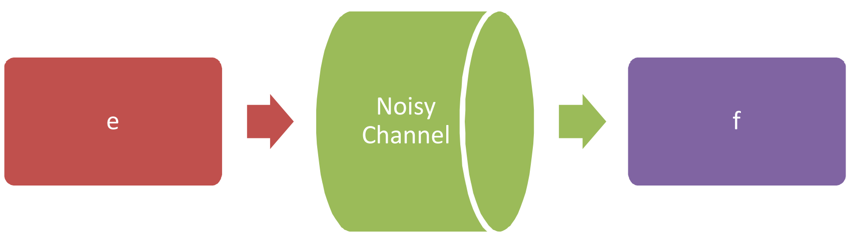
Statistical Machine Translation
IBM Model 1
CS626/CS460

Anoop Kunchukuttan
anoopk@cse.iitb.ac.in
Under the guidance of Prof. Pushpak Bhattacharyya

Why Statistical Machine Translation?

- Not scalable to build rule based systems between every pair of languages as in transfer based systems
  - Can translation models be learnt from data?
- Many language phenomena and language divergences which cannot be encoded in rules
  - Can translation patterns be memorized from data?

Noisy Channel Model



- Depicts model of translation from sentence *f* to sentence *e.*
- Task is to recover *e* from noisy *f.*

$$\hat{\mathbf{e}} = \underset{\mathbf{e}}{\operatorname{argmax}} \Pr(\mathbf{e}) \Pr(\mathbf{f}|\mathbf{e})$$

- *P(f|e)*: Translation model
  - Addresses adequacy
- *P(e)*: Language model
  - addresses fluency

Three Aspects

- Modelling
  - Propose a probabilistic model for sentence translation
- Training
  - Learn the model parameters from data
- Decoding
  - Given a new sentence, use the learnt model to translate the input sentence

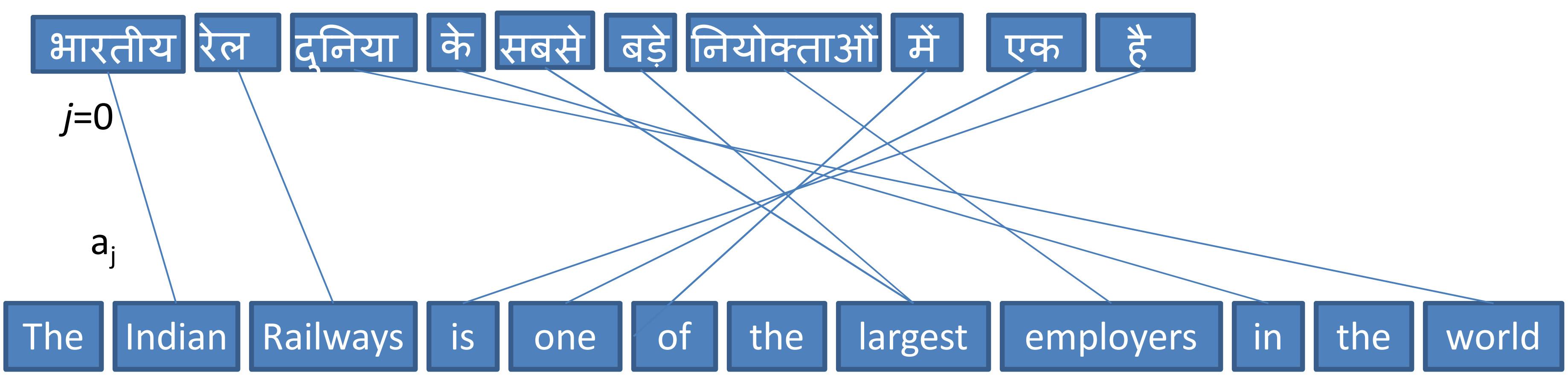
IBM Models 1 to 5 [1] define various generative models, and their training procedures.

This process serves as the basis for IBM Models 1 and 2

Generative Process 1

- Given sentence **e** of length *l*
- Select the length of the sentence **f,** say *m*
- For each position *j* in **f**
  - Choose the position a_j to align in sentence **e**
  - Choose the word f_j

$$\Pr(\mathbf{f},\mathbf{a}|\mathbf{e}) = \Pr(m|\mathbf{e})\prod_{j=1}^{m}\Pr(a_j|a_1^{j-1},f_1^{j-1},m,\mathbf{e})\Pr(f_j|a_1^{j},f_1^{j-1},m,\mathbf{e})$$



Alignments

- The generative process explains only one way of generating a sentence pair
  - Each way corresponds to an alignment
- Total probability of the sentence pair is the sum of probability over all alignments

$$\Pr(\mathbf{f}|\mathbf{e}) = \sum_{\mathbf{a}} \Pr(\mathbf{f}, \mathbf{a}|\mathbf{e})$$

