Incorporating Discriminative DPGMM Posteriorgrams for Low-resource ASR

Bin Wu\textsuperscript{1}, Sakriani Sakti\textsuperscript{1,2}, and Satoshi Nakamura\textsuperscript{1,2}

\{wu.bin.vq9,ssakti,s-nakamura\}@is.naist.jp

1. Nara Institute of Science and Technology, Japan
2. RIKEN, Center for Advanced Intelligence Project AIP, Japan
Incorporating Discriminative DPGMM Posteriorgrams for Low-resource ASR
Bin Wu, Sakriani Sakti, and Satoshi Nakamura

Speech features

• First step of ASR system: extracting proper speech feature
  • Typical speech features: MFCC and PLP
  • Effective in discriminating phonemes by extracting smooth formant envelopes
• Speech feature for ASR should have sufficient ability to discriminate phonemes (words), robust to genders, emotions, and noises.

Phonemic discriminability comparison

• Feature discriminability are compared based on ABX error rate in Zerospeech challenges
  • Acoustic features
    • MFCC and PLP
  • Neural network features
    • autoencoder, ABnet and VQVAE
  • Parametric clustering features
    • GMM and k-means
  • Nonparametric clustering features
    • DPGMM (Dirichlet Process Gaussian Mixture Model)
• DPGMM gets lowest ABX error as strongest phonemic discriminability in Zerospeech 2015, 2017 and 2019
Incorporating Discriminative DPGMM Posteriorgrams for Low-resource ASR

Bin Wu, Sakriani Sakti, and Satoshi Nakamura

Proposal

Concatenate DPGMM Posteriorgrams with MFCC features to increase phonemic discriminability to improve ASR

MFCC+DGPMM vs. MFCC on ASR

- MFCC+DGPMM better than MFCC on ASR
- More absolute improvement on smaller corpus

<table>
<thead>
<tr>
<th>Systems on WSJ train.si84 dataset (15 hrs)</th>
<th>CER (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Att Enc-Dec (Baseline ASR1) [30]</td>
<td>17.01</td>
</tr>
<tr>
<td>Att Enc-Dec (Baseline ASR2) [41]</td>
<td>17.35</td>
</tr>
<tr>
<td>Att Enc-Dec (Our ASR with MFCC)</td>
<td>16.61</td>
</tr>
<tr>
<td>Att Enc-Dec (Our ASR with MFCC + DPGMM)</td>
<td><strong>14.86</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Systems on WSJ train.si284 dataset (80 hrs)</th>
<th>CER (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Att Enc-Dec (Baseline ASR1) [30]</td>
<td>8.17</td>
</tr>
<tr>
<td>Att Enc-Dec (Baseline ASR2) [41]</td>
<td>7.12</td>
</tr>
<tr>
<td>Att Enc-Dec (Our ASR with MFCC)</td>
<td>6.57</td>
</tr>
<tr>
<td>Att Enc-Dec (Our ASR with MFCC + DPGMM)</td>
<td><strong>5.67</strong></td>
</tr>
</tbody>
</table>
Incorporating Discriminative DPGMM Posteriorgrams for Low-resource ASR

Bin Wu, Sakriani Sakti, and Satoshi Nakamura

MFCC+DGPM vs. tandem on ASR

- Compare discriminability of proposal (MFCC+DGPM) and tandem (MFCC+BNF99)
  - Tandem bottleneck features from Kaldi default script
  - Proposed feature better than tandem on smaller corpus
  - Tandem system needs the knowledge of phonemes and beginning and end time information of phonemes, which is not needed by our proposal

<table>
<thead>
<tr>
<th>Feature</th>
<th>TIMIT PER (%)</th>
<th>WSJ_si284 CER (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MFCC</td>
<td>23.92</td>
<td>6.57</td>
</tr>
<tr>
<td>MFCC + DGPM</td>
<td>22.74</td>
<td><strong>5.67</strong></td>
</tr>
<tr>
<td>MFCC + Tandem (BNF99)</td>
<td>23.22</td>
<td>5.12</td>
</tr>
</tbody>
</table>

Conclusion

- DPGMM features strong at discriminating phonemes
- Propose DPGMM + MFCC to improve the ASR
- The concatenated features work well on LVCSR, especially with fewer resources
Speech features

• First step of ASR system: extracting proper speech feature
• Typical speech features: MFCC and PLP
  • **Discriminating phonemes** by extracting smooth formant envelopes

![Formant Diagram](image-url)
Measure the phoneme discriminability

- Speech feature for ASR should have sufficient **ability to discriminate phonemes** (words) and be robust to genders, emotions, and noises.

