Minimum Error Rate Training Semiring

Artem Sokolov & François Yvon

LIMSI-CNRS & LIMSI-CNRS/Univ. Paris Sud
{artem.sokolov,francois.yvon}@limsi.fr

EAMT’2011
31 May 2011
Talk Plan

1. Introduction
   - Phrase-based statistical machine translation
   - Minimum Error Rate Training
   - Contribution

2. Semirings
   - Lattice MERT
   - MERT Semiring

3. Implementation

4. Experiments
   - Setup
   - Results

5. Future Work
Probability model and inference in SMT system

Probability of translation $e$ given source sentence $f$:

$$p(e|f) = Z(f)^{-1} \exp(\bar{\lambda} \cdot \bar{h}(e, f))$$

- $\bar{h}(e, f)$ – feature vector (various compatibility measures of $e$ and $f$)
- $\bar{\lambda}$ – parameter vector, $\lambda_i$ regulates importance of the feature $h_i(e, f)$

Translating by MAP-inference:

$$\tilde{e}_f(\bar{\lambda}) = \arg\max_{e \in E} p(e|f) = \arg\max_{e \in E} \bar{\lambda} \cdot \bar{h}(e, f)$$

- $E$ – reachable translations (search space), can be approximated by:
  - list of n-best hypotheses
  - word lattice
Tuning SMT system with MERT

**Given:** development set \{ (f, r_f) \} (source f & reference r_f pairs)

**Solve:**

\[
\tilde{\lambda}^* = \arg \max_{\tilde{\lambda}} \text{BLEU}(\{ \tilde{e}_f(\tilde{\lambda}, E(\tilde{\lambda})), r_f \})
\]

- BLEU is non-convex and not differentiable, hence heuristics (MERT).
- Search space approximation depends on \( \tilde{\lambda} \), so iterative tuning:

```
\{(f, r_f)\}  \rightarrow  \text{Decoder}  \rightarrow  \text{Approximation of } E
```

```
\text{Configuration: } \tilde{\lambda}_t  \uparrow  \text{Updater}
```

```
\text{Updated } \tilde{\lambda}_t  \downarrow  \text{Tuning: MERT}
```

```
\text{Updater}  \uparrow  \text{Scorer}
```

```
\text{Features}
```
MERT proceeds in series of optimizations along directions $\bar{r}$:

$$\bar{\lambda} = \bar{\lambda}_0 + \gamma \bar{r}$$

Optimal translation:

$$\tilde{e}_f(\gamma) = \arg \max_{e \in E} \bar{\lambda} \cdot \bar{h}(e, f) = \arg \max_{e \in E} \bar{\lambda}_0 \cdot \bar{h}(e, f) + \gamma \bar{r} \cdot \bar{h}(e, f)$$

- each translation hypothesis is associated with a line,
- **upper envelope**: dominating lines when $\bar{\lambda}$ is moved along $\bar{r}$
- $\gamma$-projections of intersections give intervals of constant optimal hypothesis
- optimal $\gamma^*$ found by merging intervals for $f \in F$ and scoring each
  update $\bar{\lambda} = \lambda_0 + \gamma^*_i \bar{r}_i^*$,
  where $i^*$ is the index of the direction yielding the highest BLEU
MERT problems

- very slow, because of:
  - overall number of iterations
    folklore: number of iterations $\simeq$ number of dimensions
  - slowness of each iteration (dominated by decoding time)
- non-monotonicity/instability of the training process
- sensitivity of the resulting solutions to initial conditions

Ways to tackle the problems

- improve optimization
  - other target function approximations
  - changes into optimization algorithms
- improve search space processing ← this presentation
  - use lattices (better approximation of the complete search space)
  - reduce search to standard operations (facilitates implementation)
- reduce number of iteration ← this presentation
Contribution

- Recast Lattice MERT algorithm of [Macherey et al., 2008] in a semiring framework
  - has already been hinted to in [Dyer et al., 2010]
  - but was never formally described
  - lack of implementation details
- Reimplement MERT using this reformulation
  - and general-purpose FST toolbox OpenFST
Semirings

Semiring $\mathbb{K} = \langle K, \oplus, \otimes, \bar{0}, \bar{1} \rangle$:

- $\langle K, \oplus, \bar{0} \rangle$ is a commutative monoid with identity element $\bar{0}$:
  - $a \oplus (b \oplus c) = (a \oplus b) \oplus c$
  - $a \oplus b = b \oplus a$
  - $a \oplus \bar{0} = \bar{0} \oplus a = a$

- $\langle K, \otimes, \bar{1} \rangle$ is a monoid with identity element $\bar{1}$
- $\otimes$ distributes over $\oplus$
  - $a \otimes (b \oplus c) = (a \otimes b) \oplus (a \otimes c)$
  - $(b \oplus c) \otimes a = (b \otimes a) \oplus (c \otimes a)$

- Element $\bar{0}$ annihilates $K$
  - $a \otimes \bar{0} = \bar{0} \otimes a = \bar{0}$.

