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

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

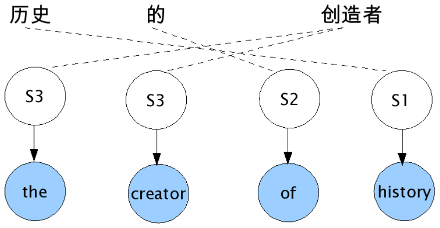
Giza implementation of IBM model 4 is dominant

“Viterbi” alignment from IBM model 4 is used

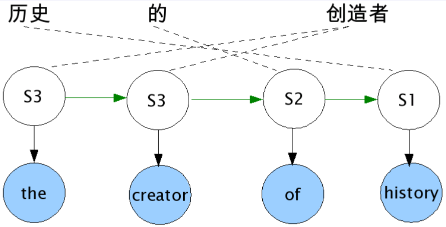
Outline

Efficient Model: HMM Model [Vogel et al., 1996]

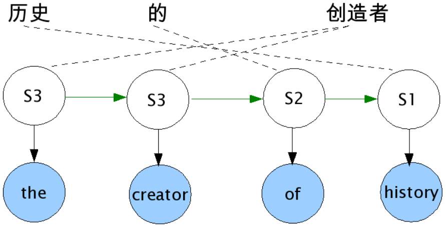
HMM emission (translation) model $p(t_j|s_{a_j})$



HMM transition (alignment) model $p(a_j|a_j - a_{j-1})$



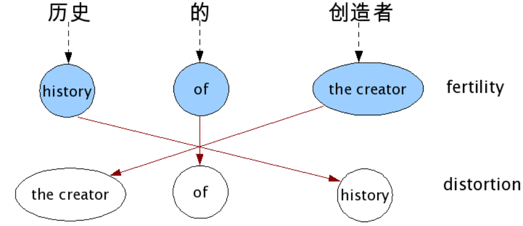
HMM transition (alignment) model $p(a_j|a_j - a_{j-1})$



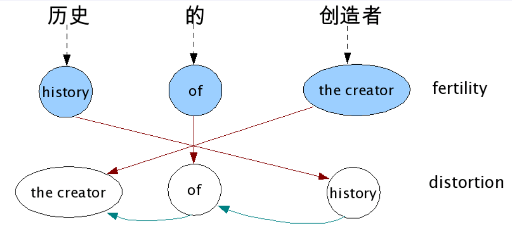
$$p(t, a|s) = \prod_j p(a_j|a_j - a_{j-1}) \cdot p(t_j|s_{a_j}) \tag{1}$$

Deficient Model: IBM Model 3 and 4

Model 3: zero-order distortion model



Model 4: first-order distortion model



Derivation

$$P(t_1^J, a_1^J | s_1^I) = P(t_1^J, B_0^I | s_1^I) \tag{2}$$

Derivation

$$P(t_1^J, a_1^J | s_1^I) = P(t_1^J, B_0^I | s_1^I) \quad (2)$$

$$= P(B_0 | B_1^I) \times \prod_{i=1}^{I} P(B_i | B_1^{i-1}, e_1^I) \times P(f_1^J | B_0^I, e_1^I) \quad (3)$$

Derivation

$$P(t_1^J, a_1^J | s_1^I) = P(t_1^J, B_0^I | s_1^I) \tag{2}$$

$$= P(B_0 | B_1^I) \times \prod_{i=1}^{I} P(B_i | B_1^{i-1}, e_1^I) \times P(f_1^J | B_0^I, e_1^I) \tag{3}$$

$$= P(B_0 | B_1^I) \times \prod_{i=1}^{I} \underbrace{p(B_i | B_{i-1}, e_i)}_{\text{fertility-distortion}} \tag{4}$$

Derivation

$$
\begin{aligned}
P(t_1^J, a_1^J | s_1^I) &= P(t_1^J, B_0^I | s_1^I) && (2)\\
&= P(B_0|B_1^I) \times \prod_{i=1}^{I} P(B_i|B_1^{i-1}, e_1^I) \times P(f_1^J|B_0^I, e_1^I) && (3)\\
&= P(B_0|B_1^I) \times \prod_{i=1}^{I} \underbrace{p(B_i|B_{i-1}, e_i)}_{\text{fertility-distortion}} && (4)\\
&\quad \times \prod_{i=0}^{I} \prod_{j \in B_i} \underbrace{p(f_j|e_i)}_{\text{translation}} && (5)
\end{aligned}
$$

