# Statistical Machine Translation

Le Anh Cuong

# Reading

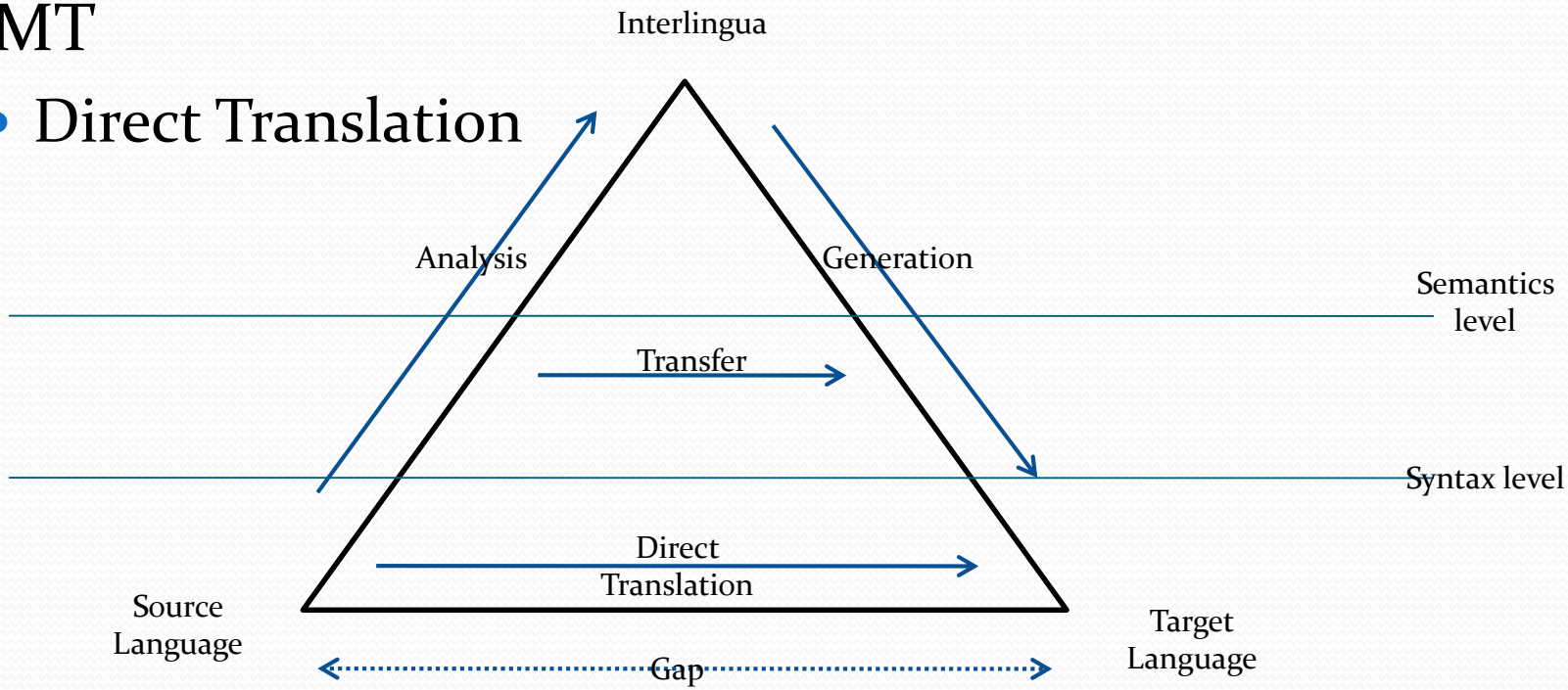
- Chapter 25 [1]
- Chapter 13 [2]

# Contents

- **What Samsung requires to survey**
  - Rule-base and Statistical approach
  - Alignment Model
  - Decoding Algorithms
  - Open Sources
  - Evaluation Methods
  - Using Syntax Info. in SMT

# Machine Translation Pyramid

- RBMT (Rule-Based machine Translation)
  - Analysis, Structure transfer, and Generation
- SMT
  - Direct Translation

