System Combination for Machine Translation

Xiaodong He
NLP, Microsoft Research

March 07, 2008
Outline

• MT system combination – a Bayesian perspective
• HMM based hypothesis alignment for MT system combination
• Experimental results
• Discussion
Minimum Bayes-Risk Classification

- Given the observation $F$ and a hypothesis $E'$
- Bayes-risk of classifying $F$ to $E'$
  - $R(E') = \sum_{E \in E_e} P(E | F)L(E', E)$
- MBR classification
  - $E^* = \arg \min_{E' \in E_h} \sum_{E \in E_e} P(E | F)L(E', E)$
- Key components of MBR
  - $P(E | F)$: posterior probability
  - $L(E', E)$: loss function, application specific
  - $E_h$: hypothesis space, for selecting classification candidate
  - $E_e$: evidence space, for computing Bayes-risk
N-best based MBR Classification

- N-best based MBR
  - Hypothesis and evidence space
    - \( E_h = E_e = \{E_1, ..., E_N\}^T \)
  - Loss function \( L(E', E) \)
    - Levenshtein distance (WER)
    - Translation Edit Rate (TER)
    - BLEU (?)
  - Posterior distribution \( P(E | F) \)
    - Normalized log-linear model scores
- Simple, but small \( E_h \) limits the performance
If hypotheses are segmented
○ i.e., $E_i = e_{i,1},..., e_{i,L}$, where $e_{i,l}$ can be a real word or null

and hypotheses are aligned
○ i.e., words of different hypotheses at position $l$, $e_{1,l}, e_{2,l},..., e_{N,l}$, are aligned to each other.
Denote by $e_l = \{e_{1,l},..., e_{N,l}\}$ the correspondence set.

and loss function is decomposable
○ i.e., $L(E_i, E_j) = \sum_l L(e_{i,l}, e_{j,l})$

then the hypothesis space is expanded to a confusion network (CN)
○ $E_h = e_1 \times e_2 \times ... \times e_L$
Illustrations of N-best List and CN

1) N-best from MT systems
   \( E_1: \) he have good car
   \( E_2: \) he has nice sedan
   \( E_3: \) it a nice car
   \( E_4: \) a sedan he has

2) Backbone selection
   \[
   E_B = \arg \min_{E' \in E_B} \sum_{E \in E_e} P(E | F) L(E', E)
   \]
   e.g., Backbone \( E_B: \) he have good car

3) Hypothesis alignment
   \( E_B: \) he have \( \varepsilon \) good car
   \( E_4: \) a \( \varepsilon \) sedan he has

4) Confusion network (\( L=5 \))

<table>
<thead>
<tr>
<th></th>
<th>have</th>
<th>( \varepsilon )</th>
<th>good</th>
<th>car</th>
</tr>
</thead>
<tbody>
<tr>
<td>he</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>he</td>
<td>has</td>
<td>( \varepsilon )</td>
<td>nice</td>
<td>sedan</td>
</tr>
<tr>
<td>it</td>
<td>( \varepsilon )</td>
<td>a</td>
<td>nice</td>
<td>car</td>
</tr>
<tr>
<td>he</td>
<td>has</td>
<td>a</td>
<td>( \varepsilon )</td>
<td>sedan</td>
</tr>
</tbody>
</table>

|   | \( e_1 \) | \( e_2 \) | \( e_3 \) | \( e_4 \) | \( e_5 \) |
The global risk can also be decomposed

\[ R(E') = \sum_{E \in E_e} P(E | F)L(E', E) \]

\[ = \sum_{E \in E_e} P(E | F) \sum_{i=1}^{L} L(e'_i, e_i) \]

\[ = \sum_{i=1}^{L} \sum_{e_i \in e_i} L(e'_i, e_i) \sum_{E: E \in E_e \& e_i \in E} P(E | F) \]

local posterior: \( P(e_i | F) \)

local risk: \( R(e'_i) \)

Minimizing global risk can be done by minimizing local risks
Segmental MBR (S-MBR)

- Segmental MBR (Goel and Byrne, 03)
  - Expand hypothesis space
  - Convert the global risk minimization problem to a series of local ones
  - S-MBR based classification is
    \[ E^* = e_1^*, \ldots, e_L^* \]
    where
    \[ e_l^* = \arg\min_{e_l'\in e_l} \sum_{e_l\in e_l} L(e_l', e_l) P(e_l | F) \]
    \[ P(e_l | F) = \sum_{E\in E_{e_l}} P(E | F) \]

