Low Cost Lexicon

Nagendra Kumar Goel, Samuel Thomas, Pinar Akyazi
Speech Recognition

Speech → Feature Extraction → Features

Lexicon or Pronunciation Dictionary

Acoustic Model: k a t R a n

arg max \( p(W) p(x | W) \)

Language Model: \( P(w_n | w_{n-1}, w_{n-2}) \)

\[
\hat{W} = \arg \max_w P(W | X) = \arg \max_w \frac{P(W) P(X | W)}{P(X)}
\]
Cost of Developing a New Language

• Transcribed audio data
  ▫ Subspace acoustic models (UBM’s) need less data

• Text data for language modeling
  ▫ Obtain from the web if possible

• Pronunciation Lexicon
  ▫ Qualified phoneticians are expensive
  ▫ Phoneticians may make mistakes
  ▫ Conversational (callhome) English has 4.6% OOV rate for a 5K lexicon and 0.4% for a 62K lexicon
  ▫ Try to guess pronunciation given a limited lexicon and audio
Estimating Pronunciations

• Ideal Situation will be to just estimate all the pronunciations for the word that maximize the likelihood given the audio

\[ \hat{P}_{rn} = \arg \max_{P_{rn}} P(X \mid P_{rn}) \]

• There are words for which spoken audio is not available but they need to exist in the recognizer.

• Multiple pronunciations have not yet significantly improved the performance.

• This objective function needs a lot of regularization.
Estimating Pronunciation from Graphemes

- One way is to guess the pronunciation from the orthography of the word (e.g. Bisani & Ney)

- Iterative process based on grapheme/phoneme alignment

  - Start with an initial set of graphone probabilities.
  - Use the probabilities to realign graphones with phones on training data.
  - Re-estimate graphone probabilities from the alignments.

\[
P(\hat{w} | \hat{n}) = \text{arg max}_{P(n | w)} P(w, Pn) \]

\[
P(n) = \text{arg max}_{P(n | w)} P(w, Pn) \]
Training a Pronunciation Dictionary

Training

Initial Pronunciation Dictionary → G2P Training → Model for Predicting P from G

Prediction

Out of Vocabulary Words

G2P

Predicted Phoneme Sequence
G2P Plot for English

% Error

Model Context Size

% String errors
% Symbol errors
Estimating Pronunciations...

- If the audio recording is also available, that can be used to augment the estimates

\[ P\hat{r}n = \arg\max_{Prn} P(X \mid Prn)P(Prn \mid W) \]

- We use an approximation to the above

\[ P\hat{r}n = \arg\max_{Prn \in \{\text{Top 5 } Prn\}} P(X \mid Prn) \]
Estimating Pronunciations...

1. Start with a hand-made phone set and a dictionary
2. Train g2p with dictionary
3. Train acoustic models
4. Force align the training data with multiple pronunciations
5. Create new dictionary with selected pronunciation
Estimating Pronunciations...

1. Start with a hand-made phone set and a dictionary
2. Train g2p with dictionary
3. Train acoustic models
4. Force align the training data with multiple pronunciations
5. Create new dictionary with selected pronunciation
6. Match with word level transcripts
7. Pick pronunciations
8. Free Phonetic Recognition

Start with a hand-made phone set and a dictionary. Train g2p with dictionary. Train acoustic models. Force align the training data with multiple pronunciations. Create new dictionary with selected pronunciation.
Estimating Pronunciations...

Start with a hand-made phone set and a dictionary

Train g2p with dictionary

Train acoustic models

Force align the training data with multiple pronunciations

Create new dictionary with selected pronunciation

Introduce new pronunciation from unsupervised learning

Force align and create pronunciations

Pick words with high confidence

Create lattices on similar acoustic datasets

Pick pronunciations

Match with word level transcripts

Free Phonetic Recognition
Training Procedure - Bootstrapping

1000 most frequently occurring words of training data
Remaining Training data
Words used for building LM which covers the test data
Train g2p
Trained g2p model
Multiple pronunciation dictionary
Dictionary

- Callhome training lexicon size – 5 K
- LM vocabulary size – 62 K
- Training acoustic data without partial words – 6 hrs
- Complete training data – 15 hrs
Training Procedure - Bootstrapping

1000 most frequently occurring words → Train g2p

Training data → Train g2p model

Words used for building LM which covers the test data

Multiple pronunciation dictionary → Dictionary

Test data → Trained acoustic model

Train acoustic model → Recognition
Training Procedure - Building Up

1. **Train data**
2. **Force alignment**
3. **Multiple pronunciation dictionary**
4. **Best pronunciation for training words**
5. **Acoustic Models from previous iteration**
Training Procedure - Building Up