Examples

- $\langle \mathbb{R}, +, \times, 0, 1 \rangle$ – real semiring
- $\langle S, \Delta, \cap, \emptyset, \cup; S_i \rangle$ – semiring of sets
source **fr**: Vénus est la jumelle infernale de la Terre

target **en**: Venus is Earth’s hellish twin

- Decomposability of $\bar{h}(e, f)$ into a sum of *local* features $h_{01}, h_{02}$...
- Envelopes are distributed over nodes in the lattice
MERT Semiring

\[ \mathbb{D} = \langle D, \oplus, \otimes, \bar{0}, \bar{1} \rangle \]

Host set:
- a line: \( d_y + d_s \cdot x \) (hypothesis)
- set of lines \( d_i \): \( d = \{ d_{i,y} + d_{i,s} \cdot x \} \) (set of hypotheses)
- set of sets \( d^k \) of lines: \( D = \{ \{ d_{i,y}^k + d_{i,s}^k \cdot x \} \} \)

Operations \( \oplus \) and \( \otimes \):
- for \( d^1, d^2 \in D \)
- \( d^1 \oplus d^2 = \text{env}(d^1 \cup d^2) \)
- \( d^1 \otimes d^2 = \text{env}(\{ (d_{i,y}^1 + d_{j,y}^2) + (d_{i,s}^1 + d_{j,s}^2) \cdot x \mid \forall d_i^1 \in d^1, d_j^2 \in d^2 \}) \)

Unities:
- \( \bar{0} = \emptyset \)
- \( \bar{1} = \{ 0 + 0 \cdot x \} \)
Semiring Operations Illustration

⊗-example

\[ d^1 \otimes d^2 = \text{env}(\{(d^1_{i.y} + d^2_{j.y}) + (d^1_{i.s} + d^2_{j.s}) \cdot x | \forall d^1_i \in d^1, d^2_j \in d^2\}) \]

⊕-example

\[ d^1 \oplus d^2 = \text{env}(d^1 \cup d^2) \]
Shortest Paths for MERT Semiring

Each arc in the FST carries:
- target word \( a \)
- vector \( \vec{h}(a, \mathbf{f}) \) of local features associated with \( a \)
- singleton set containing line \( d \) with
  - slope \( d_s = (\bar{r} \cdot \vec{h}(a, \mathbf{f})) \)
  - \( y \)-intercept \( d_y = (\bar{\lambda}_0 \cdot \vec{h}(a, \mathbf{f})) \)

Weight of a candidate translation path \( \mathbf{e} = e_1 \ldots e_\ell \):

\[
w(\mathbf{e}) = \bigotimes_{i=1}^{\ell} w(e_i) = \{ \bar{\lambda}_0 \cdot \sum_{i=1}^{\ell} \vec{h}(e_i, \mathbf{f}) + (\bar{r} \cdot \sum_{i=1}^{\ell} \vec{h}(e_i, \mathbf{f})) \} \cdot x\}
\]

Upper envelope of all the lines (hypotheses):

\[
\text{env} \left( \bigcup_{\mathbf{e}} w(\mathbf{e}) \right) = \bigoplus_{\mathbf{e}} w(\mathbf{e}) = \bigoplus_{\mathbf{e}} \bigotimes_{i=1}^{\ell} w(e_i).
\]

Generic shortest distance algorithms over acyclic graphs calculate this.
Implementation

- **Basics**: OpenFST toolbox
  - works with any semiring
  - proven and well optimized ShortestPath algorithms
  - other useful algorithms: Union, Determinize, etc.

- **Lattice minimization**:
  - Union of lattices between decoder runs
  - Determinize+Minimize to eliminate duplicate hypotheses
    won’t work – MERT semiring is not divisible
  - circumvent by performing Union+Determinize over \((\min,+)\) semiring

- **All directions simultaneously**
  - weights as arrays of envelopes
  - 20-30 random direction \(\simeq +0.3\text{-}0.5\) BLEU

- Random restarts help only for the first iteration
Experiments

Data:
- NewsCommentary (dev: 2051) & WMT10 (dev: 1026), common test
- French to English

FST MERT tuning:
- OpenFST-based multi-threaded implementation
- zero restart points
- axes and additional random directions

Baseline MERT tuning:
- MERT implementation included in MOSES toolkit
- 100-best list, 20 restart points
- Koehn’s coordinate descend (only axis directions)

Decoder: \( n \)-gram phrase-based SMT system N-code\(^1\), 11 features

\(^1\)Demo on http://ncode.limsi.fr/
Experiments

1. For the newsco dataset:
   - n-best MERT, dev
   - fst MERT (r=0), dev
   - fst MERT (r=20), dev

2. BLEU scores:
   - 15.0
   - 15.5
   - 16.0
   - 16.5
   - 17.0
   - 17.5
   - 18.0
   - 18.5

3. For WMT10:
   - n-best MERT, dev
   - fst MERT (r=0), dev
   - fst MERT (r=20), dev
   - fst MERT (r=50), dev

4. BLEU scores:
   - 21.0
   - 22.0
   - 23.0
   - 24.0
   - 25.0
   - 26.0

5. For the test set:
   - n-best MERT, test
   - fst MERT (r=0), test
   - fst MERT (r=20), test
   - fst MERT (r=50), test

Artem Sokolov & François Yvon (LIMSI)
Minimum Error Rate Training Semiring
EAMT’2011  16 / 18
Conclusion & Future Work

Conclusion
- Semiring formalization allows using generic FST toolkits to do MERT
- Convergence in less iterations

Future Work
- Better stopping criteria to detect saturation
- Faster $\oplus$ – should be most helpful for speed up
Thank you for your attention!