Note: the local posterior probability, \( P(e_l | F) \), is computed across the whole span of \( E \) and \( F \).
System Combination for MT

- Combining multiple MT systems
  - The real MT hypothesis space is very large
    - Non-monotonic word ordering causes a big complexity issue for MT
  - Different MT paradigms exist
    - Syntax based, Phrasal based, Hiero based, Rule based …
    - Each of which may explore only a sub-space
    - They may produce different modeling and search errors
  - Combining multiple MT systems can greatly enrich the hypothesis space $E_h$
  - Also help robust estimation of $P(e_l | F)$, so as to mitigate modeling/search errors of individual systems
S-MBR for MT System Combination

- Major issues for S-MBR based MT system combination
  - Computing posterior probability
    - Need robust and efficient estimation of $P(e_l \mid F)$
  - Selecting a loss function
    - Not all desirable loss functions are decomposable
  - Constructing the confusion network
    - High quality CN is crucial to S-MBR
  - Decoding the confusion network
Compute Posterior Probability

- Global posterior probability: $P(E \mid F)$
  - Evidence space
    - Aggregation of N-best lists from all individual systems
  - Based on ranking and system weights
  - Formula (denote $S_k$ the $k$-th MT system)
    \[
    P(E \mid F) = \sum_{k=1}^{K} w_k P(E \mid S_k, F), \text{ where}
    \]
    \[
    P(E \mid S_k, F) = 0, \text{ if } E \text{ is not in the N-best list of system } S_k;
    \]
    \[
    P(E \mid S_k, F) = \frac{1/(1 + \text{Rank}(E, S_k))^{\eta}}{\sum_{E \in E_{S_k}} [1/(1 + \text{Rank}(E, S_k))]^{\eta}}, \text{ otherwise.}
    \]

- Local posterior probability: $P(e_l \mid F) = \sum_{E \in E_{e_l} \in E} P(E \mid F)$
Common Loss Functions for MT

- Criteria: reflects MT quality and is decomposable

- Conventional loss functions
  - **Levenshtein distance/Word Error Rate (WER)**
    - Don’t accommodate non-monotonic word ordering of MT
    - Decomposable at the word level
  - **Translation Edit Rate (TER)**
    - Measure number of Ins/Del/Sub/Shift
    - Good measure of MT quality
    - Not decomposable at the word level (due to shift)
  - **BLEU score**
    - Measure N-gram precision, plus a brevity penalty
    - Good measure of MT quality
    - Not decomposable at the word level
Loss Function for S-MBR/MT

- **Loss function for S-MBR: WER-after-alignment**
  - Measure WER after aligning hyp. to ref.
    - Two hypotheses with different word orders may both be valid
    - Shifts suggested by the alignment model are not counted as errors
  - Good measure of MT quality (similar to TER)
  - Decomposable at the word level

- **Compute loss** $L(E', E)$
  - Loss is measured after $E'$ and $E$ are aligned
  - Global loss is sum of local losses, i.e., $L(E', E) = \sum_l \delta(e_l', e_l)$. 
Construct Confusion Network

- High quality CN is essential
  - Hypothesis set is constrained by *correspondence set*, which is specified by the CN
  - Posterior probability calculation also depends on CN

- Procedure
  - Select an alignment backbone using sentence-level MBR
    
    \[ E_{BB} = \arg \min_{E' \in E_{BB}} \sum_{E \in E_e} P(E \mid F) L(E', E) \]
  - Align all hypotheses to that backbone
    - May need to add *null* in backbone (insertion error) or hypothesis (deletion error)
  - Convert all aligned hypotheses to a confusion network
MT hypothesis alignment

- MT hypothesis alignment is critical
  - Loss function depends on hypothesis alignment
  - CN is constructed by incrementally aligning hypotheses

- MT hypothesis alignment is difficult
  - MT outputs have non-monotonic word ordering
  - MT outputs have synonyms
  - It is a mono-lingual word alignment, but few mono-lingual parallel sentence data available
Previous Hypothesis Alignment Work