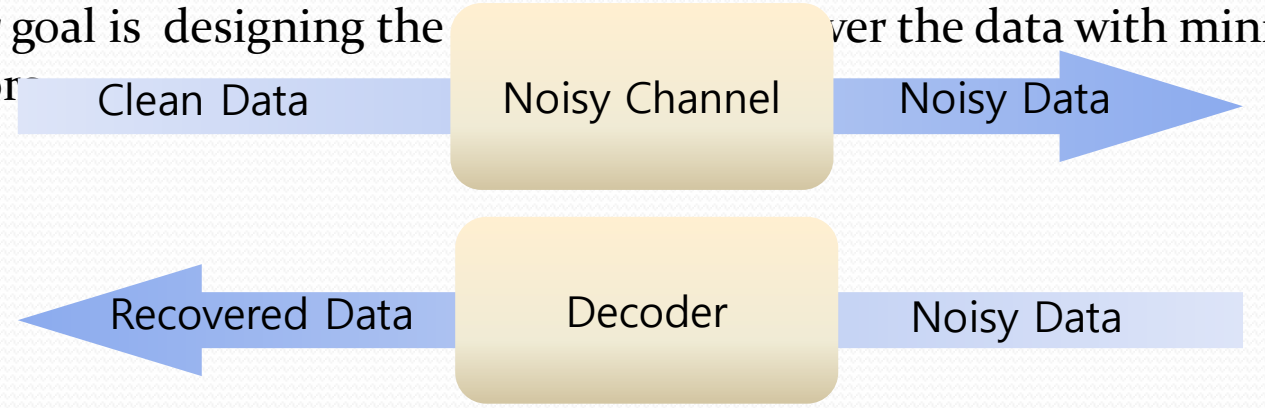


# Comparison

| | RBMT | SMT |
|---|---|---|
| Approach | Analytic | Empirical |
| Based on | Transfer rules | Statistical Evidence |
| Analysis level | Various<br>(morpheme ~ interlingua) | Generally, almost not |
| Translation Speed | Fast | (Relatively) Slow |
| Required Knowledge | Linguistic knowledge<br>Dictionary<br>(Ontology)<br>(Conceptual and cultural differences) | Parallel Texts<br>(morphology for spacing) |
| Adaptability | Low | High |

# SMT: Noisy Channel Model

- Noisy Channel
  - Encoder
  - We Get the data through noisy channel
  - The clean (original) data can not be observed directly
  - Noisy channel adds some noise to the data
- Decoder
  - Estimates the original data from the noisy data
  - Recovered data may contain some errors
  - Our goal is designing the ver the data with minimum error



# Noisy Channel Model

- We want to recover original pattern for a given noisy pattern
  - Choose most probable A given B
  - A : Original data
  - B : Noisy data
  - A' : Estimation of original data

$$A' = \arg\max_{A} \Pr(A \mid B)$$

$$= \arg\max_{A} \frac{\Pr(B \mid A)\Pr(A)}{\Pr(B)}$$

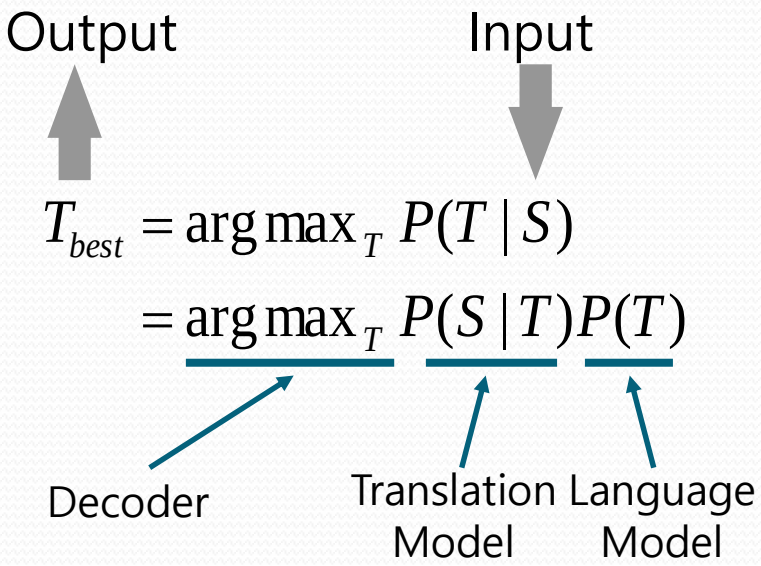
$$= \arg\max_{A} \Pr(B \mid A)\Pr(A)$$

# Noisy Channel Model in SMT

- Given a source sentence S find T that maximizes probability of T given S [Brown et al 1988, 1990]

- Language model:
  - Role : Making fluent sentence
  - Model for target language

- Translation Model:
  - Role : Making correct translation
  - Model for both languages




- Decoder
  - Role : Find a sentence which gives best score
  - We use P(S|T)P(T) rather than P(T|S).

