

[illegible]

which of the following phrase pairs are consistent with the alignment, which are not?

1. (michael assumes , michael geht davon aus)
2. (michael assumes , michael get davon aus)
3. (michael assumes , michael geht davon aus , dass)
4. (he will stay , er im Haus bleibt)
5. (he will stay in the house , er im Haus bleibt)
6. (stay in the house , im Haus bleibt)



3. Scoring phrases: given a phrase pair (\bar{e}, \bar{f}) , how can you estimate $P(\bar{e}, \bar{f})$ from data using MLE and counts?

4. PB-SMT: draw a map of basic PB-SMT: which is the translation, the reordering and the language model in the formula below:

$$e_{\text{best}} = \operatorname{argmax}_e \prod_{i=1}^I \phi(\bar{f}_i | \bar{e}_i) d(\text{start}_i - \text{end}_{i-1} - 1) \prod_{i=1}^{|\bar{e}|} p_{LM}(e_i | e_1 \dots e_{i-1})$$



5. PB-SMT: in your own words, what are the main differences between IBM Model 3 and basic PB-SMT?

$$P(a, f | e) = \binom{m - \varphi_0}{\varphi_0} \times p_0^{(m - 2\varphi_0)} \times p_1^{\varphi_0} \times \prod_{i=1}^l n(\varphi_i | e_i) \times \prod_{j=1}^m t(f_j | e_{a_j}) \times \prod_{j: a_j \neq 0}^m d(j | a_j, l, m) \times \prod_{i=0}^l \varphi_i! \times \frac{1}{\varphi_0!}$$

$$e_{\text{best}} = \operatorname{argmax}_e \prod_{i=1}^I \phi(\bar{f}_i | \bar{e}_i) d(\text{start}_i - \text{end}_{i-1} - 1) \prod_{i=1}^{|\bar{e}|} p_{LM}(e_i | e_1 \dots e_{i-1})$$

Think about: words, phrases, NULL, fertility, what are the independence assumptions in each?



6. Logarithms: in your own words, explain $\log_a(b)$. What happens to probabilities in log-space? What are $\log(1)$ and $\log(0)$? If you want to maximise a probability, what do you have to do with the corresponding log, what would you have to do with the corresponding negative of the log? Can you express the log of a product as a sum? What is $\log(x^y)$ and why? Given $\log_e(x)$ what is its inverse function?

7. PB-SMT: in your own words, how are the following related:

$$e_{\text{best}} = \operatorname{argmax}_e \prod_{i=1}^I \phi(\bar{f}_i | \bar{e}_i) d(\text{start}_i - \text{end}_{i-1} - 1) \prod_{i=1}^{|\mathbf{e}|} p_{LM}(e_i | e_1 \dots e_{i-1})$$

$$e_{\text{best}} = \operatorname{argmax}_e \prod_{i=1}^I \phi(\bar{f}_i | \bar{e}_i)^{\lambda_\phi} d(\text{start}_i - \text{end}_{i-1} - 1)^{\lambda_d} \prod_{i=1}^{|\mathbf{e}|} p_{LM}(e_i | e_1 \dots e_{i-1})^{\lambda_{LM}}$$



$$p(x) = \exp \sum_{i=1}^n \lambda_i h_i(x)$$

$$p(e, a | f) = \exp(\lambda_\phi \sum_{i=1}^I \log \phi(\bar{f}_i | \bar{e}_i) + \lambda_d \sum_{i=1}^I \log d(a_i - b_{i-1} - 1) + \lambda_{LM} \sum_{i=1}^{|\mathbf{e}|} \log p_{LM}(e_i | e_1 \dots e_{i-1}))$$

8. Distance-based reordering: given the following simple definition of PB-SMT:

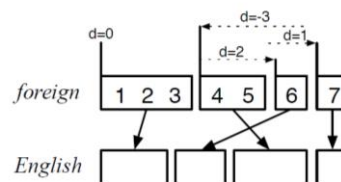
$$e_{\text{best}} = \operatorname{argmax}_e \prod_{i=1}^I \phi(\bar{f}_i | \bar{e}_i) d(\text{start}_i - \text{end}_{i-1} - 1) \prod_{i=1}^{|\mathbf{e}|} p_{LM}(e_i | e_1 \dots e_{i-1})$$

with simple distance based reordering:

$$d(\text{start}_i - \text{end}_{i-1} - 1)$$

in your own words please describe:

