$ Mitsubishi Electric Research Laboratories (MERL), Cambridge, MA, USA
$^2$ MMK, Technische Universität München, Munich, Germany
$^3$ IT R&D Center, Mitsubishi Electric Corporation, Kamakura, Japan

Florence, Italy
May 10, 2014
Motivation

• Deep recurrent neural network (DRNN) feature enhancement: promising for reverberated ASR

• Potential performance improvement by additional:
  • Discriminative GMM training
  • DRNN acoustic modeling
  • Integration of multi- and single-channel enhancement

F. Weninger et al., Deep Recurrent De-Noising Auto-Encoder and Blind De-Reverberation for Reverberated Speech Recognition, ICASSP 2014

Y. Tachioka et al., Effectiveness of discriminative training for recognition of reverberated and noisy speech, ICASSP 2013

J. Geiger et al., Memory-Enhanced Recurrent Neural Networks and NMF for Robust ASR, T-ASLP 2014
System Overview

- Cascade multi- and single-channel enhancement
- DRNN always sees single-channel input
- Multi-stream HMM decoding
  - Cf. CHiME Challenge  (Geiger et al., T-ASLP, 2014)
Multi-Channel Processing

• Cross-spectrum phase (CSP) + delay-and-sum (DS) beam-forming in the spectral domain

\[ \tau_{1,m} = \arg \max S^{-1} \left[ \frac{z_t(1) \odot z_t(m)^*}{|z_t(1)||z_t(m)|} \right] \]

\[ \hat{z}_t = \sum_m z_t(m) \odot \exp(-j\omega \tau_{1,m}) \]

• Peak-hold process
• Noise component suppression
Single-channel DRNN-DAE enhancement

• Enhancement by de-noising auto-encoder (DAE)
  – Supervised training of mapping from reverberated and noisy to clean speech features (Log Mel)
  – Trained on simulated parallel data – does it generalize?

• Implement DAE as deep recurrent neural network (RNN) with Long Short-Term Memory (LSTM) architecture

• Successful in ASR feature enhancement task
  – Outperforms DNN on CHiME

• LSTM-RNN:
  – Adaptive context size
  – Models output dynamics

(Weninger et al., CSL, 2014)
LSTM de-reverberation

• Can learn long-term dependencies without blowing up input layer → More concise model
• Context size depends on history → useful for varying acoustic conditions

Noisy + reverberated features

Matrices obtained from supervised training

Compute input / forget gate activation based on feed-forward and recurrent part

Update cell state

Estimated clean speech features

Output cell state to hidden activation

\[
\begin{align*}
  h_t^{(0)} &:= x_t, \\
  f_t^{(n)} &:= \sigma(W_f^{(n)} [h_t^{(n-1)}; h_{t-1}^{(n)}; c_{t-1}^{(n)}; 1]) \\
  i_t^{(n)} &:= \sigma(W_i^{(n)} [h_t^{(n-1)}; h_{t-1}^{(n)}; c_{t-1}^{(n)}; 1]) \\
  c_t^{(n)} &:= f_t^{(n)} \otimes c_{t-1}^{(n)} \\
  &\quad + i_t^{(n)} \otimes \tanh(W_c^{(n)} [h_t^{(n-1)}; h_{t-1}^{(n)}; 1]), \\
  o_t^{(n)} &:= \sigma(W_o^{(n)} [h_t^{(n-1)}; h_{t-1}^{(n)}; c_t^{(n)}; 1]) \\
  h_t^{(n)} &:= o_t^{(n)} \otimes \tanh(c_t^{(n)}), \\
  \hat{y}_t &:= W^{(N+1)} [h_t^{(N)}; 1].
\end{align*}
\]
DAE training

- Training tasks:
  - 1-channel system: Map REVERB multi-condition training set to WSJCAM0 clean training set
  - 8-channel system: Map CSP+DS processed REVERB multi-condition training set to WSJCAM0 clean training set

- Dimension:
  - 1-channel: 3 bidirectional LSTM layers w/ 128 units
  - 8-channel: 2 bidirectional LSTM layers w/ 128 units