# Noisy Channel model in SMT

- Estimating P(S|T) or P(T|S) at a sentence level is impossible
  - Data sparseness problem
    - → Estimate P(S|T) or P(T|S) at a smaller level. (Typically, words)
  - Assuming independence of translation units, our approximation is

$$P(T \mid S) \cong \prod_i P(T_i \mid S_i)$$

$$P(S \mid T)P(T) \cong \prod_i P(S_i \mid T_i)P(T_i)$$

  - Actually the assumption is not true, we lose many information
  - But, P(T) may model dependency on previous term
    - P(T) may recover some portion of the lost information

$$P(T_i) \cong P(T_i \mid T_{i-1}T_{i-2}\ldots)$$

# Log-linear Model

- Recently, Log-linear model is popular
  - Maximize Sum of logarithms

$$T_{best} = \arg\max_T P(S \mid T) P(T)$$

$$= \arg\max_T [\log P(S \mid T) + \log P(T)]$$

  - Introduce additional features and weights

$$T_{best} = \arg\max_T [w_1 \log P(S \mid T) + w_2 \log P(T) + w_3 \log F(S,T)]$$

  - Generally, we write

$$T_{best} = \arg\max_T \sum_i w_i f_i(S,T)$$

# Parallel Corpus

- Two or more texts written in different languages have same meaning
- We need alignments at least sentence level
- Example

| | |
|---|---|
| 이곳에서 아침식사를 할 수 있습니까? | Can I have breakfast here ? |
| 아침 식사는 얼마 지요? | How much for breakfast ? |
| 취사가 가능 합니까? | Can I cook for myself ? |
| 짐을 여기 두어도 될까요? | Can I leave my baggage here ? |
| 금요일에 체크아웃 하려고 합니다. | I 'm going to leave on Friday . |
| 관광 안내소는 어디 있습니까? | Where can I find tourist information ? |
| 관광 안내를 바랍니다. | Can I get some information , please ? |
| 이곳에서 관광 가이드 한 명을 고용 할 수 있나요? | Can I hire a tour guide here ? |
| 한국말을 하는 가이드를 구할 수 있나요? | Is there a Korean-speaking guide available ? |

# Word Alignment

- IBM-Model 1~5 [Brown et. al 1993]
  - Finding Best alignment
  - Estimating P(S|T)

# N-gram Language Model

- Probability of next words given history
  - We can not store all the word sequences if the length is not limited
  - The words sequence is very scarce if the length is long
    - For the very scarce data, the probability or statistics is meaningless
- Approximation
  - Assume that the probability of a word is independent to too far history
  - Use limited history
    - 0 history – Unigram

$$P(w_n \mid w_1^{n-1}) \approx P(w_n)$$

    - 1 history – Bigram

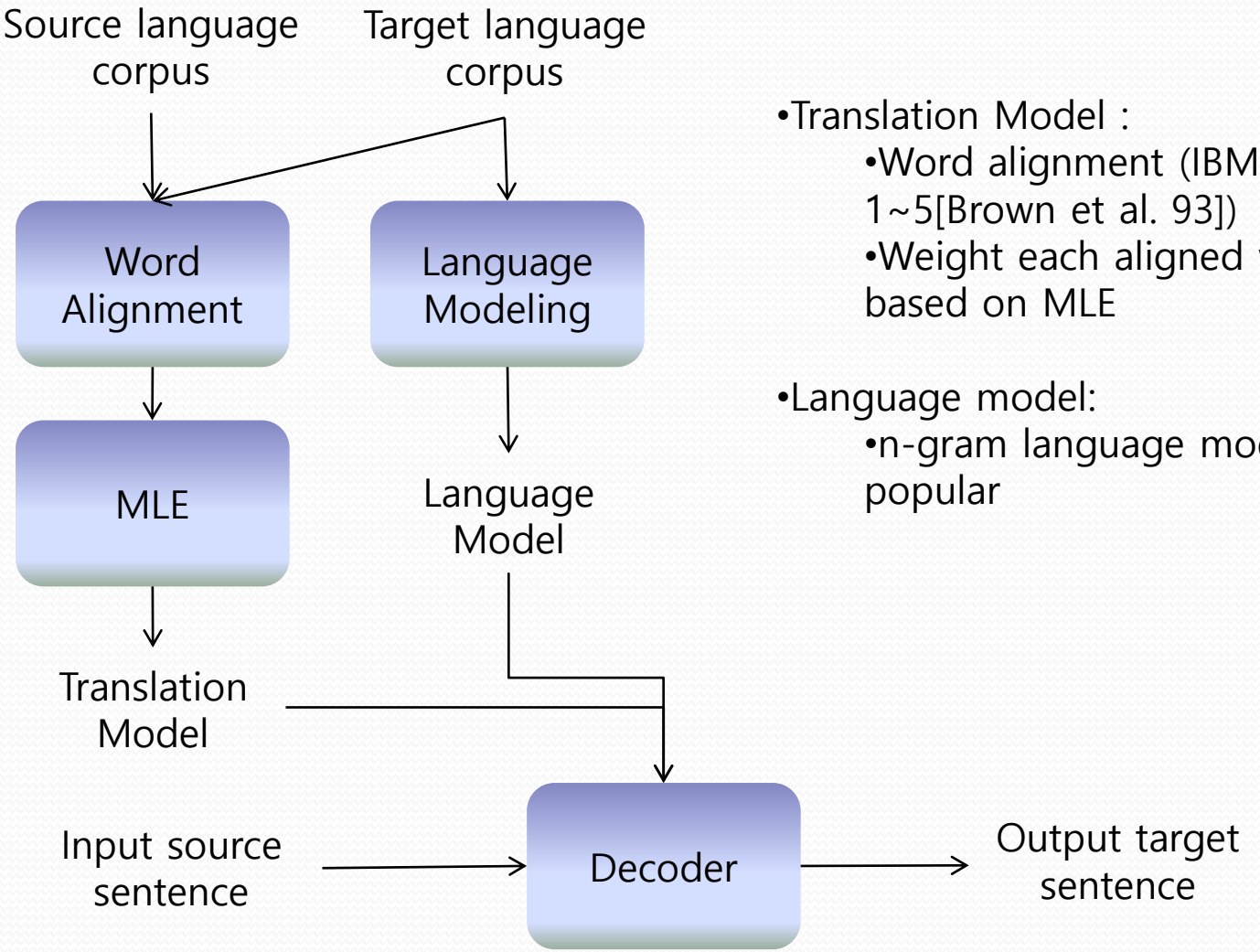
$$P(w_n \mid w_1^{n-1}) \approx P(w_n \mid w_{n-1})$$

