A Unified Approach to Minimum Risk Training and Decoding

Abhishek Arun, Barry Haddow and Philipp Koehn

University of Edinburgh

Fifth Workshop on Machine Translation, Uppsala, July 16th 2010
Current Approaches to Minimum Risk Decoding
A Unified Approach
Markov Chain Monte Carlo for Phrase-based MT
Minimum risk training
Optimising corpus BLEU
Experiments
Conclusions and Future work
Minimum Risk Decoding in MT

Optimal Decision Rule?
- Find the target sentence which minimises expected risk
  - Equivalently: Maximises expected gain
- Summarised by the following equation

\[ e^* = \arg \max_e \sum_{e'} p(e' \mid f) \text{Gain}(e', e) \]

\[ f - \text{source, } e - \text{target} \]

- We use BLEU as the gain function
- Referred to as **Minimum Bayes Risk (MBR) Decoding.**
Current Approaches to MBR Decoding

- First-pass decoder scores translations with linear model
- The scores must be scaled and normalised to give probabilities
  - Scaling requires hyper-parameter search
  - Normalisation requires intractable sum
- MBR Decoding Implemented as a list re-ranker
- Feature weights in linear model trained with MERT
  - Non-probabilistic training algorithm
  - Aims to maximise 1-best (MAP) performance
Lattice-Based Approaches

- Represent many hypotheses compactly
- State-of-the-art performance from Lattice MBR
- But
  - Feature weights trained with MERT
  - Biased pruning - May be bad for sparse features
  - Need to approximate BLEU- more hyperparameters
A Unified Approach

<table>
<thead>
<tr>
<th>Training</th>
<th>Decoding</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optimise Expected <strong>BLEU</strong></td>
<td>Maximise Expected <strong>BLEU</strong></td>
</tr>
</tbody>
</table>

- Objective is differentiable
  - Can use gradient-based optimisation
- Use **Markov Chain Monte Carlo (MCMC)** to estimate:
  - Feature expectations during training - for gradient
  - Expected **BLEU** during decoding
Benefits of Our Approach

- Maintains a probabilistic formulation throughout
  - Theoretically sound
  - Unbiased estimates
- Avoids dynamic programming so non-local features easier
- Compared to MERT:
  - More stable
  - Generalises better
  - Gives better performance
MCMC Sampler for Phrase-based MT

Used to draw samples \( \{(e_i, a_i)\} \) from \( p(e, a|f) \)
- Use the samples to estimate expectations

\[
E(h) \approx \frac{1}{N} \sum_{(e_i, a_i)} h(e_i, a_i, f)
\]

Transitions \( T_i \) defined by Transition Operators
- Make small local changes to hypothesis
- Apply all operators in sequence before collecting sample
MCMC Operators

**RETRANS**
Retranslates one source-target phrase pair

**MERGE-SPLIT**
Operates at an inter-word position. May merge or split segments as appropriate, and retranslate.

**REORDER**
Swaps target position of two source-target phrase pairs
MCMC Example

(a) Initial

\[
\begin{align*}
\text{c'est} & \quad \circ \quad \text{un} & \quad \circ \quad \text{résultat} & \quad \circ \quad \text{remarquable} \\
\text{it is} & \quad \circ \quad \text{some} & \quad \circ \quad \text{result} & \quad \circ \quad \text{remarkable}
\end{align*}
\]

(b) Retrans

\[
\begin{align*}
\text{c'est} & \quad \bullet \quad \text{un} & \quad \bullet \quad \text{résultat} & \quad \bullet \quad \text{remarquable} \\
\text{but} & \quad \bullet \quad \text{some} & \quad \bullet \quad \text{result} & \quad \bullet \quad \text{remarkable}
\end{align*}
\]

(c) Merge

\[
\begin{align*}
\text{c'est} & \quad \circ \quad \text{un} & \quad \bullet \quad \text{résultat} & \quad \bullet \quad \text{remarquable} \\
\text{it is a} & \quad \bullet \quad \text{result} & \quad \bullet \quad \text{remarkable}
\end{align*}
\]

(d) Reorder

\[
\begin{align*}
\text{c'est} & \quad \bullet \quad \text{un} & \quad \bullet \quad \text{résultat} & \quad \bullet \quad \text{remarquable} \\
\text{it is a} & \quad \bullet \quad \text{remarkable} & \quad \bullet \quad \text{result}
\end{align*}
\]
Our objective is the expected gain plus an entropic prior

\[ \hat{G} = \sum_{\langle e,f \rangle \in \mathcal{D}} \left[ \left( \sum_{e,a} p(e, a | f) \text{BLEU}_{\hat{e}}(e) \right) + T \cdot H(p) \right] \]

- The temperature \((T)\) starts off high and is gradually reduced.
- This moves from high entropy to low entropy, and helps avoid local maxima.
- Known as Deterministic Annealing (DA)
- The gradient is calculated using the sampler, and optimisation is by stochastic gradient descent.
Corpus Sampling

- **But** we’re optimising sentence BLEU
  - And testing with corpus BLEU
- To eradicate this mismatch, we propose **Corpus Sampling**
- Each sample is an aligned translation of the whole corpus
  - Sentence samples are collected for all sentences
  - These are resampled to give corpus samples
  - Now we can optimise corpus BLEU
Corpus Sampling Illustration

