Minimum Bayes Risk Estimation and Decoding in Large Vocabulary Continuous Speech Recognition

Bill Byrne
with Vlasios Doumpiotis, Veera Venkataramani, and Stavros Tsakalidis

Machine Intelligence Lab, Department of Engineering, Cambridge University
Trumpington Street, Cambridge CB2 1PZ, U.K.

Center for Language and Speech Processing. The Johns Hopkins University
3400 N. Charles Street, Baltimore, MD 21218, U.S.A.

wjb31@eng.cam.ac.uk

December 20, 2004
Outline

Introduction and Motivation
   Role of Loss in Speech Recognition
   MBR Search Strategies
   Lattice-to-String Alignment and Risk Computation
   Efficient MBR Search – Segmental MBR Decoding

MBR Discriminative Training
   EBW & Risk-Based Parameter Estimation
   Risk-Based Estimation, Lattice Cutting, and Induced Loss Functions
   Analysis

Acoustic Codebreaking
   Search Space – Unsupervised LVCSR SubProblem Selection
   Specialized Acoustic Models for LVCSR SubProblems
   Analysis and Discussion
Outline

Introduction and Motivation
- Role of Loss in Speech Recognition
- MBR Search Strategies
- Lattice-to-String Alignment and Risk Computation
- Efficient MBR Search – Segmental MBR Decoding

MBR Discriminative Training
- EBW & Risk-Based Parameter Estimation
- Risk-Based Estimation, Lattice Cutting, and Induced Loss Functions
- Analysis

Acoustic Codebreaking
- Search Space – Unsupervised LVCSR SubProblem Selection
- Specialized Acoustic Models for LVCSR SubProblems
- Analysis and Discussion
Loss-Based Estimation and Modeling

Goal: A statistical modeling framework to study the integration of ASR, MT, and other language processing technologies into complex information processing systems

- Need an objective characterization of overall performance

Loss functions
- Describe performance under different applications

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Loss</th>
</tr>
</thead>
<tbody>
<tr>
<td>Words</td>
<td>2</td>
</tr>
<tr>
<td>Keywords</td>
<td>1</td>
</tr>
<tr>
<td>Sentence</td>
<td>1</td>
</tr>
<tr>
<td>Meaning</td>
<td>“big”</td>
</tr>
</tbody>
</table>

Loss functions capture application-specific errors
- Word Error Rate, Sentence Error Rate, Keyword Error Rate, ...

Relevant to ‘Moving Beyond HMMs’ ...
Statistical Speech Recognition

Maximum Likelihood and MAP Criteria for estimation and decoding

Acoustic Model Estimation:

\[ \hat{\theta} = \arg \max_\theta P(O|\bar{W}; \theta) \]

Decoding:

\[ \hat{W} = \arg \max_W P(O|W; \theta) P(W; \theta) \]

These have guided the development and evolution of HMMs in ASR

As criteria, these lead to uniform search and estimation criteria

- Conditions under which these criteria are optimal are well defined, but probably not applicable in practice
- **Goal:** loss-specific estimation and search strategies
- tailor search and estimation to specific tasks

*Start with search strategies ...*
Minimum Bayes-Risk Decoding

Given a model and a loss function, performance is defined under the expected loss

\[ E_{P(W,O)}L(W, \delta(O)) = E_{P(O)}E_{P(W|O)}L(W, \delta(O)) \]

Goal is to find the optimum decoder \( \hat{\delta}(O) \) with minimum risk

\[ \hat{W} = \arg\min_{W \in \mathcal{W}} \sum_{W' \in \mathcal{W}} L(W, W')P(W'|O) \]

Assumptions:

- \( P(W'|O) \) are derived from models
- Risk computation and search is over \( \mathcal{W} \)
- \( \mathcal{W} \) are lattices generated by a first-pass ASR system
MBR Search is Closely Linked to MAP Decoding

MAP decoder is motivated by

- Bayes risk under the Sentence Error loss function
- Efficient search algorithms