    - 2 history – Trigram

$$P(w_n \mid w_1^{n-1}) \approx P(w_n \mid w_{n-2} w_{n-1})$$

    - …
- N- gram is most popular method for scoring sentences

# Data flow




- Translation Model :
  - Word alignment (IBM Model 1~5[Brown et al. 93])
  - Weight each aligned words based on MLE
- Language model:
  - n-gram language model is popular

# Contents

- What Samsung requires to survey
  - Rule-base and Statistical approach
  - Alignment Model
  - Decoding Algorithms
  - Open Sources
  - Evaluation Methods
  - Using Syntax Info. in SMT

# Alignment Model

- GIZA++
  - IBM Translation Model 1~5
  - HMM Alignment Model
- Phrase Level Alignment
- Other Word Alignment Models

# IBM Translation Model Outline

- Goal $\Pr(f \mid e)$
  - Modeling the conditional probability distribution
    - f : French sentence (or source sentence)
    - e: English sentence (or target sentence)

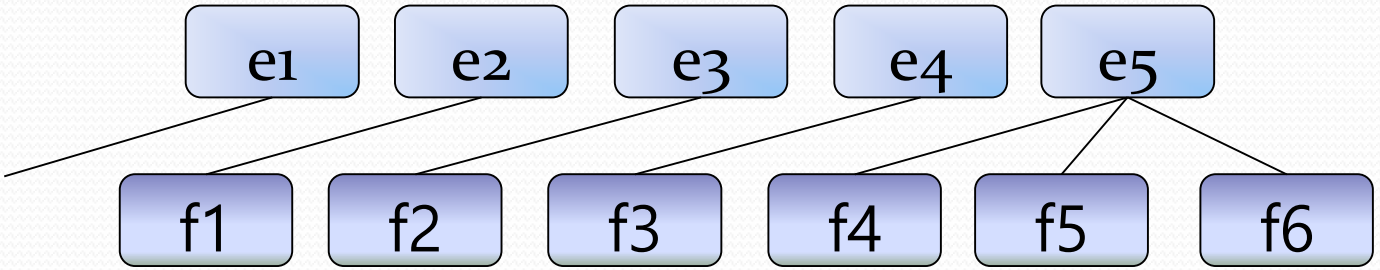
- Models [Brown et al. 1993]
  - A series of five translation models: Model1 ~ Model5
  - Train Model1
  - Train Model2 with the result of Model1 training
  - …
  - Train Model5 with the result of Model4 training

