Modeling Pronunciation Variations for Automatic Speech Recognition with Context-dependent Acoustic Models

Qian Yang and Jean-Pierre Martens
Pronunciations and ASR

• Architecture of an ASR

- Training speech → Acoustic models → Recognizer
- Human Knowledge → Lexicon → Recognizer
- Training text → Language model → Recognizer
- Test speech → Recognized text

• Lexicon = phonetic transcriptions (1/word)
• Our goal: improve ASR by automatic design of lexicon with multiple pronunciations/word
Pronunciation variations (PV)

• Different types of pronunciation variations
  – variations due to regional accent
    change of vowel quality, etc.
  – variations due to cross-word coarticulation
    liaisons (you-w-are), (de)voicing (has to), place of articulation (in place)
  – variations due to speaker / style / region
    /n/-deletion in Dutch, data vs data, etc.

• How to model them in ASR?
  – in the acoustic models : cross-word CD phone models
  – in the lexicon : CD word pronunciation models
  – basic question : what to model where and how?
Our approach to PVM in the lexicon

1. Baseline Word Transcriptions
2. Word Pronunciation Variants
3. Compile variants into word models

- Stochastic Pronunciation Rules
- Orthographically Transcribed Training Corpus
Pronunciation Variants & Word Models

• Context-dependent word model of “either”:

```
* * *
   aI
     I
       0.3
     0.7

   D
     @
       0.6
     0.4

   r
     n **
```
Pronunciation Variants & Word Models

- Context-dependent word model of “either”:

- Integration of word models in recognizer
Learn pronunciation rules from data

- Need rule formalism and rule hierarchy (grammar)
- Need target transcriptions of training utterances
- Need method for learning stochastic rules from discrepancies between target and lexicon-based transcriptions
1. Pronunciation Rule Formalism

- **General format**
  \[ \text{LFR} \rightarrow F' \quad \text{with probability} \quad P_{\text{fir}} \]

- **Further specifications**
  - all patterns are strings of phonemes or word boundary symbols
  - F and F’ cannot contain word boundary symbols, L and R can

- **Possible extension**
  - use phonetic classes in L and R
2. Rule Hierarchy

• Two possible solutions

- top-down decision trees (1 per phoneme)
- bottom-up rule networks (1 or 1 per (F,F’))
2. Rule Hierarchy

- **Transformation-based hierarchy HT**
  - one network per transformation
  - determine all \((F,F')\) with matching \(F\)
  - apply most specific matching rule per \((F,F')\) (per focus position)

- **Condition-based hierarchy HC**
  - one network for the whole thing
  - determine most specific condition \(C\) around focus position
  - apply all rules attached to that \(C\) (per focus position)
3. Generation of Target Transcriptions

- **General principles**
  - use acoustic models to control the process
  - retain only pronunciation variations that are likely to improve the recognition

- **Different methods**
  - M1: align speech with lexicon-based utterance model
  - M2: perform phoneme recognition
  - M3: perform word + phoneme recognition
Target Transcriptions : M1

- Align speech with lexicon-based sentence model
  - concatenate lexicon transcriptions of the words
  - add insertion, deletion and substitution arcs

- insert acoustic models in the network
Target Transcriptions : M1

• **Advantage**
  – finds variations that raise likelihood of correct word sequence

• **Problems**
  – impractical in case of CD acoustic models (modification of one phoneme changes context of surrounding ones)
  – nothing known about the likelihoods of incorrect word sequences
Target Transcriptions : M2

- Create a phoneme loop
  - put all acoustic models in parallel
  - create all legal connections between models (CI vs CD)
  - put phoneme n-gram probabilities on these connections
Target Transcriptions : M2

• **Phoneme recognition**
  – most likely phoneme sequence given loop + acoustics

• **Advantage**
  – easy to implement, also in the case of CD models

• **Problems**
  – produces too many variations (20..30% for CD, more for CI)
  – this causes too much lexical confusion
    e.g. command - comment
Target Transcriptions : M3 (ELIS)

- **Methodology**
  - perform phoneme recognition
    ⇒ purely acoustic-based phonetic transcription (APT)
  - perform word recognition
    ⇒ recognized word sequence (RWS)
    ⇒ first lexicon-based phonetic transcription (RWPT)
  - get correct word sequence (CWS) from orthography
    ⇒ second lexicon-based phonetic transcription (CWPT)
  - align both RWPT and CWPT with APT (also use time info)
  - **accept** transformation from CWPT to APT if it appears in a word that is not correctly recognized
Example of Target Transcription

Orthography (CWS):  ... in place of a ...  
Recognized (RWS):  ... on trace of a ... 

