Out of GIZA—Efficient Word Alignment Models for SMT

Yanjun Ma

National Centre for Language Technology
School of Computing
Dublin City University

NCLT Seminar Series
March 4, 2009
Outline

1. Contexts

2. HMM and IBM Model 4

3. Improved HMM Alignment Models

4. Simultaneous Word Alignment and Phrase Extraction
Outline

1. Contexts

2. HMM and IBM Model 4

3. Improved HMM Alignment Models

4. Simultaneous Word Alignment and Phrase Extraction
Word Alignment and SMT

All SMT systems rely on word alignment

- Word-Based SMT
- Phrase-Based SMT
- Hiero, hierarchical SMT
- Syntax-Based SMT, i.e., tree-to-string, string-to-tree, tree-to-tree
Word Alignment and SMT

All SMT systems rely on word alignment

- Word-Based SMT
- Phrase-Based SMT
- Hiero, hierarchical SMT
- Syntax-Based SMT, i.e, tree-to-string, string-to-tree, tree-to-tree

Giza implementation of IBM model 4 is dominant
All SMT systems rely on word alignment

- Word-Based SMT
- Phrase-Based SMT
- Hiero, hierarchical SMT
- Syntax-Based SMT, i.e, tree-to-string, string-to-tree, tree-to-tree

Giza implementation of IBM model 4 is dominant

“Viterbi” alignment from IBM model 4 is used
1. Contexts

2. HMM and IBM Model 4

3. Improved HMM Alignment Models

4. Simultaneous Word Alignment and Phrase Extraction
Efficient Model: HMM Model [Vogel et al., 1996]

HMM emission (translation) model $p(t_j|s_{a_j})$
HMM transition (alignment) model $p(a_j | a_j - a_{j-1})$
HMM transition (alignment) model \( p(a_j|a_j - a_{j-1}) \)

\[
p(t, a|s) = \prod_j p(a_j|a_j - a_{j-1}) \cdot p(t_j|s_{a_j}) \tag{1}
\]
Deficient Model: IBM Model 3 and 4

Model 3: zero-order distortion model

Out of Giza
Model 4: first-order distortion model
Derivation

\[ P(t_1^J, a_1^J | s_1^I) = P(t_1^J, B_0^I | s_1^I) \] (2)
Derivation

\[ P(t_j^1, a_j^1 | s_i^1) = P(t_j^1, B_0^I | s_i^1) \]  \hspace{1cm} (2)

\[ = P(B_0 | B_1^I) \times \prod_{i=1}^{I} P(B_i | B_{i-1}^I, e_i^I) \times P(f_j^I | B_0^I, e_i^I) \]  \hspace{1cm} (3)
Derivation

\[ P(t_1^J, a_1^J | s_1^I) = P(t_1^J, B_0^I | s_1^I) \]  \hspace{1cm} (2)

\[ = P(B_0^I | B_1^I) \times \prod_{i=1}^{I} P(B_i | B_{i-1}^I, e_1^I) \times P(f_1^J | B_0^I, e_1^I) \]  \hspace{1cm} (3)

\[ = P(B_0^I | B_1^I) \times \prod_{i=1}^{I} p(B_i | B_{i-1}, e_i) \]  \hspace{1cm} (4)
Derivation

\[ P(t^J_1, a^J_1|s^I_1) = P(t^J_1, B^I_0|s^I_1) \]  \hspace{1cm} (2)

\[ = P(B_0|B^I_1) \times \prod_{i=1}^{I} P(B_i|B^{i-1}_1, e^I_1) \times P(f^J_1|B^I_0, e^I_1) \]  \hspace{1cm} (3)

\[ = P(B_0|B^I_1) \times \prod_{i=1}^{I} p(B_i|B_{i-1}, e_i) \]  \hspace{1cm} (4)

\[ \times \prod_{i=0}^{I} \prod_{j \in B_i} p(f_j|e_i) \]  \hspace{1cm} (5)
Model 3 fertility and distortion

\[ p(B_i | B_{i-1}, e_i) = p(\phi_i | e_i) \phi_i! \prod_{j \in B_i} p(j | i, J) \]  