- Algorithm
  - Apply EM algorithm to estimate parameters

# Word Alignment

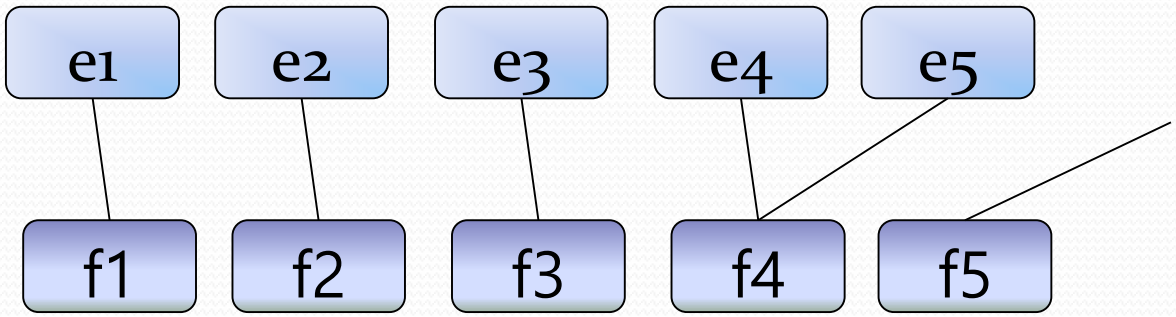
- Type 1
  - An alignment with independent English words
  - n : 1 alignment
  - # of possible alignments
    - with *l* English words and *m* French words
    - for each French words, *l+1* alignments are possible including NULL
  - IBM Models use this restriction

$(l+1)^m$



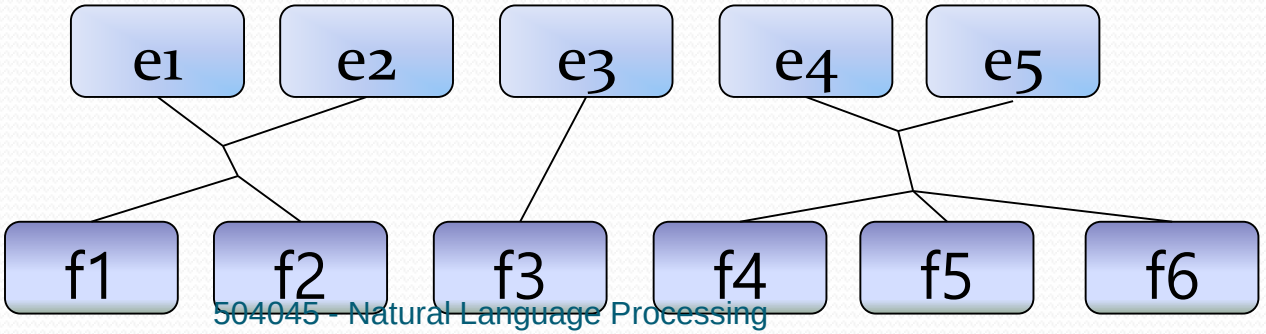
# Word Alignment

- Type 2
  - An alignment with independent French words
  - 1 : n alignment
  - # of possible alignments $(m+1)^l$
    - with $l$ English words and $m$ French words
    - for each English words, $m+1$ alignments are possible including NULL
  - Inverse direction



# Word Alignment

- Type 3
  - An alignment with independent English words
  - n : n' alignment, general alignment
  - # of possible alignments : Very large
    - with *m* English words and *l* French words
    - For the first English word $2^l+1$ alignments are possible
    - For the second English word $2^l+1-c$ alignments are possible where c is number of French words aligned with the first English word
  - Alignment for Phrase based Machine Translation



# Word Alignment: variable

- Our goal is estimating the conditional probability

$$\Pr(\mathbf{f} \mid \mathbf{e})$$

- We can introduce a hidden variable a (that is word alignment)

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

- Assume that each French word has exactly one connection
  - The word alignment a can be represented by a series

$$a_1^m \equiv a_1 a_2 \ldots a_m$$

    - Values are between 0 and l, where l is the length of English sentence
    - if $a_2 = 4$, it means
    - "The French word at position 2 aligned with the English word at position 4"
  - Position 0 is reserved for the null word

# Model 1,2 Likelihood: Exact Equation

- A possible form of exact equation
  - "Exact" means that it is not an approximation

$$\Pr(\mathbf{f},\mathbf{a}\mid\mathbf{e}) = \underbrace{\Pr(m\mid\mathbf{e})}_{1}\underbrace{\prod_{j=1}^{m}}\underbrace{\Pr(a_j \mid a_1^{j-1}, f_1^{j-1}, m, \mathbf{e})}_{2}\underbrace{\Pr(f_j \mid a_1^{j}, f_1^{j-1}, m, \mathbf{e})}_{3}$$

  - Part 1. Choose the length of the French string
    - Given English string
  - Part 2. Choose where to connect
    - Given English string
    - Given length of French string
    - Given history
  - Part 3. Choose identity of the word
    - Given English string
    - Given length of French string
    - Given history
    - Given current word alignment (Part 2)

# Alignment Process: Model 1,2

- An alignment process corresponding to exact equation on previous page
- Choose a length for French string f
- **for** i = 1 to m
- **begin**
  - Decide which position in e is connected to $f_i$
  - Decide which identity of $f_i$ is
- **end**