- GIZA++ based alignment (RWTH, Matusov et al. 06)
  - GIZA++ is a bilingual word alignment tool
    - Uses statistical models and unsupervised training
    - Models non-monotonic word-ordering and synonyms
    - Needs a large amount of parallel sentences as training data
  - GIZA++ for hypothesis alignment (monolingual alignment)
    - Uses pairs of hypothesis from N-best lists of the test set as parallel data for GIZA++
    - However, all hypothesis pairs of one sentence are derived from the same source. They are not i.i.d. samples
    - Moreover, MT hypotheses are very noisy
    - Therefore, the GIZA++ training maybe unreliable
Previous Hypothesis Alignment Work

- TER based alignment (BBN, Rosti et al. 07)
  - Align one hyp. to another such that the TER is minimized
    - TER counts *insertion, deletion, substitution,* and *shift*
    - No statistical model needed so no training required
    - Accommodates non-monotonic word ordering
  - Nice results are reported
  - However,
    - Deterministic edit counting when searching for alignment
    - No consideration of synonym, rely on exact lexicon match
    - Weak treatment for non-monotonic word ordering (zero-order modeling, i.e., shift)
MSR’s Hypothesis Alignment Model

- HMM based alignment (MSR, He et al. 08, forthcoming)
  - Use fine-grained statistical models
    - Uses 1st order hidden Markov model for non-monotonic word ordering
    - Models synonym, i.e., how likely that two different words have similar meanings and should be aligned to each other
  - Don’t need to “train” the model
    - In training, bilingual word alignment models are reliably trained on bilingual parallel training data
    - In testing, HMM for hypothesis alignment are derived from bilingual word alignment models
    - No mono-lingual parallel sentence data needed
  - Significant improvements were observed
HMM based Hypothesis Alignment

\[ E_B : e_1 \quad e_2 \quad e_3 \]

\[ E_{hyp} : e'_1 \quad e'_3 \quad e'_2 \]

- HMM is built on the backbone side
- HMM aligns the hypothesis to the backbone
- After alignment, a CN is built
HMM Parameter Estimation

- Emitting Probability (via words in source sentence)
  - $P(e'_1|e_1)$ models how likely $e'_1$ and $e_1$ have similar meanings.
  - Use the source word sequence $\{f_1, \ldots, f_M\}$ as a hidden layer, $P(e'_1|e_1)$ takes a mixture-model form, i.e.,
    \[
    P_{src}(e'_1|e_1) = \sum_m w_m P(e'_1|f_m)
    \]
    where $w_m = P(f_m|e_1)/\sum_m P(f_m|e_1)$
  - $P(e'_1|f_m)$ is from the bilingual word alignment model, $F \rightarrow E$ direction.
  - $P(f_m|e_1)$ is from that of $E \rightarrow F$.

\[
P(e'_1|e_1) = \sum_m w_m P(e'_1|f_m)
\]
Emitting Probability (via word surface similarity)

- Normalized similarity measure $s$
  - Based on Levenshtein distance
  - Based on matched prefix length
- Use an exponential mapping to get $P(e'_1|e_1)$
  \[ P_{\text{simi}}(e'_1|e_1) = \exp[\rho \cdot (s(e'_1, e_1) - 1)] \]
  $s(e'_1, e_1)$ is normalized to $[0,1]$

Overall Emitting Probability

\[ P(e'_1|e_1) = \alpha \cdot P_{\text{src}}(e'_1|e_1) + (1 - \alpha) \cdot P_{\text{simi}}(e'_1|e_1) \]
HMM Parameter Estimation (cont.)

- Transition Probability
  - $P(a_j|a_{j-1})$ models word ordering
    - Takes the same form as a bilingual word alignment HMM
    - Strongly encourages monotonic word ordering
    - Allows non-monotonic word ordering

$$d(i, i') = \left(1 + |i - i' - 1|\right)^{-\kappa}$$

$$P(a_j | a_{j-1}) = \frac{d(a_j, a_{j-1})}{\sum_i d(i, a_{j-1})}$$

$a_j$ – alignment of the $j$-th word
Find the Optimal Alignment

- Viterbi decoding:

\[
\hat{\alpha}_1^J = \arg \max_{a_1^J} \prod_{j=1}^J \left[ p(a_j \mid a_{j-1}, I) p(e'_j \mid e_{a_j}) \right]
\]