Suppose the loss function is the 0/1 ‘Sentence Error’ loss function:

\[ L_{0/1}(W, W') = \begin{cases} 0 & W = W' \\ 1 & \text{otherwise} \end{cases} \]

then

\[ \hat{W} = \arg\min_{W \in \mathcal{W}} \sum_{W' \in \mathcal{W}} L_{0/1}(W, W') P(W'|O) = \arg\max_{W \in \mathcal{W}} P(W|O) \]

MAP decoder is optimum in terms of expected loss under 0/1 Loss

- may be suboptimal, depending on the loss function
- but efficient search is possible due to the HMM conditional independence assumptions
An Optimal MBR Decoding Strategy

Decision theory specifies the optimum decoder

1. Make a list of all sentence hypotheses: \( \mathcal{W} \)
2. Find the loss between all strings \( W \) and \( W' \) from \( \mathcal{W} \)
3. Evaluation the risk associated with taking \( W \) as the decision

\[
R(W; \mathcal{W}) = \sum_{W' \in \mathcal{W}} L(W, W')P(W'|O)
\]

4. Pick the hypothesis with the \textbf{minimum risk}

\[
\hat{W} = \arg\min_{W \in \mathcal{W}} R(W; \mathcal{W})
\]

Given models and a loss, MBR decoding is a search problem ...

- Steps 2, 3 and 4 can be done simultaneously over lattices via \( A^* \)
Difficulties in Lattice MBR Computation

Computing with likelihoods over a lattice is easy

- likelihoods of hypotheses are maintained over lattice arcs

What about computing with loss, specifically Word Error Rate?

For every lattice path $W \in \mathcal{W}$ the following has to be found

$$ R(W; \mathcal{W}) = \sum_{W' \in \mathcal{W}} L(W, W')P(W'|O) $$

- Path-by-path comparison of $W$ to all other lattice paths $W'$
- Path distances are weighted by the complete likelihood of $W'$
- Weighted path distances are accumulated to form $R(W)$
- Overall complexity is determined by the size of $\mathcal{W}$
Word Error Rate Requires String-to-String Alignment

Levenshtein distance between $W$ and $W'$

- Introduce a word-to-word alignment process $a$
- Find the optimum alignment based on a symbol-to-symbol cost

$$L(W, W') = \min_a \sum_i L(W_i, W'_a)$$

- For a single string $W'$, this is easily computed over permissible alignments by dynamic programming

Finding $R(W; \mathcal{W})$ requires aligning every lattice path $W' \in \mathcal{W}$ to $W$
- this can be done efficiently...
Efficient (nearly) Exact Lattice-to-String Alignment

Original Lattice

Aligned Lattice

- Every path in the original lattice is retained
- Word arcs are duplicated, if needed
- Word arcs are marked by alignment to one of the words of $W$
Computation of Risk via Lattice-to-String Alignments

Easy computation for loss between $W$ and all $W' \in \mathcal{W}$

$$L(W, W') = \sum_i L(W_i, W'_{a_i})$$

Likelihoods of paths from the original lattice are retained.

Easy computation of the overall risk of $W$

$$R(W; \mathcal{W}) = \sum_{W' \in \mathcal{W}} L(W, W') P(W'|O)$$

$$= \sum_{W' \in \mathcal{W}} \left[ \sum_i L(W_i, W'_{a_i}) \right] P(W'|O) \quad \text{over N-Best lists}$$

$$= \sum_i \sum_{w \in \mathcal{W}_i} L(W_i, w) P_i(w|O) \quad \text{over lattices}$$

- $\mathcal{W}_i$ are all the word arcs aligned to $W_i$
- Computational cost is still determined by the size of $\mathcal{W}$
Lattice Segmentation and Induced Loss Functions

**Aligned Lattice**

**Pinched Lattice**
- No paths lost

**Pinched and Pruned Lattice**
- $\mathcal{W}$ is reduced

\[
L_I(W, W') = L(W_1, W'_1) + L(W_2, W'_2) + L(W_3, W'_3), + L(W_4, W'_4)
\]

Alignments are approximate: $L(W, W') \leq L_I(W, W')$
Segmental MBR Decoding

Risk-based lattice-to-string alignment was developed to facilitate MBR decoding.