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

# Model 1

- Exact equation
  - Too complex → We need some approximation

- Approximation $\Pr(m \mid \mathbf{e})$
  - Part 1:
    - Assume that it is independent of m and e
    - → $\epsilon$, constant
      $\Pr(a_j \mid a_1^{j-1}, f_1^{j-1}, m, \mathbf{e})$
  - Part 2:
    - Depends only on the length of the English string
    - → $(l+1)^{-1}$, uniform
      $\Pr(f_j \mid a_1^{j}, f_1^{j-1}, m, \mathbf{e})$
  - Part3:
    - Depends only on the French word and corresponding English word: $f_j, e_{a_j}$
    - → $t(f_j \mid e_{a_j})$, translation probability

# Model 1

- Likelihood function

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

- The Simplest model
- The order of the words in e and f does not affect the likelihood
- Model 1 likelihood function has only one maximum
  - Model 1 always finds global maximum

# Model 2

- Approximation $\Pr(m \mid \mathbf{e})$
  - Part 1:
    - Assume that it is independent of *m* and *e*
    - → Є, constant
    - same to Model 1
  - Part 2: $\Pr(a_j \mid a_1^{j-1}, f_1^{j-1}, m, \mathbf{e})$
    - Depends on
      - Positions of French word and corresponding English word $(j, a_j)$
      - Length of French string and English string $(m, l)$
    - We introduce alignment probabilities

$$a(a_j \mid j, m, l) = \Pr(a_j \mid a_1^{j-1}, f_1^{j-1}, m, l)$$

  - Part3: $\Pr(f_j \mid a_1^{j}, f_1^{j-1}, m, \mathbf{e})$
    - Depends only on the French word and corresponding English word $f_j, e_{a_j}$
    - → $t(f_j \mid e_{a_j})$, translation probability
    - same to Model 1

# Model 2

- Likelihood function

$$\Pr(\mathbf{f} \mid \mathbf{e}) = \varepsilon \sum_{a_1=0}^{l} \ldots \sum_{a_m=0}^{l} \prod_{j=1}^{m} \mathrm{t}(f_j \mid e_{a_j}) a(a_j \mid j, m, l)$$

- Alignment probability is introduced compared to Model1
- Model 1 is a special case of Model 2

# Fertility

- Definition : Fertility of *e*
  - A random variable $\Phi_e$ that corresponds the number of French words to which e is connected in a randomly selected alignment
- Modeling the Fertility
  - Model 1 and 2 : not clear
  - Model 3, 4 and 5 : Parameterize fertilities directly
- Tablet
  - A list of French words to connect to each English word
- Tableau
  - The collection of tablets, A random variable
  - $T_i$ : the tablet for $i^{th}$ English word
  - $T_{ik}$ : $k^{th}$ French word in the $i^{th}$ tablet

# Model 3, 4 and 5 Likelihood: Exact Equation

- The Joint likelihood for a tableau, $\tau$, and a permutation, $\pi$

$$\Pr(\tau,\pi\mid\mathbf{e}) = \underbrace{\prod_{i=1}^{l}\Pr(\phi_i\mid\phi_1^{i-1},\mathbf{e})}_{1}\underbrace{\Pr(\phi_0\mid\phi_1^{l},\mathbf{e})}_{5}\times\underbrace{\prod_{i=0}^{l}\prod_{k=1}^{\phi_i}\Pr(\tau_{ik}\mid\tau_{i1}^{k-1},\tau_0^{i-1},\phi_0^{l},\mathbf{e})}_{2}\times$$
$$\underbrace{\prod_{i=1}^{l}\prod_{k=1}^{\phi_i}\Pr(\pi_{ik}\mid\pi_{i1}^{k-1},\pi_1^{i-1},\tau_0^{l},\phi_0^{l},\mathbf{e})}_{3}\times\underbrace{\prod_{k=1}^{\phi_0}\Pr(\pi_{0k}\mid\pi_{01}^{k-1},\pi_1^{l},\tau_0^{l},\phi_0^{l},\mathbf{e})}_{4}$$

- Knowing $\tau$ and $\pi$ determine a French string and an alignment
  - Different $\tau$ and $\pi$ may lead to same pair ***f***, ***a***
  - $\langle \boldsymbol{f}, \boldsymbol{a}\rangle$: pairs of $\tau$ and $\pi$ that lead pair ***f*** and ***a***

- From the above, we have $\Pr(\mathbf{f},\mathbf{a}\mid\mathbf{e}) = \sum_{(\tau,\pi)\in\langle\mathbf{f},\mathbf{a}\rangle}\Pr(\tau,\pi\mid\mathbf{e})$