(6)
Model 3 fertility and distortion

\[ p(B_i|B_{i-1}, e_i) = p(\phi_i|e_i) \phi_i! \prod_{j \in B_i} p(j|i, J) \]  

(6)

Model 4 fertility and distortion

\[ p(B_i|B_{i-1}, e_i) = p(\phi_i|e_i) p_{=1}(B_{i1} - \overline{B_{\rho(i)}}|\cdots) \prod_{k=2}^{\phi_i} p_{>1}(B_{ik} - B_{i,k-1}|\cdots) \]  

(7)
Decoding

HMM

- Viterbi decoding: $\hat{a} = \arg\max_a p(a|s, t)$
- Posterior decoding: Align point $a_j \rightarrow i$ iff. $p(a_j \rightarrow i|s, t) \geq \delta$
Decoding

HMM

- Viterbi decoding: \( \hat{a} = \arg\max_a p(a|s, t) \)
- Posterior decoding: Align point \( a_j \rightarrow i \) iff. \( p(a_j \rightarrow i|s, t) \geq \delta \)

IBM model 3 and 4

- No efficient algorithm available
Advantages of HMM models

Efficient parameter estimation algorithm: forward-backward algorithm (Baum-Welch algorithm)
Advantages of HMM models

Efficient parameter estimation algorithm: forward-backward algorithm (Baum-Welch algorithm)

Figure: Eric B. Baum (son of Leonard E. Baum, who was the inventor of the algorithm) and Lloyd R. Welch
Advantages of HMM models

Efficient parameter estimation algorithm: forward-backward algorithm (Baum-Welch algorithm)

Figure: Eric B. Baum (son of Leonard E. Baum, who was the inventor of the algorithm) and Lloyd R. Welch

The resulting posterior probabilities are useful
Disadvantages of standard HMM models

Objective is maximising the likelihood
Disadvantages of standard HMM models

Objective is maximising the likelihood

- There is no guarantee that the optimised parameters correspond to more accurate alignments
Disadvantages of standard HMM models

Objective is maximising the likelihood

- There is no guarantee that the optimised parameters correspond to more accurate alignments
- To complicate things (sometimes!) does help, e.g. IBM model 4
Outline

1. Contexts
2. HMM and IBM Model 4
3. Improved HMM Alignment Models
4. Simultaneous Word Alignment and Phrase Extraction
Improved HMM models

Two more sophisticated HMM models
- Segmental HMM model, word-to-phrase alignment model
- Constrained HMM model, agreement-guided alignment model
Introducing a segmentation model: segmental HMM
\[ P(t, a|s) = P(v^K_1, K, a^K_1, h^K_1, \phi^K_1|s) \]
\[ P(t, a | s) = P(\nu^K_1, K, a^K_1, h^K_1, \phi^K_1 | s) \]
\[ = P(K | J, s) \] (9)
\[ P(t, a|s) = P(v_1^K, K, a_1^K, h_1^K, \phi_1^K|s) \]
\[ = P(K|J, s) \text{ \_\_\_ \_ segmentation} \]
\[ \times P(a_1^K, \phi_1^K, h_1^K|K, J, s) \text{ \_\_\_\_ alignment-fertility} \]
\[ P(t, a|s) = P(v^K_1, K, a^K_1, h^K_1, \phi^K_1|s) \]
\[ = P(K|J, s) \] 
\[ \text{segmentation} \]
\[ \times P(a^K_1, \phi^K_1, h^K_1|K, J, s) \] 
\[ \text{alignment-fertility} \]
\[ \times P(v^K_1|a^K_1, \phi^K_1, h^K_1, K, J, s) \] 
\[ \text{translation} \]
$P(a^K_1, \phi^K_1, h^K_1 | K, J, s) = \prod_{k=1}^{K} P(a_k, h_k, \phi_k | a_{k-1}, \phi_{k-1}, h_{k-1}, K, J, s)$ (12)
\[ P(a_1^K, \phi_1^K, h_1^K | K, J, s) = \prod_{k=1}^{K} P(a_k, h_k, \phi_k | a_{k-1}, \phi_{k-1}, h_{k-1}, K, J, s) \quad (12) \]

\[ = \prod_{k=1}^{K} p(a_k, | a_{k-1}, h_k; I) \cdot d(h_k) \cdot n(\phi_k; s_{a_k}) \quad (13) \]
MTTK implementation
Performance of HMM Word-to-Phrase Alignment