If MBR decoding is performed with the Induced Loss function over the pinched lattice, the procedure simplifies as follows:

\[
\begin{align*}
\arg\min_{W \in \mathcal{W}} \sum_{W' \in \mathcal{W}} L(W, W') P(W'|O) & \approx \arg\min_{\tilde{W}} \sum_{\tilde{W}' \in \tilde{W}} L_I(W, W') P(W'|O) \\
& \approx \arg\min_{\tilde{W}} \sum_{i} \sum_{w} L(W_i, w) P_i(w|O)
\end{align*}
\]

Sequence of independent MBR searches over the sublattices using costs and path probabilities derived from the original lattice

- leads to great improvements in search speed relative to A* search over the original lattice
- performance improves because search can proceed without pruning

Can be used for LVCSR decoding, system combination, multilingual acoustic modeling, ...
Outline

**Introduction and Motivation**
- Role of Loss in Speech Recognition
- MBR Search Strategies
- Lattice-to-String Alignment and Risk Computation
- Efficient MBR Search – Segmental MBR Decoding

**MBR Discriminative Training**
- EBW & Risk-Based Parameter Estimation
- Risk-Based Estimation, Lattice Cutting, and Induced Loss Functions
- Analysis

**Acoustic Codebreaking**
- Search Space – Unsupervised LVCSR SubProblem Selection
- Specialized Acoustic Models for LVCSR SubProblems
- Analysis and Discussion
Outline

Introduction and Motivation
  Role of Loss in Speech Recognition
  MBR Search Strategies
  Lattice-to-String Alignment and Risk Computation
  Efficient MBR Search – Segmental MBR Decoding

MBR Discriminative Training
  EBW & Risk-Based Parameter Estimation
  Risk-Based Estimation, Lattice Cutting, and Induced Loss Functions
  Analysis

Acoustic Codebreaking
  Search Space – Unsupervised LVCSR SubProblem Selection
  Specialized Acoustic Models for LVCSR SubProblems
  Analysis and Discussion
MBR Discriminative Training of Acoustic Models

MBR and SMBR are extensions of MAP decoding; investigate analogous extensions of ML parameter estimation.

Suppose we have labeled training data $(\bar{W}, O)$.

One possibility is to minimize the Expected Risk:

$$\theta^* = \arg\min_\theta \sum_{W' \in W} L(\bar{W}, W') P(O|W'; \theta)$$

**Rationale:** A hypothesis contributes to the risk if it is has many errors and is relatively likely. The objective in training is to move the probability away from these and towards hypotheses with fewer errors.

Variations of this have been studied for ASR:

- Kaiser, Horvat, & Kacic, 2000; Povey & Woodland, 2002
  - good idea
  - difficult to implement
Generalization of MMI

Conditional Maximum Likelihood Criterion

$$\max_{\theta} P(\bar{W}|O; \theta)$$

Instance of Minimum Risk estimation under the 0/1 loss function

$$\min_{\theta} \sum_{W' \in \mathcal{W}} L(\bar{W}, W') P(O|W'; \theta) \leftrightarrow \max_{\theta} P(\bar{W}|O; \theta)$$

Iterative estimation procedures for each can be realized via the Extended Baum Welch algorithm
The Extended Baum Welch Algorithm and Risk-Based Parameter Estimation

Kaiser, Horvat, & Kacic, 2000 noted that the objective function

\[ \sum_{W' \in \mathcal{W}} L(\bar{W}, W') P(O|W'; \theta) \]

is a rational polynomial that can be optimized via the Extended Baum Welch algorithm.