- **Measure phoneme discriminability** of a feature by **ABX error rate**
  - Each utterance to segments of features
  - Make an ABX error: \( d(\text{seg. in same phn category}) > d(\text{seg. in different phn categories}) \)
  - ABX error rate: ratio of samples making ABX errors

\[
\text{ABXErr}(|I|, |u|) = \frac{\#\text{triplets}(d(A,B) > d(A,X))}{\#\text{triplets}}
\]

Any A, B and X triplet (three segments as an sample)
A, B from same category |I|
A, X from different categories |u|

Copyright © 2021 NAIST Japan, All rights reserved
Feature discriminability comparison

• Compare Feature discriminability based on ABX error rate in Zerospeech challenges
  • Acoustic features
    • MFCC and PLP
  • Neural network features
    • autoencoder, ABnet and VQVAE
  • Parametric clustering features
    • GMM and k-means
  • Nonparametric clustering features
    • DPGMM (Dirichlet Process Gaussian Mixture Model)

• DPGMM gets lowest ABX error rate in Zerospeech 2015, 2017 and 2019
Why DPGMM strong at discriminating phonemes

- DPGMM dynamically changes number of clusters to fit data with max probability with segmental patterns clearly discriminated
Why DPGMM strong at discriminating phonemes

• DPGMM dynamically changes number of clusters to fit data with max probability with segmental patterns clearly discriminated
Why DPGMM strong at discriminating phonemes

- DPGMM dynamically changes number of clusters to fit data with max probability with segmental patterns clearly discriminated
Why DPGMM strong at discriminating phonemes

• DPGMM dynamically changes number of clusters to fit data with max probability with segmental patterns clearly discriminated
Why DPGMM strong at discriminating phonemes

• DPGMM dynamically changes number of clusters to fit data with max probability with segmental patterns clearly discriminated
Proposal

Traditional ASR

Input speech → Feature extraction → MFCC → ASR

Concatenate MFCC and its DPGMM posteriorgram

Our proposed ASR

Input speech → Feature extraction → MFCC → DPGMM clustering → DPGMM posteriorgram → ASR
Proposal

- Proposal: Combine DPGMM features and acoustic features to improve the ASR system

**Waveform**

**Feature Extraction**

**DPGMM Clustering**

**Speech feature (MFCC)**

**DPGMM posteriorgram (DPGMM posterior vectors)**

\[
\begin{pmatrix}
\frac{1}{0} & \frac{1}{0.9} & \frac{0.2}{0}
\end{pmatrix}
\]

**Concatenated feature of MFCC and DPGMM Posteriorgram**

\[
\begin{pmatrix}
\frac{1}{0} & \frac{1}{0.9} & \frac{0.2}{0} & \frac{1}{0}
\end{pmatrix}
\]
# ASR System

<table>
<thead>
<tr>
<th>Module</th>
<th>Cascaded layers of module</th>
</tr>
</thead>
<tbody>
<tr>
<td>Encoder</td>
<td>FC-512 $\rightarrow$ ReLU $\rightarrow$ Dropout $\rightarrow$ 3-layer pBiLSTM-256 (reduce half of the frames per layer)</td>
</tr>
<tr>
<td>Decoder pre-net</td>
<td>EMBED-256 $\rightarrow$ Dropout</td>
</tr>
<tr>
<td>Decoder [31]</td>
<td>(Pre-net output + Prev. decoder output) Single-layer LSTM-512 $\rightarrow$ Dropout $\rightarrow$ Contextual FC-256 $\rightarrow$ Tanh</td>
</tr>
<tr>
<td>Decoder post-net</td>
<td>Softmax</td>
</tr>
<tr>
<td>MLP attention</td>
<td>FC-256 $\rightarrow$ Tanh</td>
</tr>
</tbody>
</table>

![Diagram](image)  

Figure 1: The graphical illustration of the proposed model trying to generate the $t$-th target word $y_t$ given a source sentence $(x_1, x_2, \ldots, x_T)$.  

