Analyzing Hidden Representations in End-to-End Automatic Speech Recognition Systems

Leda Sarı

January 31, 2019

\(^1\)Belinkov and Glass, NIPS 2017
Introduction

- End-to-End (E2E) directly maps acoustic features to symbol (character or word) sequences
  - Connectionist temporal classification (CTC) ✓
  - Sequence-to-sequence learning (seq2seq)
- **Question**: If and to what extent E2E models implicitly learn phonetic representations internally
- **Goal**: Make interpretations of the hidden layer activations in an E2E ASR system
  Use a pretrained model $\Rightarrow$ get frame level features $\Rightarrow$ evaluate representations and compare layers
E2E ASR Model

- It is based on DeepSpeech2 architecture (CNN and RNN layers)
- Maps acoustics to character sequence using CTC
- Inputs are spectrograms
- If \( x \) is the input spectrogram, evaluate \( \text{ASR}_k^t(x) \) output of the \( k \)-th layer at the \( t \)-th input frame
- Trained on LibriSpeech with PyTorch implementation of Baidu DeepSpeech2 model

<table>
<thead>
<tr>
<th>Layer</th>
<th>Type</th>
<th>Input Size</th>
<th>Output Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>cnn1</td>
<td>161</td>
<td>1952</td>
</tr>
<tr>
<td>2</td>
<td>cnn2</td>
<td>1952</td>
<td>1312</td>
</tr>
<tr>
<td>3</td>
<td>rnn1</td>
<td>1312</td>
<td>1760</td>
</tr>
<tr>
<td>4</td>
<td>rnn2</td>
<td>1760</td>
<td>1760</td>
</tr>
<tr>
<td>5</td>
<td>rnn3</td>
<td>1760</td>
<td>1760</td>
</tr>
<tr>
<td>6</td>
<td>rnn4</td>
<td>1760</td>
<td>1760</td>
</tr>
<tr>
<td>7</td>
<td>rnn5</td>
<td>1760</td>
<td>1760</td>
</tr>
<tr>
<td>8</td>
<td>rnn6</td>
<td>1760</td>
<td>1760</td>
</tr>
<tr>
<td>9</td>
<td>rnn7</td>
<td>1760</td>
<td>1760</td>
</tr>
<tr>
<td>10</td>
<td>fc</td>
<td>1760</td>
<td>29</td>
</tr>
</tbody>
</table>

Table 1: The ASR models used in this work.

**Figure:** ASR Network architecture

---

Belinkov and Glass, NIPS 2017
Phoneme Classifier

- **Input**: Features from different layers of the DeepSpeech2
- **Output**: Phoneme label
- Single hidden layer with ReLU nonlinearity
- Kept simple because the goal is to evaluate the features not achieving the best phoneme recognition
- Phoneme recognition is performed on TIMIT
Results - Phoneme Classification Accuracy

- Top layers of the deeper model focus more on modeling character sequences
- Stride effects time resolution ⇒ better frame accuracy
Results - Clustering

- k-means (k=500) cluster layer activations
- Plot the cluster centers using t-SNE
- Assign phone label based on majority voting

Figure 3: Centroids of frame representation clusters using features from different layers.
Sound Classes

- Coarse classes: affricates, fricatives, nasals, semivowels, stops and vowels
- Train the classifier to predict these classes
- Better classification accuracy as compared to phonemes
- Class based comparison between rnn5 and the input layer
  - rnn5 is better at distinguishing between different nasals
  - Affricates are better predicted at rnn5

Figure 4: Accuracy of classification into sound classes using representations from different layers of DeepSpeech2.
Sound Classes - Confusions

Maximum confusions are between:

1. semivowels/vowels
2. affricates/stops
3. affricates/fricatives

Figure 6: Confusion matrices of sound class classification using representations from different layers.
Summary

1. Empirically evaluate the quality of hidden representations with phoneme classification
2. First CNN better represents the phonetic information than the 2nd CNN layer
3. After certain number of RNN layers, accuracy drops \(\rightarrow\) top layers do not preserve all the phonetic information
4. Relatively similar coarse classes are confused more