- Other variations:
  - posterior probability & threshold based decoding
  - max posterior mode decoding
Alignment Normalization

- Viterbi decoding generates the raw alignment
  - Contain 1-to-many alignments
  - Some words are aligned to *null*

- Normalize the raw alignment
  - Need 1-to-1 alignment
    - For 1-to-many alignment, only keep the link that gives the highest posterior probability, let other words aligned to *null*
  - Need to insert *null* at the right place
    - For words aligned to *null* due to the above 1-to-many alignment reduction, insert *null* around that ‘1’ word at the other side, such as keep the relative word order.
  - Other heuristics …
An example: HMM vs. TER alignment

REF: China’s fourteen border economic
TER: China’s fourteen border significant
HMM: China’s and fourteen opening up border economic

REF: achievements in construction
TER: achievements economic construction
HMM: achievements municipalities construction

REF: are remarkable
TER: and opening up municipalities
HMM: significant
Decode the Confusion Network

- **S-MBR decoding**
  - Pretty simple, e.g., with 0/1 local loss function,
    \[ e_l^* = \arg \min_{e_i' \in e_l} \sum_{e_j \in e_l} L(e_i', e_j) P(e_j | F) = \arg \max_{e_i' \in e_l} P(e_i' | F) \]
  - Need to control the output length
    - Discount \( P(null | F) \) to balance selecting *null* or a real word

Confusion network decoding (L=5)

<table>
<thead>
<tr>
<th>( e_1 )</th>
<th>( e_2 )</th>
<th>( e_3 )</th>
<th>( e_4 )</th>
<th>( e_5 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>he</td>
<td>have</td>
<td>( \varepsilon )</td>
<td>good</td>
<td>car</td>
</tr>
<tr>
<td>he</td>
<td>has</td>
<td>( \varepsilon )</td>
<td>nice</td>
<td>sedan</td>
</tr>
<tr>
<td>it</td>
<td>( \varepsilon )</td>
<td>a</td>
<td>nice</td>
<td>car</td>
</tr>
<tr>
<td>he</td>
<td>has</td>
<td>a</td>
<td>( \varepsilon )</td>
<td>sedan</td>
</tr>
</tbody>
</table>
Extend to Log-Linear Model

- Log-linear model based decoding (BBN, Rosti et al. 07)
  - Incorporate more features
    - E.g., extra language models, etc.
    
    \[
    E^* = \arg \max_{E' \in \mathcal{E}_h} \ln P(E' \mid F)
    \]
    
    where
    
    \[
    \ln P(E' \mid F) = \ln \prod_{l=1}^{L} P_{S-MBR}(e'_l \mid F) + \nu \ln P_{LM}(E') + \xi \mid E' \mid
    \]
  
  - Use minimum error rate training
    - Directly optimize the evaluation metric (e.g., BLEU)
    - Feature weights can be trained on a dev set
      \(\nu\): Language model score weight
      \(\xi\): word count feature weight
Experimental results

- Evaluation metric: BLEU-4
  - Geometric average of n-gram precision, higher is better
  - Usually 0.5% difference can be considered significant
  - \[ \text{BLEU-4}(E, E_r) = \exp \left( \frac{1}{4} \sum_{n=1}^{4} \log p_{n-\text{gram}}(E, E_r) \right) \cdot bp(E, E_r) \]

- Test conditions
  - Chinese-to-English constrained training data track
  - NIST MT open evaluation 2002 – 2008 test data
    - Test 02-05 are newswire data; 06 and 08 contain web data
  - Total ~7M parallel sentence pairs
  - 3G words monolingual (English) data for LM
Experimental results

Single system vs. S-MBR system combination

<table>
<thead>
<tr>
<th>Single System</th>
<th>BLEU-4 score</th>
</tr>
</thead>
<tbody>
<tr>
<td>MT1</td>
<td>21.75</td>
</tr>
<tr>
<td>MT2</td>
<td>20.42</td>
</tr>
<tr>
<td>MT3</td>
<td>21.69</td>
</tr>
<tr>
<td>MT4</td>
<td>24.57</td>
</tr>
<tr>
<td>MT5</td>
<td>24.40</td>
</tr>
<tr>
<td>MT6</td>
<td>25.52</td>
</tr>
<tr>
<td>MT7</td>
<td>25.51</td>
</tr>
<tr>
<td>MT8</td>
<td>26.24</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Combined system (8-way)</th>
<th>BLEU-4 score</th>
</tr>
</thead>
<tbody>
<tr>
<td>w/ uniform system weights</td>
<td>29.83</td>
</tr>
<tr>
<td>w/ trained system weights</td>
<td>30.01</td>
</tr>
</tbody>
</table>