- Input: Parallel sentences 1…S in languages ***E*** and ***F***
- But alignments are not known
- Goal: Learn the model P(**f**|**e**)

IBM Model 1

- Is a special case of Generative Process 1

$$\Pr(\mathbf{f}, \mathbf{a}|\mathbf{e}) = \frac{\epsilon}{(l+1)^m} \prod_{j=1}^{m} t(f_j|e_{a_j})$$

- Assumptions:
  - Uniform distribution for length of **f**
  - All alignments are equally likely
- Goal: Learn parameters t(f|e) for model P(**f**|**e**)

  for all f є ***F*** and e є ***E***
- Chicken and egg situation w.r.t
  - Alignments
  - Word translations

Model 1 Training

- If the alignments are known, the translation probabilities can be calculated simply by counting the aligned words.
- But, if translation probabilities were not known then the alignments could be estimated.
- **We know neither!**
- Suggests an iterative method where the alignments and translation method are refined over time.
- It is the **Expectation-Maximization** Algorithm

Model 1 Training Algorithm

Initialize all t(f|e) to any value in [0,1].
Repeat the E-step and M-step till t(f|e) values converge

c(f|e) is the expected count that f and e are aligned

E-Step

- **for** each sentence in training corpus
  - **for** each f,e pair : Compute c(f|e;f(s),e(s))
  - Use t(f|e) values from previous iteration

$$c(f|e;\mathbf{f},\mathbf{e}) = \frac{t(f|e)}{t(f|e_0)+\cdots+t(f|e_l)} \underbrace{\sum_{j=1}^{m}\delta(f,f_j)}_{\text{count of } f \text{ in } \mathbf{f}} \overbrace{\sum_{i=0}^{l}\delta(e,e_i)}^{\text{count of } e \text{ in } \mathbf{e}}$$

M-Step

- for each f,e pair: compute t(f|e)
- Use the c(f|e) values computed in E-step

$$t(f|e) = \lambda_e^{-1}\sum_{s=1}^{S} c(f|e;\mathbf{f}^{(s)},\mathbf{e}^{(s)}).$$

$$\lambda_e = \sum_{s=1}^{S}\sum_{\text{f in Vocab(F)}} c(f|e;\mathbf{f}^{(s)},\mathbf{e}^{(s)})$$

Let’s train Model 1

Corpus

- आकाश बैंक जाने के रस्ते पर चला
- Akash walked on the road to the bank

- श्याम नदी तट पर चला
- Shyam walked on the river bank

- आकाश द्वारा नदी तट से रेट की चोरी हो रही है
- Sand on the banks of the river is being stolen by Akash

Stats

- 3 sentences
- English (e) vocabulary size: 15
- Hindi (f) vocabulary size: 18

Model 1 in Action

c(f|e)	sentence	Iteration 1	Iteration 2	Iteration 5	Iteration 19	Iteration 20
आकाश|akash	1	0.066	0.083	0.29	0.836	0.846
आकाश|akash	2	0	0	0	0	0
आकाश|akash	3	0.066	0.083	0.29	0.836	0.846
बैंक|bank	1	0.066	0.12	0.09	0.067	0.067
बैंक|bank	2	0	0	0	0	0
बैंक|bank	3	0	0	0	0	0

t(f|e)		Iteration 1	Iteration 2	Iteration 5	Iteration 19	Iteration 20
आकाश|akash		0.125	0.1413	0.415	0.976	0.976
बैंक|bank		0.083	0.1	0.074	0.049	0.049
तट|bank		0.083	0.047	0.019	0.002	0.002
तट|river		0.142	0.169	0.353	0.499	0.499

Where did we get the Model 1 equations from?

- See the presentation model1_derivation.pdf, for more on parameter training

IBM Model 2

- Is a special case of Generative Process 1

$$\Pr(\mathbf{f}, \mathbf{a}|\mathbf{e}) = \epsilon \prod_{j=1}^{m} t(f_j|e_{a_j}) a(a_j|j, m, l)$$

- Assumptions:
  - Uniform distribution for length of **f**
  - ~~All alignments are equally likely~~

Model 2 Training Algorithm

Training process as in Model 1, except that equations become messier!