1. Train data → Force alignment
2. Acoustic Models from previous iteration → Best pronunciation for training words
3. Words used for building the LM which covers the test data → Train g2p
4. Dictionary → Acoustic model
5. Test data → Recognition
Training Procedure - Building Up

Start with a hand-made phone set and a dictionary

Train g2p with dictionary

Train acoustic models

Force align the training data with multiple pronunciations

Create new dictionary with selected pronunciation
# Results

<table>
<thead>
<tr>
<th>Results</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy with full dictionary available</td>
<td>44.35</td>
</tr>
<tr>
<td>Accuracy if 5K manual lexicon is available</td>
<td>40.53</td>
</tr>
<tr>
<td>Accuracy with 1000 words available</td>
<td>37.58</td>
</tr>
<tr>
<td>After retraining acoustic models</td>
<td>39.37</td>
</tr>
<tr>
<td>2nd iteration of g2p &amp; acoustic re-train</td>
<td>41.60</td>
</tr>
<tr>
<td>3rd iteration of g2p &amp; acoustic re-train</td>
<td>42.11</td>
</tr>
<tr>
<td>After increasing the amount of data to 15 hrs</td>
<td>43.56</td>
</tr>
</tbody>
</table>
Unsupervised Learning

1. Start with a hand-made phone set and a dictionary
2. Train \( g2p \) with dictionary
3. Train acoustic models
4. Force align the training data with multiple pronunciations
5. Create new dictionary with selected pronunciation
6. Introduce new pronunciation from unsupervised learning
7. Force align and create pronunciations
8. Pick words with high confidence
9. Create lattices on similar acoustic datasets
### Unsupervised Lexicon Learning Results

<table>
<thead>
<tr>
<th>Training Data</th>
<th>Baseline accuracy</th>
<th>After Unsupervised Learning</th>
</tr>
</thead>
<tbody>
<tr>
<td>6 Hrs of training data</td>
<td>42.11</td>
<td>42.33</td>
</tr>
<tr>
<td>15 Hrs of training data</td>
<td>43.56</td>
<td>43.44</td>
</tr>
</tbody>
</table>
WER dilemma for Spanish Callhome

- Spanish pronunciation is very graphemic.
- Accuracy for Spanish are about 31.13% (about 13% lower than callhome english).
- Phone recognition accuracy is better than callhome english:
  - English: 45.13%
  - Spanish: 53.77%
- LM Perplexity is not too bad: 127.
- Can learning alternate pronunciations of reduced words help?
Possible lexicon training paths...

- Pick pronunciations
  - Match with word level transcripts
  - Free Phonetic Recognition

- Start with a hand-made phone set and a dictionary
  - Train g2p with dictionary
    - Train acoustic models
      - Force align the training data with multiple pronunciations
        - Force align and create pronunciations
          - Create new dictionary with selected pronunciation

- Introduce new pronunciation from unsupervised learning
  - Pick words with high confidence
    - Create lattices on similar acoustic datasets

Match with word level transcripts
Lexicon Enhancement for Spanish

G2P accuracies after augmenting with phone recognition based pronunciations

% String Errors

Iteration (Model) #
Lexicon Enhancement for Spanish

G2P Plot for Spanish

% String Errors

Iteration (Model) #

Dev
Eval
English Results and Spanish Results with unconstrained phonetic recognition approach

<table>
<thead>
<tr>
<th>Language</th>
<th>Baseline</th>
<th>After adding pronunciations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spanish</td>
<td>31.13</td>
<td>30.71</td>
</tr>
<tr>
<td>English</td>
<td>43.54</td>
<td>42.71</td>
</tr>
</tbody>
</table>

- Log likelihood of training data increases with the new lexicon.
Lexicon Enhancement

• Keep the manual Lexicon but augment with most likely pronunciation in the training data
• Affected about 250 pronunciations

• Accuracy improved from 44.33 to 45.01%

• Multiple Pronunciations had no significant impact: 45.02%
Summary

• G2p based lexicon retraining method helps in achieving accuracies close to hand made lexicons
• It can also help in improving an existing lexicon
• Unsupervised lexicon learning approach and phonetic recognition based lexicon learning approaches hold promise and need to be explored with a wider variety of smoothing and pronunciation extraction scenarios
Training Procedure

• Train g2p to generate pronunciations using your best baseline lexicon

• Generate multiple pronunciations using the g2p

• Use the training data to select the best pronunciation out of these multiple choices

• Retrain the acoustic models and iterate over the above process