Update for a Gaussian mean of an HMM observation distribution:

\[ \bar{\mu}_s = \frac{\sum_{W' \in \mathcal{W}} K(W'; \mathcal{W}, \theta) \sum_{\tau} \gamma_s(\tau; W') o(\tau) + D_s \mu_s}{\sum_{W' \in \mathcal{W}} K(W'; \mathcal{W}, \theta) \sum_{\tau} \gamma_s(\tau; W') + D_s} \]

where

\[ K(W'; \mathcal{W}, \theta) = \left[ \sum_{W'' \in \mathcal{W}} P(W''|O; \theta) L(\bar{W}, W'') - L(\bar{W}, W') \right] P(W'|O; \theta) \]

As with MAP and SMBR decoding, this simplifies to the MMI estimation procedure under the 0/1 loss function.
Direct Realization of EBW Update Rule

\[
\bar{\mu}_s = \frac{\sum_{W' \in \mathcal{W}} K(W'; \mathcal{W}, \theta) \sum_{\tau} \gamma_s(\tau; W') o(\tau) + D_s \mu_s}{\sum_{W' \in \mathcal{W}} K(W'; \mathcal{W}, \theta) \sum_{\tau} \gamma_s(\tau; W') + D_s}
\]

where \( K(W'; \mathcal{W}, \theta) = [R(\tilde{W}; \mathcal{W}, \theta) - L(\tilde{W}, W')] P(W'|O; \theta) \)

Straightforward to compute over N-Best lists:

- \( R(\tilde{W}; \mathcal{W}, \theta) \) can be found over the N-Best list
- the \( W' \) in \( \mathcal{W} \) are enumerated
- for each \( W' \)
  - \( L(\tilde{W}, W') \) can be found
  - the statistics \( \gamma_s(\tau, W') \) have to be found, e.g. by Baum Welch

N-Best lists are not ideal for LVCSR discriminative training

- enumeration of likely hypotheses leads to very large N-Best lists
- for this particular problem, there is the danger of underestimating the risk by using too small a search space
Risk-Based Pruning of the Evidence Space

N-Best lists in LVCSR are large because LVCSR lattices are large

- Can use lattice pinching and pruning to reduce $\mathcal{W}$

Extended Baum Welch parameter estimation can be performed over N-Best lists extracted from the reduced lattices

- Corresponds to MBR estimation with respect to the induced loss function over the pinched lattice

$$\theta^* = \arg \min_{\theta} \sum_{W' \in \tilde{\mathcal{W}}} L_I(\tilde{W}, W') P(O|W'; \theta)$$

- Statistics computed under $P(O|w'; \theta)$ can be computed by Baum Welch
- Time segmentation, at word or phone boundaries, is not needed
Lattice Cutting for MBR

- Generate lattices over the acoustic training set; also used for MMI
- Perform lattice-to-string alignment to the reference transcription
- Pinch and prune lattices
- Identify the frequently occurring confusion pairs
- Prune all other choices back to the reference path

Greatly reduces the size of the original lattice
- retains the likely paths with frequent confusions
- many training utterances are discarded – lattices pruned back to the reference contribute nothing to the overall risk
### Lattice Segmentation and LVCSR Training Set Refinement