MTTK implementation

Used by Cambridge University Engineering Department

- Arabic–English NIST 2008 (6th out of 16, third best university participant, behind LIUM and ISI)
- Consistent performance for Chinese–English for differently sized collections of corpus
- Parallelised to handle large amount of data (e.g. 10M sentence pairs)
Agreement Constrained HMM Alignment
[Ganchev et al., 2008]

Objective

\[
\underset{q(a) \in Q}{\text{argmin}} \{ KL(q(a) \| p_\theta(a|s,t)) \} \quad \text{s.t.} \quad E_q[f(s,t,a)] \leq b
\]  

(14)
Agreement Constrained HMM Alignment
[Ganchev et al., 2008]

Objective

\[
\arg\min_{q(a) \in (Q)} \left\{ KL(q(a) \parallel p_{\theta}(a \mid s, t)) \right\} \text{ s.t. } E_q[f(s, t, a)] \leq b
\]  

Figure: \( \overrightarrow{p}_{\theta}(a \mid s, t), \overleftarrow{p}_{\theta}(a \mid s, t) \) and \( \overrightarrow{q}(a), \overleftarrow{q}(a) \)
Agreement Constrained HMM Alignment
[Ganchev et al., 2008]

Constrained E(M)

\[ \theta \quad q \]

- Regular E-Step
- E-Step
- M-Step

\[ Q \]

- Desired Posterior
Performance of Agreement Constrained HMM

PostCAT implementation

Evaluation
- Six language pairs, from 100,000 to 1M sentence pairs
- Outperform IBM Model 4 (16 out 18 times)
- However... getting slightly worse when the training data is over 1M
Algorithm 1 Agreement Constrained HMM Alignment

1: $\lambda_{ij} \leftarrow \forall i, j$
2: for T iterations do
3: $\theta'_t(t_j|s_i) \leftarrow \theta_t(t_j|s_i)e^{\lambda_{ij}} \forall i, j$
4: $\theta'_t(s_i|t_j) \leftarrow \theta_t(s_i|t_j)e^{-\lambda_{ij}} \forall i, j$
5: $\overleftarrow{q} \leftarrow \text{forwardBackward}(\theta'_t, \theta_a)$
6: $\overrightarrow{q} \leftarrow \text{forwardBackward}(\theta'_t, \theta_a)$
7: end for
8: return ($\overleftarrow{q}, \overrightarrow{q}$)
Outline

1. Contexts
2. HMM and IBM Model 4
3. Improved HMM Alignment Models
4. Simultaneous Word Alignment and Phrase Extraction
Phrase Pair Extraction

State-of-the-art: using viterbi alignment only

<table>
<thead>
<tr>
<th></th>
<th>Journal</th>
<th>officiel</th>
<th>des</th>
<th>Communautés</th>
<th>européennes</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Official</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>journal</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>of</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>the</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>European</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Communities</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Phrase Pair Extraction

State-of-the-art: using viterbi alignment only

Using all possible alignments

\[ A(i_1, i_2; j_1, j_2) = \{ a = a^I_j : a_j \in [i_1, i_2] \text{ iff. } j \in [j_1, j_2] \} \] (15)
Derivation

\[
P(t, A(i_1, i_2; j_1, j_2)|s; \theta) = \sum_{a \in A(i_1, i_2; j_1, j_2)} P(t, a|s; \theta)
\] (16)
Derivation

\[ P(t, A(i_1, i_2; j_1, j_2) \mid s; \theta) = \sum_{a \in A(i_1, i_2; j_1, j_2)} P(t, a \mid s; \theta) \]  \hspace{1cm} (16) 

\[ P(A(i_1, i_2; j_1, j_2) \mid s, t; \theta) = \frac{P(t, A(i_1, i_2; j_1, j_2) \mid s; \theta)}{P(t, a \mid s; \theta)} \]  \hspace{1cm} (17)
Evaluation

- Significant gains when used as an augmentation to the original phrase extraction strategy
