Structural Support Vector Machines for Log-linear Approach in Statistical Machine Translation

Katsuhiko Hayashi†, Taro Watanabe††, Hajime Tsukada††, Hideki Isozaki††
†Doshisha University
††NTT Communication Science Laboratories
Outline

• Introduction

• Proposed Method

• Experiments

• Related Work

• Conclusion and Future Work
Introduction

• Background
  – Minimum Error Rate Training (MERT) is widely used.

• Problem
  – MERT tends to overfit to development data.

• Approach
  – We propose a training method that incorporates regularizer into objective function inspired by Structural Support Vector Machines.
Proposed Method (1/3)

- Objective Function inspired by 1-slack Structural SVM

Oracle Translation

\[
\min_{w, \xi \geq 0} \frac{\lambda}{2} \|w\|^2 + \xi \\
\text{s.t. } \forall (\hat{e}_1, \cdots, \hat{e}_S) \in \mathcal{C}^S:
\frac{1}{S} \sum_{s=1}^{S} \langle w, \delta h_s \rangle \geq \Delta(\hat{e}_s^*, \hat{e}_s^\prime)^S - \xi
\]

\[
\delta h_s = h_s(\hat{e}_s^*, f_s) - h_s(\hat{e}_s, f_s)
\]

\[
\Delta(\{\hat{e}_s^*, \hat{e}_s^\prime\}) = Q \times \left( \text{BLEU}(f_s, \hat{e}_s^\prime) - \text{BLEU}(f_s, \hat{e}_s^*) \right)
\]

Model parameter

Slack Variable

BLEU loss

Feature vector of a translation

Feature vector of the reference

Document-wise BLEU scores
Proposed Method (2/3)

- Och’s Line Search Algorithm can be utilized for optimization.

\[
\hat{e}_{s, best} = \arg\max_{\hat{e}_s \in C_s} \langle w + \alpha d, h(\hat{e}_s, f_s) \rangle
\]

\[
= \arg\max_{\hat{e}_s \in C_s} \left\{ \langle w, h(\hat{e}_s, f_s) \rangle + \alpha \langle d, h(\hat{e}_s, f_s) \rangle \right\}
\]

Update

\[
\hat{w} + \hat{\alpha} \cdot d
\]
Proposed Method (3/3)

• The slope and intercept for each line
  – We need to calculate the slope and intercept by using the following equation

\[
\arg\max_{\hat{e}_s \in C_s} \left\{ \frac{\Delta(e^*, e_s) - \langle w, \delta h_s \rangle + \alpha \langle d, \delta h_s \rangle}{	ext{intercept}} \right\}
\]

However, the slope and intercept calculated by this equation are very noisy because of sentence-wise BLEU scores.
## Comparison to MERT

<table>
<thead>
<tr>
<th></th>
<th>Objective Function</th>
<th>Regularizer</th>
<th>Hyper Parameter</th>
<th>Optimization</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>MERT</strong></td>
<td>Evaluation Metrics (BLEU)</td>
<td>×</td>
<td>×</td>
<td>Line Search Algorithm</td>
</tr>
<tr>
<td><strong>Proposed Method</strong></td>
<td>L2 Regularizer + Empirical Risk (incorporated with BLEU loss)</td>
<td>○</td>
<td>○</td>
<td>Line Search Algorithm, SVM$^\text{struct}$</td>
</tr>
</tbody>
</table>

### Notes
- MERT uses Evaluation Metrics (BLEU) as its objective function.
- The regularizer for MERT is not specified.
- MERT does not use hyper parameters.
- MERT uses a Line Search Algorithm for optimization.

- The proposed method uses L2 Regularizer + Empirical Risk (incorporated with BLEU loss) as its objective function.
- The regularizer for the proposed method is specified.
- The proposed method uses hyper parameters.
- The proposed method uses a Line Search Algorithm and SVM$^\text{struct}$ for optimization.
Advantage

• Our proposed method is a natural extension to regularize MERT’s objective function.

• It is easy to implement
  – We can use almost the same line search algorithm as used in MERT.
Experiments

• **Goal**
  – To investigate a validity of our proposed method, compared with MERT.
  
  – Compare generalization ability
    • In case of out-of-domain
    • With sparse data
Common Settings

• Decoder
  - Moses (Koehn et al., 2007)
    - 14 real-valued features

• Translation Model
  - GIZA++ (Och et al., 2003)

• Language Model
  - SRILM (et al., 2002)
Data Set

- Europarl French-English WMT08-shared task
- **Training data**
  - 1.28M (Europarl)
- **Development data**
  - 2.0K (Europarl)
- **Test data**
  - In-domain test set 2.0k (Europarl)
  - out-of-domain test set 1.5k (News)
Hyper Parameters

- Two hyper parameters were tuned by Cross Validation Method.

\[ \min_{w, \xi \geq 0} \frac{1}{2} \|w\|^2 + \xi \]

\[ s.t. \forall (\hat{e}_1, \ldots, \hat{e}_S) \in C^S : \frac{1}{S} \sum_{s=1}^{S} \langle w, \delta h_s \rangle \geq \Delta([\hat{e}_s^*, \hat{e}_s]) - \xi \]

\[ \Delta([\hat{e}_s^*, \hat{e}_s]) = Q \times \left| \text{BLEU}([r_s, \hat{e}_s]) - \text{BLEU}([r_s, \hat{e}_s]) \right| \]

\( \lambda \) emphasizes the convex regularizer.

\( Q \) is a constant for scaling the BLEU scores.
## Result

- **In-Domain vs Out-of-Domain** -

<table>
<thead>
<tr>
<th>Method</th>
<th>In-Domain</th>
<th>Out-of-Domain</th>
</tr>
</thead>
<tbody>
<tr>
<td>MERT</td>
<td>32.36</td>
<td>13.81</td>
</tr>
<tr>
<td>Smoothed-MERT (Och 03)</td>
<td>31.96</td>
<td>13.76</td>
</tr>
<tr>
<td>Proposed Method</td>
<td>32.42</td>
<td>14.13</td>
</tr>
</tbody>
</table>
Data Sparseness

• We reduced development data (2.0K).
  – 400 sentences randomly selected from a full development data
  – Experiments were conducted 4 times

• We expected our proposed method to reduce overfitting problem.
Result
- Data Sparsness -

- The average BLEU scores on 4 times experiments

<table>
<thead>
<tr>
<th>Method</th>
<th>In-Domain</th>
</tr>
</thead>
<tbody>
<tr>
<td>MERT</td>
<td>31.24</td>
</tr>
<tr>
<td>Smoothed MERT (Och, 03)</td>
<td>31.06 (-0.18)</td>
</tr>
<tr>
<td>Proposed Method</td>
<td>31.76 (+0.52)</td>
</tr>
</tbody>
</table>
Related Work

• Och (2003) tried to regularize MERT’s objective function by using the same regularization as used in Speech Community.

• Cer (2008) proposed window smoothing method with line search algorithm.

Conclusion

• We proposed a learning method for SMT by using a objective function inspired by Structural SVM.

• The objective function involves both document-wise BLEU and a regularizer.

• The proposed method (1-slack) outperforms MERT when the development data size is small.
Conclusion (2/2)

Future Work

• We will apply 1-slack SVM to the decoder which has a large number of features.

• In this case, SVM$^{struct}$ may be a more appropriate optimization algorithm.
Thank you very much for your attention !!