<table>
<thead>
<tr>
<th>CWPT</th>
<th>APT</th>
<th>Word-pair</th>
<th>Accept</th>
</tr>
</thead>
<tbody>
<tr>
<td>/n/ \rightarrow /m/</td>
<td>in - on</td>
<td>yes</td>
<td>(*)</td>
</tr>
<tr>
<td>/p/ \rightarrow /b/</td>
<td>place - trace</td>
<td>yes</td>
<td></td>
</tr>
<tr>
<td>/~/ \rightarrow /j/</td>
<td>place - trace</td>
<td>yes</td>
<td>(*)</td>
</tr>
<tr>
<td>/v/ \rightarrow /f/</td>
<td>of - of</td>
<td>no</td>
<td></td>
</tr>
<tr>
<td>/~/ \rightarrow /t/</td>
<td>a - a</td>
<td>no</td>
<td></td>
</tr>
</tbody>
</table>

LEXICON : In p le s @ v @  
TARGET : Im b le j s @ v @
4. Learn Pronunciation Rules

• Alignment of lexicon and target transcription

Orthography:  # he is alone #

LEXICON:  % # % h i i % I s % @ l O w n % # %

TARGET:  % # % h i j % I z % l o n % # %

Discrepancies:  - - - - - - i - - s - d - s d - - - -

• General outline of learning process

– Identify frequently occurring transformations (F,F’)
– Identify candidate rules (different for HT and HC)
– Prune rules (use entropy or validation set)
– Compute firing probabilities of retained rules
Experimental Results

- **Database**
  - TIMIT: core test set
  - WSJ: Nov93 Hub 2 test set

- **Baseline system** (Table 1)

- **System with PVM** (Table 2)
  - M1 for targets CI-HMM
  - M2 for targets CD-HMM
  - HT rule hierarchy

- **Conclusions**
  - PVM helps when CI-HMMs
  - little effect when CD-HMMs
  - need CD-HMMs for high performance

<table>
<thead>
<tr>
<th>Database</th>
<th>System</th>
<th>WER(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>TIMIT</td>
<td>CI-HMM</td>
<td>7.26</td>
</tr>
<tr>
<td></td>
<td>CD-HMM</td>
<td>4.59</td>
</tr>
<tr>
<td>WSJ</td>
<td>CI-HMM</td>
<td>19.62</td>
</tr>
<tr>
<td></td>
<td>CD-HMM</td>
<td>9.15</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Database</th>
<th>System</th>
<th>WER (%)</th>
<th># rules</th>
<th>Rel. Impr.</th>
</tr>
</thead>
<tbody>
<tr>
<td>TIMIT</td>
<td>CI-HMM</td>
<td>6.50</td>
<td>69</td>
<td>10.5%</td>
</tr>
<tr>
<td></td>
<td>CD-HMM</td>
<td>4.52</td>
<td>220</td>
<td>1.5%</td>
</tr>
<tr>
<td>WSJ</td>
<td>CI-HMM</td>
<td>18.21</td>
<td>112</td>
<td>7.2%</td>
</tr>
<tr>
<td></td>
<td>CD-HMM</td>
<td>9.02</td>
<td>148</td>
<td>1.3%</td>
</tr>
</tbody>
</table>
Experimental Results on CD-HMMs

- M2 versus M3 for target transcription generation
- HT versus HC for rule hierarchy

### TIMIT

<table>
<thead>
<tr>
<th>Target Trans.</th>
<th>Rule Learning</th>
<th># rules</th>
<th>WER (%)</th>
<th>Rel. Impr.</th>
</tr>
</thead>
<tbody>
<tr>
<td>M2</td>
<td>HT</td>
<td>220</td>
<td>4.52</td>
<td>1.5%</td>
</tr>
<tr>
<td>M3</td>
<td>HT</td>
<td>94</td>
<td>4.27</td>
<td>7.0%</td>
</tr>
<tr>
<td>M2</td>
<td>HC</td>
<td>246</td>
<td>4.46</td>
<td>2.8%</td>
</tr>
<tr>
<td>M3</td>
<td>HC</td>
<td>84</td>
<td>4.20</td>
<td>8.5%</td>
</tr>
</tbody>
</table>

### WSJ

<table>
<thead>
<tr>
<th>Target Trans.</th>
<th>Rule Learning</th>
<th># rules</th>
<th>WER (%)</th>
<th>Rel. Impr.</th>
</tr>
</thead>
<tbody>
<tr>
<td>M2</td>
<td>HT</td>
<td>148</td>
<td>9.02</td>
<td>1.4%</td>
</tr>
<tr>
<td>M3</td>
<td>HT</td>
<td>162</td>
<td>9.15</td>
<td>0.0%</td>
</tr>
<tr>
<td>M2</td>
<td>HC</td>
<td>151</td>
<td>9.15</td>
<td>0.0%</td>
</tr>
<tr>
<td>M3</td>
<td>HC</td>
<td>98</td>
<td>9.05</td>
<td>1.1%</td>
</tr>
</tbody>
</table>
Conclusions

• PVM does help in combination with CI-HMMs
• PVM can give small improvement in combination with CD-HMMs
  – careful target transcription generation needed
  – rule hierarchy seems not important (up to now)
• Future work
  – use phonetic classes in rule conditions
  – use decision trees as a rule hierarchy