Model 3 fertility and distortion

$$p(B_i|B_{i-1}, e_i) = \underbrace{p(\phi_i|e_i)}_{\text{fertility}} \phi_i! \prod_{j \in B_i} \underbrace{p(j|i, J)}_{\text{distortion}} \tag{6}$$

Model 3 fertility and distortion

$$p(B_i|B_{i-1}, e_i) = \underbrace{p(\phi_i|e_i)}_{\text{fertility}} \phi_i! \prod_{j \in B_i} \underbrace{p(j|i, J)}_{\text{distortion}} \tag{6}$$

Model 4 fertility and distortion

$$p(B_i|B_{i-1}, e_i) = \underbrace{p(\phi_i|e_i)}_{\text{fertility}} \underbrace{p_{=1}(B_{i1} - \overline{B_{\rho(i)}}|\cdots)}_{\text{first word}} \underbrace{\prod_{k=2}^{\phi_i} p_{>1}(B_{ik} - B_{i,k-1}|\cdots)}_{\text{remaining words}} \tag{7}$$

Decoding

HMM

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

Decoding

HMM

- Viterbi decoding: $\hat{a} = \underset{a}{\text{argmax}} p(a|s,t)$
- 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 garantee that the optimised parameters correspond to more accurate alignments

Disadvantages of standard HMM models

Objective is maximising the likelihood

- There is no garantee 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

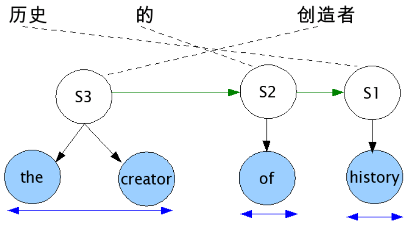
Improved HMM models

Two more sophisticated HMM models

- Segmental HMM model, word-to-phrase alignment model
- Constrained HMM model, agreement-guided alignment model

HMM Word-to-Phrase Alignment [Deng and Byrne, 2008]

Introducing a segmentation model: segmental HMM



$$P(t, a|s) = P(v_1^K, K, a_1^K, h_1^K, \phi_1^K|s) \tag{8}$$

$$
\begin{aligned}
P(t, a|s) &= P(v_1^K, K, a_1^K, h_1^K, \phi_1^K | s) && (8) \\
&= \underbrace{P(K|J, s)}_{\text{segmentation}} && (9)
\end{aligned}
$$

$$
\begin{aligned}
P(t, a|s) &= P(v_1^K, K, a_1^K, h_1^K, \phi_1^K | s) && (8)\\
&= \underbrace{P(K|J, s)}_{\text{segmentation}} && (9)\\
&\quad \times \underbrace{P(a_1^K, \phi_1^K, h_1^K | K, J, s)}_{\text{alignment-fertility}} && (10)
\end{aligned}
$$

$$
\begin{aligned}
P(t,a|s) &= P(v_1^K, K, a_1^K, h_1^K, \phi_1^K | s) && (8)\\
&= \underbrace{P(K|J,s)}_{\text{segmentation}} && (9)\\
&\quad \times \underbrace{P(a_1^K, \phi_1^K, h_1^K | K, J, s)}_{\text{alignment-fertility}} && (10)\\
&\quad \times \underbrace{P(v_1^K | a_1^K, \phi_1^K, h_1^K, K, J, s)}_{\text{translation}} && (11)
\end{aligned}
$$

HMM Word-to-Phrase Alignment

$$P(a_1^K, \phi_1^K, h_1^K | K, J, s) = \prod_{k=1}^{K} P(a_k, h_k, \phi_k | a_{k-1}, \phi_{k-1}, h_{k-1}, K, J, s) \quad (12)$$

HMM Word-to-Phrase Alignment

$$P(a_1^K, \phi_1^K, h_1^K | K, J, s) = \prod_{k=1}^{K} P(a_k, h_k, \phi_k | a_{k-1}, \phi_{k-1}, h_{k-1}, K, J, s) \quad (12)$$

$$= \prod_{k=1}^{K} \underbrace{p(a_k, | a_{k-1}, h_k; I)}_{\text{alignment}} \cdot \underbrace{d(h_k)}_{\text{null alignment}} \cdot \underbrace{n(\phi_k; s_{a_k})}_{\text{fertility}} \quad (13)$$

Performance of HMM Word-to-Phrase Alignment

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

$$\underset{q(a)\in(Q)}{\operatorname{argmin}}\{KL(q(a)||p_\theta(a|s,t))\} \text{ s.t. } E_q[f(s,t,a)] \leq b \tag{14}$$