<table>
<thead>
<tr>
<th></th>
<th>SWITCHBOARD</th>
<th>MALACH-CZ</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acoustic training data</td>
<td>16.9 / 22,580</td>
<td>62.4 / 24,065</td>
</tr>
<tr>
<td>(hours / utterances)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Initial confusion pairs</td>
<td>25,948 / 99,199</td>
<td>31,467 / 120,695</td>
</tr>
<tr>
<td>(types / tokens)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Occurrence threshold used</td>
<td>5</td>
<td>75</td>
</tr>
<tr>
<td>to select confusion pairs</td>
<td></td>
<td>100</td>
</tr>
<tr>
<td>Confusion pairs after</td>
<td>2,139 / 66,349</td>
<td>159 / 33,821</td>
</tr>
<tr>
<td>filtering (types / tokens)</td>
<td></td>
<td>117 / 48,302</td>
</tr>
<tr>
<td>Avg. confusion pairs</td>
<td>0.35 / 3.37</td>
<td>0.2 / 2.14</td>
</tr>
<tr>
<td>(per word / per utterance)</td>
<td></td>
<td>0.13 / 3.12</td>
</tr>
<tr>
<td>Reduced acoustic training</td>
<td>15.0 / 19,687</td>
<td>13.0 / 15,741</td>
</tr>
<tr>
<td>data (hours / utterances)</td>
<td></td>
<td>52.4 / 15,436</td>
</tr>
<tr>
<td>Avg. depth of N-Best lists</td>
<td>48.8</td>
<td>13.1</td>
</tr>
<tr>
<td>from pinched lattices</td>
<td></td>
<td>36.5</td>
</tr>
</tbody>
</table>
MBRDT Can Improve Over MMI for LVCSR

- Levenstein: MBR DT
- 0/1 Loss: MMI over pinched lattice
- One-Worst: MBR DT against the worst hypothesis in the pinched lattice

▶ 39 dim. MFCC, bigram with 83K word voc., 2 hour test set
▶ MBR DT was performed by expanding the pinched lattices into N-Best lists and performing Baum Welch over each entry
Analysis

Yields a single set of models trained to resolve the most frequently occurring binary word confusions.

- Lattice-to-String alignment creates an induced loss function
- Lattice pinching and pruning is used to restrict the evidence space to the hypotheses that are both likely and errorful
- These lattices are small enough to be extended into N-Best lists
- Risk-based estimation using Extended Baum Welch can be carried out over the N-Best lists
- Acoustic training set is refined to focus on the training utterances most relevant to the loss function
- Improvements relative to MMI
- More than simply MMI over pinched lattices – there are gains in using Levenstein loss rather than 0/1 Loss over pinched lattices

Tuning the estimation procedure to the task specific loss function is feasible
Outline

Introduction and Motivation
- Role of Loss in Speech Recognition
- MBR Search Strategies
- Lattice-to-String Alignment and Risk Computation
- Efficient MBR Search – Segmental MBR Decoding

MBR Discriminative Training
- EBW & Risk-Based Parameter Estimation
- Risk-Based Estimation, Lattice Cutting, and Induced Loss Functions
- Analysis

Acoustic Codebreaking
- Search Space – Unsupervised LVCSR SubProblem Selection
- Specialized Acoustic Models for LVCSR SubProblems
- Analysis and Discussion
Outline

Introduction and Motivation
  Role of Loss in Speech Recognition
  MBR Search Strategies
  Lattice-to-String Alignment and Risk Computation
  Efficient MBR Search – Segmental MBR Decoding

MBR Discriminative Training
  EBW & Risk-Based Parameter Estimation
  Risk-Based Estimation, Lattice Cutting, and Induced Loss Functions
  Analysis

Acoustic Codebreaking
  Search Space – Unsupervised LVCSR SubProblem Selection
  Specialized Acoustic Models for LVCSR SubProblems
  Analysis and Discussion
Acoustic Codebreaking

Divide-and-Conquer approach to ASR

- Perform a first decoding pass to generate lattices
- Align lattice to the MAP hypothesis
- Perform lattice segmentation to locate and characterize errors

Unsupervised identification of recognition errors

- Search training data to create training sets for all these errors
- Train specialized models for each type of error

Can use different types of models for different types of errors: HMMs, SVMs, language models, pronunciation models, ...