15
Corpora

- TIMIT: English corpus of reading speech
  
  
<table>
<thead>
<tr>
<th>Set</th>
<th># speakers</th>
<th>#sentences</th>
<th>#hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training</td>
<td>462</td>
<td>3696</td>
<td>3.14</td>
</tr>
<tr>
<td>Core test</td>
<td>24</td>
<td>192</td>
<td>0.16</td>
</tr>
<tr>
<td>Complete test set</td>
<td>168</td>
<td>1344</td>
<td>0.81</td>
</tr>
</tbody>
</table>

  1. TIMIT Corpus training and test sets

- WSJ: Spontaneous speech for large vocabulary continuous speech recognition (LVCSR)
  
  - Training datasets
    - “train si84” (about 15 hours)
    - “train si284” (about 80 hours)
  - development dataset of “dev93”
  - evaluation dataset of “eval92”
Analysis on TIMIT test set

Phonemes clearly classified by DPGMM posteriorgrams, although not MFCC

Framewise contextual modeling creates fragmented DPGMM posteriorgrams

ASR needs concatenated feature instead of DPGMM posteriorgram alone
Performance on WSJ

Proposed feature (DPGMM+MFCC) converged faster and retained the improvement.

It improves more obviously on the system trained on the small dataset (train_si84).
## Performance on WSJ

<table>
<thead>
<tr>
<th>Systems on WSJ train_si84 dataset (15 hrs)</th>
<th>CER (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Att Enc-Dec (Baseline ASR1) [30] (Esptnet, 2017)</td>
<td>17.01</td>
</tr>
<tr>
<td>Att Enc-Dec (Baseline ASR2) [41] (Speech Chain, 2020)</td>
<td>17.35</td>
</tr>
<tr>
<td>Att Enc-Dec (Our ASR with MFCC)</td>
<td>16.61</td>
</tr>
<tr>
<td>Att Enc-Dec (Our ASR with MFCC + DPGMM)</td>
<td>14.86</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Systems on WSJ train_si284 dataset (80 hrs)</th>
<th>CER (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Att Enc-Dec (Baseline ASR1) [30]</td>
<td>8.17</td>
</tr>
<tr>
<td>Att Enc-Dec (Baseline ASR2) [41]</td>
<td>7.12</td>
</tr>
<tr>
<td>Att Enc-Dec (Our ASR with MFCC)</td>
<td>6.57</td>
</tr>
<tr>
<td>Att Enc-Dec (Our ASR with MFCC + DPGMM)</td>
<td>5.67</td>
</tr>
</tbody>
</table>

Absolute improvement (%)
- 1.75
- 0.9
Compare with tandem features

• Compare our proposed with Kaldi default tandem features
  • On a relatively small dataset TIMIT (~3h training), our method had better performance on TIMIT corpus
  • On a bigger dataset of WSJ SI-284 (~80h training), tandem system got slightly better performance
  • Tandem bottleneck features need the knowledge of phonemes and force alignment information

<table>
<thead>
<tr>
<th>Feature</th>
<th>TIMIT PER (%)</th>
<th>WSJ_si284 CER (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MFCC</td>
<td>23.92</td>
<td>6.57</td>
</tr>
<tr>
<td>MFCC + DPGMM</td>
<td>22.74</td>
<td>5.67</td>
</tr>
<tr>
<td>MFCC + Tandem (BNF99)</td>
<td>23.22</td>
<td>5.12</td>
</tr>
</tbody>
</table>
Conclusion

• DPGMM features strong at discriminating phonemes
• DPGMM + MFCC to improve the ASR
• The concatenated feature works well on LVCSR, especially with fewer resources