- Stochastic gradient descent with momentum and input noise

- Parallel GPU training in mini-batch learning
  - CURRENNT toolkit (http://currennt.sf.net)
Baseline recognizer

• ASR features:
  • 23 Mel filterbank outputs
  • 13 MFCCs (0-12)
  • Mean normalized Log Mel features → gain-independent

• Re-implemented REVERB HTK baseline in Kaldi toolkit

• Improvements:
  • LDA-SC (MLLT) instead of Δ+ ΔΔ
    • Feature-level context
  • Basis fMLLR adaptation *per utterance*
    • Similar or better performance than fMLLR with less adaptation data
Baseline improvements (2)

• Discriminative training of GMM-HMM
  • Boosted MMI criterion:
    \[
    f_b(\lambda) = \log \sum_u \frac{p(X^u | \lambda, h^*_{wu})^\alpha p_L(w^*_u)}{\sum_{w_u} p(X^u | \lambda, h^*_{wu})^\alpha p_L(w_u) e^{-b_o(w_u, w^*_u)}}
    \]

• Tri-gram language model

• Minimum Bayes Risk (MBR) decoding
  • Don’t choose hypothesis far from the N-best
  • Minimize expected WER instead of SER (in case of MAP)
DRNN acoustic modeling

\[ b_t \]

\[ \tilde{y}_t \approx f(y_t) \]

\[ \tilde{x}_t = f(x_t) \]

Estimated phoneme posteriors

DAE pre-training
Multi-Stream DRNN+GMM-HMM

- Tandem decoding approach
- Discrete DRNN phoneme prediction:

\[ b_t = \arg \max_i \tilde{y}_{t,i} \]

- Multi-stream emission probability:

\[ p(x_t, b_t | s_t) = p(x_t | s_t)^\mu p(b_t | s_t)^{2-\mu} \]

- Stream weight \( \mu \) for GMM likelihood of acoustic feature vector \( x_t \)
- DRNN phoneme confusions modeled by \( p(b_t | s_t) \)
## Baseline ASR results

<table>
<thead>
<tr>
<th></th>
<th>SimData</th>
<th>RealData</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>REVERB baselines (HTK)</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Clean</td>
<td>51.86</td>
<td>88.38</td>
</tr>
<tr>
<td>Multi-condition</td>
<td>28.94</td>
<td>52.29</td>
</tr>
<tr>
<td>fMLLR</td>
<td>25.16</td>
<td>47.23</td>
</tr>
<tr>
<td><strong>Our baselines (Kaldi)</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Clean</td>
<td>51.23</td>
<td>88.81</td>
</tr>
<tr>
<td>Multi-condition</td>
<td>28.62</td>
<td>54.04</td>
</tr>
<tr>
<td>Basis fMLLR</td>
<td>23.60</td>
<td>47.14</td>
</tr>
</tbody>
</table>
Baseline ASR results (2)

<table>
<thead>
<tr>
<th></th>
<th>SIMData</th>
<th>REALData</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Our baselines (Kaldi)</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Clean</td>
<td>51.23</td>
<td>88.81</td>
</tr>
<tr>
<td>Multi-condition</td>
<td>28.62</td>
<td>54.04</td>
</tr>
<tr>
<td>Basis fMLLR</td>
<td>23.60</td>
<td>47.14</td>
</tr>
<tr>
<td>+LDA-STC</td>
<td>19.42</td>
<td>41.42</td>
</tr>
<tr>
<td>+DT</td>
<td>15.53</td>
<td>40.60</td>
</tr>
<tr>
<td>+Tri-gram</td>
<td>12.28</td>
<td>31.05</td>
</tr>
<tr>
<td>+MBR</td>
<td>12.05</td>
<td>30.73</td>
</tr>
</tbody>
</table>

Kaldi recipe available on REVERB homepage
DRNN enhancement training epochs

Clean recognizer, LDA-STC, ML trained, Trigram
Base: $43.4 \% / 89.6 \%$

Input: 1st channel

- Drastic improvement over noisy baseline
- More effective than MCT without front-end processing ($23 \% / 48 \%$)
- Fast convergence esp. on REALDATA