- Decode pinched lattices using the specialized models
- Repeat until all errors are resolved

Not the first Divide-and-Conquer approach to ASR

Not an attempt to replace HMMs by hierarchical classifiers
Objective is to find weakness in the HMM hypothesis, and fix them
Unsupervised Selection of Confusion Pairs

**Goal:** Identify *regions of uncertainty* in the first-pass ASR lattice that can be resolved by correctly choosing one of two words

Consider the instance of the confusion pair \( C_5 = \{V:5, B:5\} \)

The MAP hyp is \( V \). There are three scenarios

- **Case 1:** the MAP hyp is correct → don’t fix this instance of \( C_5 \)
- **Case 2:** the MAP hyp might be wrong → consider replacing \( V \) by \( B \)
- **Case 3:** neither \( V \) nor \( B \) is right → ignore this instance
A Simple Pruning Scheme for Finding Confusion Pairs

Aligning a lattice to a string yields segment sets

To select confusion pairs:

- prune links by their posteriors relative to a set threshold
- some segment sets shrink to pairs – these are the *natural confusion pairs*
- prune all other sets back to the MAP hypothesis

### MALACH-CZ 25 Hour Test Set

<table>
<thead>
<tr>
<th>Pruning Threshold</th>
<th>LER</th>
<th>Avg. # Hyps. / Segment Set</th>
<th>Segment Sets</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Types</td>
</tr>
<tr>
<td>0.00</td>
<td>27.3</td>
<td>11.65</td>
<td>94029</td>
</tr>
<tr>
<td>0.05</td>
<td>35.3</td>
<td>2.82</td>
<td>49837</td>
</tr>
<tr>
<td>0.10</td>
<td>37.9</td>
<td>2.35</td>
<td>35278</td>
</tr>
<tr>
<td>0.20</td>
<td>41.1</td>
<td>2.06</td>
<td>17132</td>
</tr>
<tr>
<td>0.30</td>
<td>43.2</td>
<td>2.00</td>
<td>7288</td>
</tr>
<tr>
<td>0.40</td>
<td>44.7</td>
<td>2.00</td>
<td>2249</td>
</tr>
<tr>
<td>0.50</td>
<td>45.6</td>
<td>-</td>
<td>0</td>
</tr>
</tbody>
</table>
## Error Analysis of Natural Confusion Pairs

- **CPOC**: Confusion Pair Oracle Correct – pair contains the truth
- **CPOE**: Confusion Pair Oracle Error – both hyps are wrong
- **MAPERR**: MAP hyp is in Error (CPOE → MAPERR)
- **MAPCOR**: MAP hyp is Correct

### MALACH-CZ 25 Hour Test Set

<table>
<thead>
<tr>
<th>Pruning Threshold</th>
<th>#CPOC/CPERR</th>
<th>#MAPERR/MAPCOR</th>
<th>Segment Sets</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Types</td>
<td>Tokens</td>
<td>Types</td>
</tr>
<tr>
<td>0.00</td>
<td>14.30</td>
<td>0.24</td>
<td>22</td>
</tr>
<tr>
<td>0.05</td>
<td>4.7</td>
<td>0.64</td>
<td>26</td>
</tr>
<tr>
<td>0.10</td>
<td>3.3</td>
<td>0.92</td>
<td>26</td>
</tr>
<tr>
<td>0.20</td>
<td>3.2</td>
<td>1.17</td>
<td>17</td>
</tr>
<tr>
<td>0.30</td>
<td>4.2</td>
<td>1.15</td>
<td>6</td>
</tr>
<tr>
<td>0.40</td>
<td>11.0</td>
<td>1.04</td>
<td>2</td>
</tr>
<tr>
<td>0.50</td>
<td>-</td>
<td>-</td>
<td>0</td>
</tr>
</tbody>
</table>

At a threshold of 0.10:

- a confusion pair is 3.3 times more likely to contain the truth than not
- the MAP hyp is wrong roughly half the time
- there are 26 distinct confusion pairs occurring 6860 times
SVMs for Resolving Binary Confusions in LVCSR

Of the 26 confusion pairs, 5 are *homonym*.

- language modeling errors are separated from acoustic modeling errors

The remaining 2991 instances of the 21 confusion pairs formed the isolated word test set within the 25 hour test set.