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

Objective

$$\underset{q(a)\in(Q)}{\operatorname{argmin}}\{KL(q(a)||p_\theta(a|s,t))\} \text{ s.t. } E_q[f(s,t,a)] \leq b \tag{14}$$

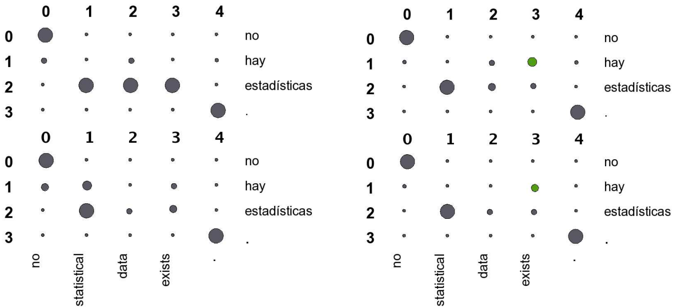
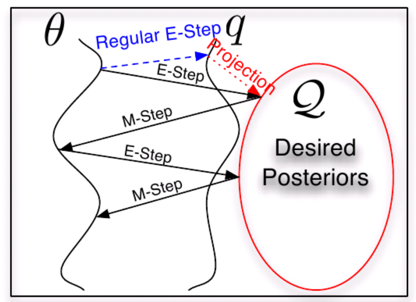


Figure: $\overrightarrow{p}_\theta(a|s,t)$, $\overleftarrow{p}_\theta(a|s,t)$ and $\overrightarrow{q}(a)$, $\overleftarrow{q}_{(}a)$

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

Constrained E(M)



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

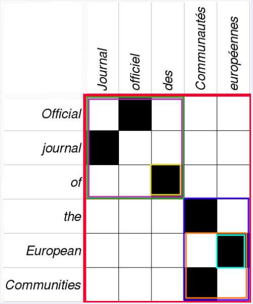
Algorithm 1 Agreement Constrained HMM Alignment

$\lambda_{ij} \leftarrow \forall i, j$
for T interations **do**
  $\overrightarrow{\theta}'_t(t_j|s_i) \leftarrow \overrightarrow{\theta}_t(t_j|s_i)e^{\lambda_{ij}} \forall i, j$
  $\overleftarrow{\theta}'_t(s_i|t_j) \leftarrow \overleftarrow{\theta}_t(s_i|t_j)e^{-\lambda_{ij}} \forall i, j$
  $\overrightarrow{q} \leftarrow \text{forwardBackward}(\overrightarrow{\theta}'_t, \overrightarrow{\theta}_a)$
  $\overleftarrow{q} \leftarrow \text{forwardBackward}(\overleftarrow{\theta}'_t, \overleftarrow{\theta}_a)$
end for
return $(\overrightarrow{q}, \overleftarrow{q})$

Outline

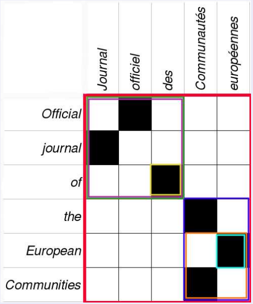
Phrase Pair Extraction

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



Phrase Pair Extraction

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



Using all possible alignments

$$A(i_1, i_2; j_1, j_2) = \{a = a_1^J : a_j \in [i_1, i_2] \text{ iff. } j \in [j_1, j_2]\} \tag{15}$$

Phrase Pair Extraction via Model-Based Posteriors

Derivation

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

Phrase Pair Extraction via Model-Based Posteriors

Derivation

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

$$P(A(i_1, i_2; j_1, j_2)|s, t; \theta) = \frac{P(t, A(i_1, i_2; j_1, j_2)|s; \theta)}{P(t, a|s; \theta)} \quad (17)$$

Evaluation

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

References

Deng, Y. and Byrne, W. (2008).
HMM word and phrase alignment for statistical machine translation.
IEEE Transactions on Audio, Speech, and Language Processing, 16(3):494–507.

Ganchev, K., Graca, J., and Taskar, B. (2008).
Better alignments = better translations?
In *Proceedings of the 46th Annual Meeting of the Association of Computational Linguistics*, Columbus, OH.

Vogel, S., Ney, H., and Tillmann, C. (1996).
HMM-based word alignment in statistical translation.
In *Proceedings of the 16th International Conference on Computational Linguistics*, pages 836–841, Copenhagen, Denmark.