Deep learning for automatic pronunciation assessment of spontaneous non-native speech based on phone distances

K. Kyriakopoulos, K.M. Knill, M.J.F. Gales
{kk492,kate.knill,mjfg}@eng.cam.ac.uk

ALTA Institute / Department of Engineering, University of Cambridge

1. Introduction

Automatic assessment: How bad is speaker’s pronunciation?
Feedback: How is speaker’s pronunciation bad?
  ▶ Individual mispronunciations
  ▶ Overall problem phones

Motivation:
  ▶ Computer assisted language learning (CALL)
  ▶ Auto-marking of spoken examinations
  ▶ Features should be predictive of grade and interpretable
  ▶ Features projected to grade through feed-forward neural network
  ▶ Extraction and grading can be separate or combined

First steps:
1. Pass audio through ASR
2. Viterbi align to get label and boundaries for each phone instance
3. Extract PLP feature vector $x_i$ for each frame $t$ of audio within each $i$

Constraints on features:
  ▶ Unstructured, spontaneous speech
  ▶ High ASR work (and phone) error rate (c. 40%)
  ▶ No native models with identical text
  ▶ Broad not narrow transcription
  ▶ Variability in speaker attributes

2. Model-based phone distance features (baseline)

  ▶ Each phone characterised relative to others
  ▶ Phone-to-phone distances act as features
  ▶ Train Gaussian model $\mathcal{N}(\mu_i, \Sigma_i)$ for all instances $i$ of each phone $\phi$
  ▶ Features are symmetric K-L divergences between pairs of models:
    $D_{\phi, \psi} = \frac{1}{2}(\mathcal{KL}(\mathcal{N}(\mu_\phi, \Sigma_\phi) || \mathcal{N}(\mu_\psi, \Sigma_\psi)) + \mathcal{KL}(\mathcal{N}(\mu_\psi, \Sigma_\psi) || \mathcal{N}(\mu_\phi, \Sigma_\phi))$

3. Deep representation of phone instances

  ▶ Bidirectional LSTM projects sequence of frame vectors $x_i$:
    $h_i^{(f)} = f(x_i^{(f)}, h_{i-1}^{(f)}, \lambda^{(f)})$
    $h_i^{(b)} = f(x_i^{(b)}, h_{i+1}^{(b)}, \lambda^{(b)})$
  ▶ Three different ways of getting instance vector $h_i$ from $h_i^{(f)}$ and $h_i^{(b)}$:
    ![Diagram of three different ways of getting instance vector $h_i$ from $h_i^{(f)}$ and $h_i^{(b)}$]

    ▶ Standard LSTM (left): Projects vector from last frame of each pass
    ▶ Centre frame method (mid): Uses middle frame of each pass
    ▶ Attention (right): Attention mechanism determines salience of each frame

4. Siamese network distance metric

  ▶ Project instances $i$ and $j$ (of phones $\phi$ and $\psi$) from same speaker to vectors $h_i^{(f)}$ and $h_j^{(f)}$, then obtain distance $d_{ij} = ||h_i^{(f)} - h_j^{(f)}||_2$

    ![Diagram of Siamese network distance metric]

  K-L training (left): Distance $d_{ij}$ directly predicts model-based K-L distance $D(\phi, \psi)$ for that speaker
  Binary training (right): Distance $d_{ij}$ passed through sigmoid to predict:
    $c_{ij} = \begin{cases} 1, & \phi = \psi \\ 0, & \phi \neq \psi \end{cases}$

    (i.e. whether the two instances are of the same phone)
  ▶ For both, train with random sample of instance pairs from each speaker

5. Predicting grade

  ▶ Bi-LSTM (trained as above) projects each instance to vector $h_i$
  ▶ With attention mechanism and end-to-end training, deep method outperforms baseline
  ▶ In both cases, set of 1081 Euclidean distances between all pairs of phones projected to predict grade
  ▶ Attention method allows feature extractors to be fine-tuned for task
  ▶ Attention weights interpretable as importance to grade of phone instance:
    $a_{ij} \in \{0, 1\}$

    ![Averaging (left): Obtain mean of vectors of all instances of each phones.](Image 1325x1261 to 2236x1840)
    ![Attention (right): Use attention mechanism to obtain weighted sum of vectors](Image 1386x798 to 2240x907)

6. Experimental Results

<table>
<thead>
<tr>
<th>Projection Criterion Combination</th>
<th>PCC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard Linen Average</td>
<td>0.698</td>
</tr>
<tr>
<td>Centre Linen Average</td>
<td>0.742</td>
</tr>
<tr>
<td>Attention Linen Average</td>
<td>0.762</td>
</tr>
<tr>
<td>Attention K-L Average</td>
<td>0.775</td>
</tr>
<tr>
<td>Attention K-L Attention</td>
<td>0.790</td>
</tr>
<tr>
<td>Baseline</td>
<td>0.785</td>
</tr>
</tbody>
</table>

  ▶ Attention LSTM performance $> \text{centre frame LSTM} > \text{standard LSTM}$
  ▶ Initialising Siamese distance using model K-L divergences improves performance over binary classification training
  ▶ With attention mechanism and end-to-end training, deep method outperforms baseline