*Gini* SVMs were trained for each of the 21 confusion pairs\(^1\)

- features: score-spaces derived from MMI-trained whole word HMMs\(^2\)
- MMI whole word HMMs: e.g. for the pair (V:5,B:5),
  - a training set was created of all utterances of V or B
  - words were extracted from continuous speech by forced alignment using a set of baseline HMMs
  - a V:5 and a B:5 model were trained over this set using MMI
- this was done for all pairs
- lattices were not generated over the training set

---

1. Venkataramani & Byrne, ASRU’03
2. Smith & Gales, ICASSP’02
A Small Recognition Task in LVCSR

SVMs were applied to ‘fix’ the confusion pair hypotheses

Confidence scores were extracted from the SVM and interpolated with the word posteriors from the segmented lattices – word hypotheses were chosen

- various voting schemes are possible
- system combination at the word level is simple in this way

In 18 of the 21 cases, the WER over the isolated word test set decreased

- consistent with previous performance over small vocabulary tasks

LVCSR WER decreased from 45.6% to 45.5% with \( p \text{value} \leq 0.001 \)

---

\(^3\)Venkataramani & Byrne, ICASSP 2004
Analysis

In Particular: Novel procedure to use SVMs in LVCSR

- loss-based lattice cutting can transform the recognition task to problems more easily solved by SVMs

In General: New framework for trying out novel recognition and modeling strategies without losing the advantages of well-trained HMM systems

Successful proof-of-concept:

- identified small vocabulary problems inside an LVCSR problem
- trained and applied novel classifiers for the small vocabulary problems
- integrated the small vocabulary classifiers back into the LVCSR system
- reduced WER of a well-trained MMI-trained model with MLLR, etc.
- large LVCSR test set → large tests for the small vocabulary subproblems and significant (if small) LVCSR results

Implications for novel techniques:

- feasible to associate modeling ‘fixes’ with specific search space choices
- unsentimental analysis of the overall gains that can be expected
  - e.g., large LVCSR gains are unlikely with only a few simple classifiers
Risk-Based Lattice Segmentation

Developed to support MBR decoding over lattices

- lattice-to-string alignment under specified loss functions
- supports computation of risk over lattices analogous to likelihood

Careful refinement of the search space

- all lattice paths are retained unless they are pruned away
- no paths are lost due to the alignment procedure itself
HMMs in LVCSR – Generalization is All-Important

Objective: a single set of models in a single system capable of transcribing any (in-domain) utterance spoken by any (in-population) speaker

Evolution of HMMs has been driven by this need for generalization:

- efficient estimation algorithms to train general purpose generative models from large and diverse training sets
- algorithms such as mixture splitting, state clustering, and backoff LMs to balance model complexity against training data size to build the largest/smallest models capable of generalizing to new data
- efficient decoding algorithms for complex models required to achieve generalization

In some ways, the state of the art in HMMs has already moved beyond HMMs:

- MMI – non-generative models
- ROVER/system-combination – abandons the idea that a single set of models is adequate to describe all speech
- Speaker Adaptation – undoes the ability to generalize across speakers
- retains the emphasis on large training sets and general purpose models
Risk-Based Lattice Segmentation and Modeling Strategies for Going Beyond HMMs

*Enabling technology* for new modeling and estimation techniques

Two examples of techniques that would otherwise be infeasible for LVCSR:

▶ Min Risk via Extended Baum Welch and SVMs

Goals:

▶ Bring novel techniques into LVCSR more quickly
▶ Enable more people to work on LVCSR problems

Consistent with other contemporary modeling techniques that depart from the ML-motivated evolution of HMMs. Specifically:

▶ risk minimization vs. likelihood maximization, in estimation and in decoding
▶ abandon the goal of developing a single set of models capable of recognizing all speech
▶ parameter estimation tailored for system combination
▶ training focuses only on what is relevant as defined by the induced loss functions – *irrelevant training data is explicitly discarded*
▶ reduced training sets but *very large test sets*
Thank You!