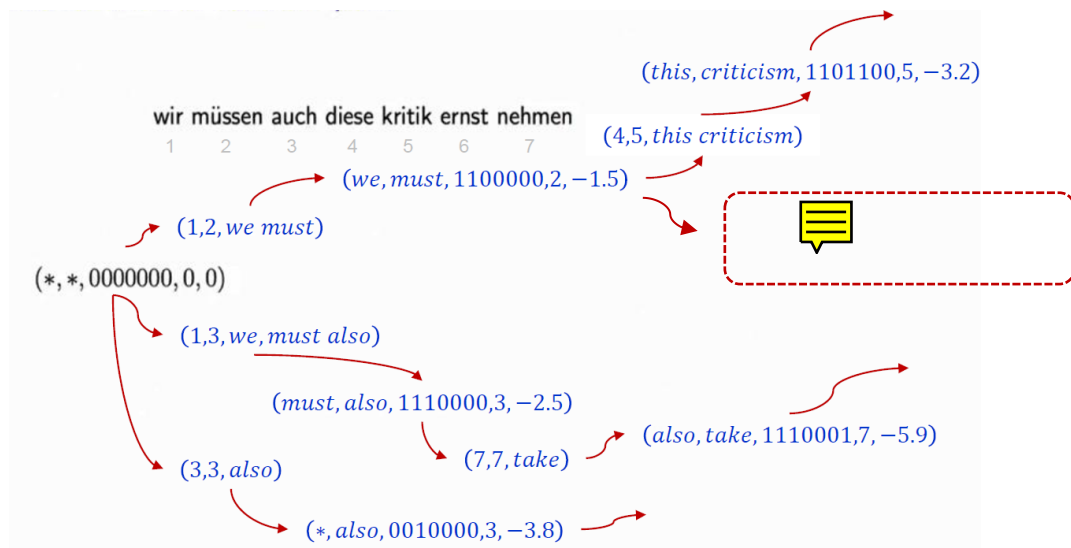
Distance-Based Reordering



phrase	translates	movement	distance
1	1-3	start at beginning	0
2	6	skip over 4-5	+2
3	4-5	move back over 4-6	-3
4	7	skip over 6	+1

Scoring function: $d(x) = \alpha^{|x|}$ — exponential with distance

9. PB-SMT decoder: please extend the following partial decoder graph at the position indicated in the red dashed rectangle below



with $(3, 3, \text{also})$ and compute the next state $(_, _, _, _, _)$. (You can make up the cost score!)

In your own words, what information do the slots in state quintuples $(_, _, _, _, _)$ capture?

What would state representation tuples look like if instead of a 3-gram LM we had used a 5-gram LM?

10. PB-SMT Decoder: in your own words, please describe how the sets Q_i (in blue on the right) evolve during decoding, given the decoder pseudo code on the left:

The Decoding Algorithm

$x_1 x_2 \dots x_7$

- Inputs: sentence $x_1 \dots x_n$. Phrase-based model $(\mathcal{L}, h, d, \eta)$. The phrase-based model defines the functions $ph(q)$ and $next(q, p)$.
- Initialization: set $Q_0 = \{q_0\}$, $Q_i = \emptyset$ for $i = 1 \dots n$.
- For $i = 0 \dots n - 1$
 - For each state $q \in \text{beam}(Q_i)$, for each phrase $p \in ph(q)$:
 - $q' = next(q, p)$
 - Add (Q_i, q', q, p) where $i = \text{len}(q')$
- Return: highest scoring state in Q_n . Backpointers can be used to find the underlying sequence of phrases (and the translation).

$q_0 = (*, *, 0000000, 0, 0)$

Michael Collins' slides
+ some explanations

\mathcal{L} = phrase table
 h = lang. model
 d = distortion lim.
 η = dist. parameter

$Q_0 = \{q_0\}$
 $Q_1 = \{ \dots, \dots, \dots \}$
 $Q_2 = \left\{ \begin{pmatrix} \dots, 1100000, \dots \\ \dots, 1010000, \dots \\ \dots, 1000100, \dots \end{pmatrix} \right\}$
 \dots
 $Q_3 = \{ \dots, \dots, \dots \}$
 $Q_4 = \{ \dots, \dots, \dots \}$
 $Q_5 = \{ \dots, \dots, \dots \}$
 $Q_6 = \{ \dots, \dots, \dots \}$
 $Q_7 = \left\{ \begin{pmatrix} \dots, 1111111, \dots \\ \dots, 1111111, \dots \\ \dots, 1111111, \dots \end{pmatrix} \right\}$
 \dots



In particular, what does index i in Q_i capture?



11. PB-SMT Decoder: please explain why we use beam search in the decoder? Can you describe two forms of beam search?