SAMPLE FROM P(e,a | f)

SAMPLE FROM EMPRICAL DISTRIBUTION

Extract Corpus Samples

Corpus Sample 1: \{A, F, L\}
Corpus Sample 2: \{B, E, L\}
Experimental Setup

NIST
Arabic-English
300k Sents Train
In-Domain Test

Europarl
French-English
1.4M Sents Train
In-Domain Test
Out-of-domain Test

Europarl
German-English
1.4M Sents Train
In-Domain Test
Out-of-domain Test

Moses Setup
- Standard phrase extraction pipeline
- Standard features (no lexicalised reordering)
- MERT/Moses for baselines
Effect of deterministic Annealing

- Graphs show heldout performance
- Converges much quicker without DA
- Maximum is lower
- At high entropy, MBR much better than max-derivation
- Advantage reduces with temperature
- We use early stopping to find best weights
## Corpus Sampling vs Sentence Sampling

<table>
<thead>
<tr>
<th>Test Set</th>
<th>Sentence</th>
<th>Corpus</th>
</tr>
</thead>
<tbody>
<tr>
<td>AR-EN MT05</td>
<td>44.6 (0.990)</td>
<td>44.5 (0.989)</td>
</tr>
<tr>
<td>FR-EN In-domain</td>
<td>32.9 (1.003)</td>
<td>33.2 (0.997)</td>
</tr>
<tr>
<td>FR-EN Out-domain</td>
<td>19.7 (1.049)</td>
<td>19.8 (1.041)</td>
</tr>
<tr>
<td>DE-EN In-domain</td>
<td>26.9 (0.987)</td>
<td>27.8 (0.993)</td>
</tr>
<tr>
<td>DE-EN Out-domain</td>
<td><strong>16.6 (0.975)</strong></td>
<td><strong>16.6 (0.980)</strong></td>
</tr>
</tbody>
</table>

- *Expected BLEU* training, MBR decoding
- Table shows BLEU and length penalty
- Corpus sampling slightly better
## Comparison with Moses Baseline

<table>
<thead>
<tr>
<th>Test set</th>
<th>MERT/Moses</th>
<th>Expected BLEU</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Best</td>
<td>σ</td>
</tr>
<tr>
<td>AR-EN MT05</td>
<td>44.5 (lMBR)</td>
<td>0.12</td>
</tr>
<tr>
<td>FR-EN In</td>
<td>33.4 (nMBR)</td>
<td>0.12</td>
</tr>
<tr>
<td>FR-EN Out</td>
<td>19.5 (nMBR)</td>
<td>0.12</td>
</tr>
<tr>
<td>DE-EN In</td>
<td>27.8 (MAP)</td>
<td>0.10</td>
</tr>
<tr>
<td>DE-EN Out</td>
<td>16.0 (lMBR)</td>
<td>0.30</td>
</tr>
</tbody>
</table>

- Compare corpus sampler with best MERT/moses result
  - For sampler, decode with n-best MBR
  - For Moses, best out of MAP, n-best MBR and lattice MBR
- Five runs of expected BLEU, ten runs of MERT, averaged.
### Expected Bleu Training, Moses Decoding

<table>
<thead>
<tr>
<th>Test Set</th>
<th>MAP</th>
<th>nMBR</th>
<th>IMBR</th>
<th>Sampler MBR</th>
</tr>
</thead>
<tbody>
<tr>
<td>AR-EN MT05</td>
<td>44.2</td>
<td>44.4</td>
<td>44.8</td>
<td>44.8</td>
</tr>
<tr>
<td>FR-EN In</td>
<td>33.1</td>
<td>33.2</td>
<td>33.3</td>
<td>33.3</td>
</tr>
<tr>
<td>FR-EN Out</td>
<td>19.6</td>
<td>19.8</td>
<td>19.9</td>
<td>19.9</td>
</tr>
<tr>
<td>DE-EN In</td>
<td>27.7</td>
<td>27.9</td>
<td>28.0</td>
<td>28.0</td>
</tr>
<tr>
<td>DE-EN Out</td>
<td>16.0</td>
<td>16.3</td>
<td>16.6</td>
<td>16.6</td>
</tr>
</tbody>
</table>

- We use the best expected **BLEU** trained weights
- Decoding with Moses (first three columns) or sampler
- Suggests that expected **BLEU** weights better for IMBR
Conclusions

- Unified Training and Decoding beats or equals MERT/Moses
- Deterministic Annealing (entropic prior) provides better performance
- Corpus sampling provides small gains over sentence sampling
- Expected bleu trained weights more suited to lattice MBR decoding, than MERT weights
- MBR and maximum-translation decoding better than maximum-derivation
Future Work

- Supplement dense features with many sparse features
  - eg. discriminative language models
- Incorporate non-local features
  - eg. long-distance agreement
- Metropolis-Hastings step to efficiently incorporate slow features
  - eg. higher-order language model
Thank you!

Questions?

Code:

https://mosesdecoder.svn.sourceforge.net/svnroot/mosesdecoder/branches/josiah