5/10/14
Felix Weninger - MERL/MELCO/TUM system
DRNN enhancement training epochs

Clean recognizer, LDA-STC, ML trained, Trigram
Base: **24.9 / 72.2**

Input: CSP+DS (Channels 1-8)

- Even faster convergence ...
- Mismatch by beam-forming alleviated
Enhancement results: Clean training w/ fMLLR adaptation

<table>
<thead>
<tr>
<th># channels</th>
<th>DRNN enh.?</th>
<th>SIMData</th>
<th>REALData</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>✗</td>
<td>33.2</td>
<td>77.8</td>
</tr>
<tr>
<td>1</td>
<td>✓</td>
<td>14.0</td>
<td>35.0</td>
</tr>
<tr>
<td>8</td>
<td>✗</td>
<td>16.4</td>
<td>54.5</td>
</tr>
<tr>
<td>8</td>
<td>✓</td>
<td>9.7</td>
<td>26.5</td>
</tr>
<tr>
<td>Oracle</td>
<td></td>
<td>6.0</td>
<td>10.1</td>
</tr>
</tbody>
</table>

Best result without using the multi-condition set!
Enhancement results: bMMI MCT recognizer

- Tuning of search parameters
- Discriminative training (boosted MMI) with (processed) multi-condition set

<table>
<thead>
<tr>
<th># channels</th>
<th>DRNN enh.?</th>
<th>SIMDATA</th>
<th>REALDATA</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>✗</td>
<td>11.2</td>
<td>30.8</td>
</tr>
<tr>
<td>1</td>
<td>✓</td>
<td>10.4</td>
<td>26.3</td>
</tr>
<tr>
<td>8</td>
<td>✗</td>
<td>7.5</td>
<td>23.9</td>
</tr>
<tr>
<td>8</td>
<td>✓</td>
<td>7.7</td>
<td>21.4</td>
</tr>
<tr>
<td>Oracle</td>
<td></td>
<td>5.1</td>
<td>9.9</td>
</tr>
</tbody>
</table>

Best result with single-channel front-end
Test set evaluation: Enhancement, GMM-HMM AM

<table>
<thead>
<tr>
<th>WER [%]</th>
<th>SIMData</th>
<th>REALData</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1-channel systems</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>REVERB baseline</td>
<td>25.3</td>
<td>49.2</td>
</tr>
<tr>
<td>GMM-HMM</td>
<td>11.7</td>
<td>30.9</td>
</tr>
<tr>
<td>+ DRNN enh.</td>
<td>10.2</td>
<td>26.7</td>
</tr>
<tr>
<td><strong>8-channel system</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+ CSP-DS</td>
<td>7.8</td>
<td>20.1</td>
</tr>
</tbody>
</table>
Test set evaluation:
DRNN+GMM-HMM AM

<table>
<thead>
<tr>
<th>Model</th>
<th>SIMData</th>
<th>REALData</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRNN+GMM-HMM</td>
<td>7.28</td>
<td>21.69</td>
</tr>
<tr>
<td>GMM-HMM w/ DRNN enh.</td>
<td>7.75</td>
<td>20.09</td>
</tr>
<tr>
<td>ROVER</td>
<td>7.02</td>
<td>19.61</td>
</tr>
<tr>
<td>GMM-HMM w/ Oracle enh.</td>
<td>5.65</td>
<td>8.47</td>
</tr>
</tbody>
</table>
Results with GMM-HMM and DRNN enhancement by room

![WER on et (%)](chart)

- R1
- R2
- R3
- Real

- Near
- Far
- Oracle
Conclusions and Outlook

• Supervised training of de-reverberation with RNN is effective for ASR
  • Works on real data
  • Particularly promising for single-channel scenario
  • Can be efficiently combined with beam-forming
  • Some over-fitting observed (less than RNN-AM)

• Future work:
  • Effectiveness of supervised training for multi-channel de-reverberation
  • Use phase information
Thank you.

felix@weninger.de