Initialize all t(f|e) and and a(i|j,m,l) to any value in [0,1].
Repeat the E-step and M-step till t(f|e) values converge

E-Step

- **for** each sentence in training corpus
  - **for** each f,e pair : Compute c(f|e;f(s),e(s)) and c(i|j,m,l)
  - Use t(f|e) and a(i|j,m,l) values from previous iteration

$$c(f|e;\mathbf{f},\mathbf{e}) = \sum_{j=1}^{m}\sum_{i=0}^{l} \frac{t(f|e)\,a(i|j,m,l)\,\delta(f,f_j)\delta(e,e_i)}{t(f|e_0)\,a(0|j,m,l)+\cdots+t(f|e_l)\,a(l|j,m,l)}$$

$$c(i|j,m,l;\mathbf{f},\mathbf{e}) = \frac{t(f_j|e_i)\,a(i|j,m,l)}{t(f_j|e_0)\,a(0|j,m,l)+\cdots+t(f_j|e_l)\,a(l|j,m,l)}$$

M-Step

- for each f,e pair: compute t(f|e)
- Use the c(f|e) and c(i|j,m,l) values computed in E-step

$$t(f|e) = \lambda_e^{-1}\sum_{s=1}^{S} c(f|e;\mathbf{f}^{(s)},\mathbf{e}^{(s)}).$$

$$\lambda_e = \sum_{s=1}^{S}\sum_{\text{f in Vocab(F)}} c(f|e;\mathbf{f}^{(s)},\mathbf{e}^{(s)})$$

$$a(i|j,m,l) = \mu_{jml}^{-1}\sum_{s=1}^{S} c(i|j,m,l;\mathbf{f}^{(s)},\mathbf{e}^{(s)})$$

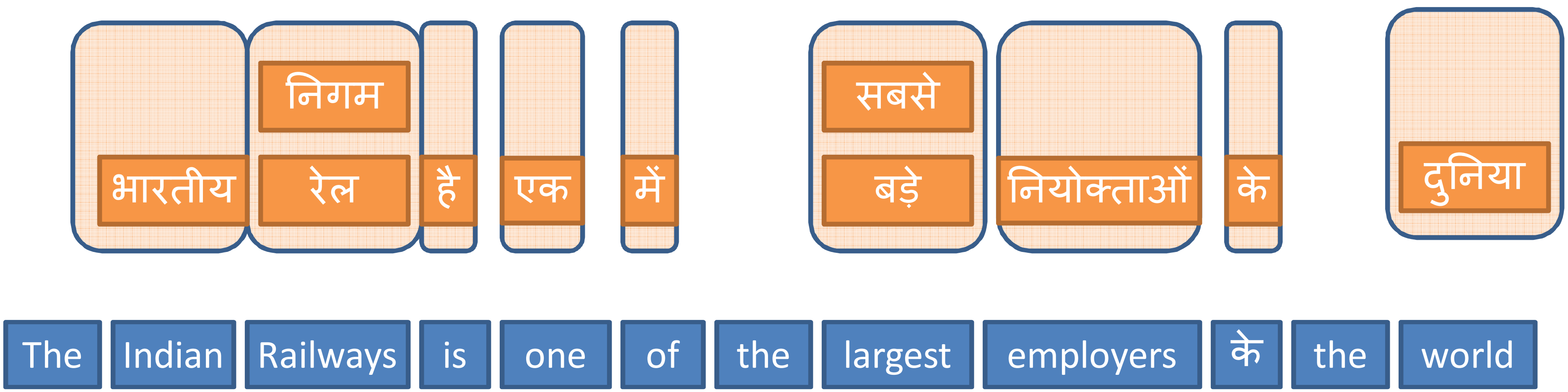
References

1. Peter Brown, Stephen Della Pietra, Vincent Della Pietra, Robert Mercer.*The Mathematics of Statistical Machine Translation: Parameter Estimation*. Computational Linguistics. 1993.
2. Kevin Knight. A Statistical *MT Tutorial Workbook*. 1999.
3. Philip Koehnn. Statistical Machine Translation. 2008.

Generative Process 2

- For each word e_i in sentence e
  - Select the number of words to generate
  - Select the words to generate
  - Permute the words
- Choose the number of words in f for which there are no alignments in e.
  - Choose the words
  - Insert them into proper locations

Generative Process 2



This process serves as the basis for IBM Models 3 to 5

Generative Process 2 (Contd …)

$$
\begin{aligned}
\Pr(\tau, \pi|\mathbf{e}) \quad = \quad & \prod_{i=1}^{l} \Pr(\phi_i|\phi_1^{i-1}, \mathbf{e}) \Pr(\phi_0|\phi_1^l, \mathbf{e}) \times \\
& \prod_{i=0}^{l} \prod_{k=1}^{\phi_i} \Pr(\tau_{ik}|\tau_{i1}^{k-1}, \tau_0^{i-1}, \phi_0^l, \mathbf{e}) \times \\
& \prod_{i=1}^{l} \prod_{k=1}^{\phi_i} \Pr(\pi_{ik}|\pi_{i1}^{k-1}, \pi_1^{i-1}, \tau_0^l, \phi_0^l, \mathbf{e}) \times \\
& \prod_{k=1}^{\phi_0} \Pr(\pi_{0k}|\pi_{01}^{k-1}, \pi_1^l, \tau_0^l, \phi_0^l, \mathbf{e}).
\end{aligned}
$$