# Alignment Process: Model 3, 4, 5

- **for each** English word
- **begin**
  - Decide the fertility of the word
  - Get a list of French words to connect to the word
- **end**
- Permute words in tableau to generate *f*

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

# Model-3

- Approximation $\Pr(\phi_i \mid \phi_1^{i-1}, \mathbf{e})$
  - Part 1 :
    - Depends only on $\phi_i$ and $e_i$
    - Fertility probability
    - → $n(\phi_i \mid e_i)$
  - Part 2 : $\Pr(\tau_{ik} \mid \tau_{i1}^{k-1}, \tau_0^{i-1}, \phi_0^l, \mathbf{e})$
    - Depends only on $\tau_{ik}$ and $e_i$
    - Translation probability
    - → $t(f \mid e_i)$
  - Part 3 : $\Pr(\pi_{ik} \mid \pi_{i1}^{k-1}, \pi_1^{i-1}, \tau_0^l, \phi_0^l, \mathbf{e})$
    - Depends only on $\pi_i$, $i$, m, $l$
    - Distortion probability
    - → $d(j \mid i, m, l)$

# Model-3

- Approximation (cont') $\Pr(\pi_{0k} \mid \pi_{01}^{k-1}, \pi_1^l, \tau_0^l, \phi_0^l, \mathbf{e})$
  - Part 4 :
    - → $\frac{1}{\phi_0 - k}$
    - → $\prod_{k=1}^{\phi_0} \Pr(\pi_{0k} \mid \pi_{01}^{k-1}, \pi_1^l, \tau_0^l, \phi_0^l, \mathbf{e}) = \frac{1}{\phi_0!}$
    - → $\Pr(\phi_0 \mid \phi_1^l, \mathbf{e})$
  - Part 5 :
    - → $\binom{\phi_1 + \ldots + \phi_l}{\phi_0} p_0^{\phi_1 + \ldots + \phi_l - \phi_0} p_1^{\phi_0} = \binom{m - \phi_0}{\phi_0} p_0^{m - 2\phi_0} p_1^{\phi_0}$

From the above approximations …

$$\Pr(\mathbf{f} \mid \mathbf{e}) = \sum_{a_1=0}^{l} \ldots \sum_{a_m=0}^{l} \binom{m - \phi_0}{\phi_0} p_0^{m-2\phi_0} p_1^{\phi_0} \prod_{i=1}^{l} \phi_i! \mathrm{n}(\phi_i \mid e_i) \times \prod_{j=1}^{m} \mathrm{t}(f_j \mid e_{a_j}) \mathrm{d}(j \mid a_j, m, l)$$

$$\Pr(f \mid e) = \sum_{a} \Pr(f,a \mid e) = \sum_{a} \sum_{(\tau,\pi)\in<f,a>} \Pr(\tau,\pi \mid e)$$

$$\Pr(\tau,\pi \mid \mathbf{e}) = \prod_{i=1}^{l} \Pr(\phi_i \mid \phi_1^{i-1},\mathbf{e}) \Pr(\phi_0 \mid \phi_1^{l},\mathbf{e}) \times \prod_{i=0}^{l}\prod_{k=1}^{\phi_i} \Pr(\tau_{ik} \mid \tau_{i1}^{k-1},\tau_0^{i-1},\phi_0^{l},\mathbf{e}) \times$$

$$\prod_{i=1}^{l}\prod_{k=1}^{\phi_i} \Pr(\pi_{ik} \mid \pi_{i1}^{k-1},\pi_1^{i-1},\tau_0^{l},\phi_0^{l},\mathbf{e}) \times \prod_{k=1}^{\phi_0} \Pr(\pi_{0k} \mid \pi_{01}^{k-1},\pi_1^{l},\tau_0^{l},\phi_0^{l},\mathbf{e})$$

$$\Pr(f \mid e) = \sum_{a} \sum_{(\tau,\pi)\in<f,a>} \prod_{i=1}^{l} \Pr(\phi_i \mid \phi_1^{i-1},\mathbf{e}) \Pr(\phi_0 \mid \phi_1^{l},\mathbf{e}) \times \prod_{i=0}^{l}\prod_{k=1}^{\phi_i} \Pr(\tau_{ik} \mid \tau_{i1}^{k-1},\tau_0^{i-1},\phi_0^{l},\mathbf{e}) \times$$

$$\prod_{i=1}^{l}\prod_{k=1}^{\phi_i} \Pr(\pi_{ik} \mid \pi_{i1}^{k-1},\pi_1^{i-1},\tau_0^{l},\phi_0^{l},\mathbf{e}) \times \prod_{k=1}^{\phi_0} \Pr(\pi_{0k} \mid \pi_{01}^{k-1},\pi_1^{l},\tau_0^{l},\phi_0^{l},\mathbf{e})$$