Trained system weights

<table>
<thead>
<tr>
<th></th>
<th>MT1</th>
<th>MT2</th>
<th>MT3</th>
<th>MT4</th>
<th>MT5</th>
<th>MT6</th>
<th>MT7</th>
<th>MT8</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.13</td>
<td>0.06</td>
<td>0.12</td>
<td>0.12</td>
<td>0.09</td>
<td>0.14</td>
<td>0.12</td>
<td>0.22</td>
</tr>
</tbody>
</table>
Experimental results

- S-MBR vs. extended Log-linear model (w/ extra LMs)

<table>
<thead>
<tr>
<th>system</th>
<th>BLEU-4 score</th>
</tr>
</thead>
<tbody>
<tr>
<td>S-MBR</td>
<td>30.01</td>
</tr>
<tr>
<td>Log-Linear</td>
<td>30.89</td>
</tr>
<tr>
<td>Best single sys.</td>
<td>26.24</td>
</tr>
</tbody>
</table>

Note: no LM is used by S-MBR while the Log-Linear model uses two extra language models (3-gram and 5-gram)
Experimental results

- HMM vs. TER based hypothesis alignment

NIST MT02-06 overall test results

NIST MT08 test results

- TER vs. HMM based hypothesis alignment

BLEU score vs. number of systems

- MSR-MSRA 6-way comb.
- MSR-NRC-SRI 8-way comb.

Microsoft Research
NIST MT08 C2E at A Glance

Note: MSR-NRC-SRI includes MSR (both Redmond and Asia sites), NRC, and SRI. MSR-MSRA is a Microsoft-only entry including both MSR Redmond and Asia sites.
Post-Eval: Combining More Entries

- MSR-NRC-SRI + ISI-LW + BBN
  - Combining more systems gives further improvements
    - MSR-NRC-SRI: 8, ISI-LW: 3, BBN: 4
  - 15 single systems in total, BLEU scores from 20.42 to 29.85

Note: MSR-NRC-SRI, ISI-LW, and BBN are already system combination entries
Post-Eval: Unconstrained Training

- Extra GALE/FBIS data available (NRC & SRI only)
- Add more systems for combination

<table>
<thead>
<tr>
<th>System</th>
<th>Description</th>
<th>BLEU-4 score</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSR-NRC-SRI</td>
<td>(primary submission, constrained)</td>
<td>30.89</td>
</tr>
<tr>
<td>+ SRI2&amp;3</td>
<td>(improved system, constrained)</td>
<td>31.12</td>
</tr>
<tr>
<td>+ SRI-GALE1&amp;2 and NRC-GALE</td>
<td>(unconstrained)</td>
<td>31.96</td>
</tr>
<tr>
<td>Google unconstrained</td>
<td></td>
<td>31.95</td>
</tr>
</tbody>
</table>
US President Bush gave an address on the 16th in Washington, calling for the holding of an international conference on the Palestine-Israel issue this fall, and inviting Israel, Palestine and some of the surrounding Arab countries and other countries concerned to take part, in order to jointly push for the restarting of the Middle East peace process.

Machine Translation:

US President George W. Bush delivered a speech on the 16th in Washington, called for holding an international conference on the Palestine-Israel issue in the autumn, to invite Israel, Palestine, and some of the neighboring Arab countries and other countries concerned to make joint efforts to promote the Middle East peace process.
Conclusion

- System combination is powerful for MT
  - Built on top of segmental MBR, extended to log-linear model
  - High quality hypothesis alignment is crucial
  - 4.7 BLEU pt. gain over the best single system in our MT08 entry
  - Combining more systems gave further improvement

- It is still very coarse yet, lots of problems to explore
  - Many approximations involved
  - So far only treat MT as a sequential pattern problem, should we consider the structure of language?

- What we can learn for single MT system?