a를 $a_1$, $a_2$, ... 의 조합으로 봄

$$\Pr(f \mid e) = \sum_{a_1} \cdots \sum_{a_m} \sum_{(\tau,\pi)\in<f,a>} \prod_{i=1}^{l} \Pr(\phi_i \mid \phi_1^{i-1},\mathbf{e}) \Pr(\phi_0 \mid \phi_1^{l},\mathbf{e}) \times \prod_{i=0}^{l}\prod_{k=1}^{\phi_i} \Pr(\tau_{ik} \mid \tau_{i1}^{k-1},\tau_0^{i-1},\phi_0^{l},\mathbf{e}) \times$$

$$\prod_{i=1}^{l}\prod_{k=1}^{\phi_i} \Pr(\pi_{ik} \mid \pi_{i1}^{k-1},\pi_1^{i-1},\tau_0^{l},\phi_0^{l},\mathbf{e}) \times \prod_{k=1}^{\phi_0} \Pr(\pi_{0k} \mid \pi_{01}^{k-1},\pi_1^{l},\tau_0^{l},\phi_0^{l},\mathbf{e})$$

Model 3 approximation

$$\Pr(f \mid e) = \sum_{a_1} \cdots \sum_{a_m} \sum_{(\tau,\pi)\in<f,a>} \prod_{i=1}^{l} n(\phi_i \mid \mathbf{e}_i) \binom{m-\phi_0}{\phi_0} p_0^{m-2\phi_0} p_1^{\phi_0} \times \prod_{j=1}^{m} t(f_j, e_{a_j}) \times$$

$$\phi_0! \prod_{j=1}^{m} d(j \mid a_j, m, l) \times \frac{1}{\phi_0!}$$

Distortion에서 도출 된 성분과 Part 4가 서로 상쇄됨(약분)

$$\Pr(\mathrm{f}\mid \mathrm{e}) = \sum_{\mathrm{a}_1}\cdots\sum_{\mathrm{a}_m}\sum_{(\tau,\pi)\in<\mathrm{f},\mathrm{a}>}\prod_{i=1}^{l} n(\phi_i\mid \mathbf{e}_i)\binom{m-\phi_0}{\phi_0} p_0^{m-2\phi_0} p_1^{\phi_0}\times\prod_{j=1}^{m}\mathrm{t}(f_j, e_{a_j})\times$$

$$\prod_{j=1}^{m}\mathrm{d}(j\mid a_j, m, l)$$

In Model 3 approximation,

If $(\tau_1,\pi_1)$ and $(\tau_2,\pi_2)$ represent same $<\mathrm{f},\mathrm{a}>$,

then $\Pr(\tau_1,\pi_1\mid \mathrm{e}) = \Pr(\tau_2,\pi_2\mid \mathrm{e})$

The number of elements in $<\mathrm{f},\mathrm{a}>$ is $\prod_{i=0}^{l}\phi_i!$

$$\Pr(\mathrm{f}\mid \mathrm{e}) = \sum_{\mathrm{a}_1}\cdots\sum_{\mathrm{a}_m}\prod_{\mathrm{i}=1}^{l}\phi_i!\prod_{i=1}^{l} n(\phi_i\mid \mathbf{e}_i)\binom{m-\phi_0}{\phi_0} p_0^{m-2\phi_0} p_1^{\phi_0}\times\prod_{j=1}^{m}\mathrm{t}(f_j, e_{a_j})\times$$

$$\prod_{j=1}^{m}\mathrm{d}(j\mid a_j, m, l)$$

식을 정리하면 …

$$\Pr(\mathbf{f}\mid \mathbf{e}) = \sum_{a_1=0}^{l}\cdots\sum_{a_m=0}^{l}\binom{m-\phi_0}{\phi_0} p_0^{m-2\phi_0} p_1^{\phi_0}\prod_{i=1}^{l}\phi_i!\,\mathrm{n}(\phi_i\mid e_i)\times\prod_{j=1}^{m}\mathrm{t}(f_j\mid e_{a_j})\,\mathrm{d}(j\mid a_j, m, l)$$

# Model 4

- A Problem of Model 3
  - Phrases should be considered as a unit
  - Model 3 does not account well for phrases
    - Every words are moved independently
  - Model 4 modifies Model 3 to consider phrases
    - Modeling the phrase property
    - Modifying distortion model

# Model 4

- Solution
  - Replace the distortion model by two sets of parameters
    - A parameter for head of each cept

$$\Pr(\Pi_{[i]1} = j \mid \pi_1^{[i]-1}, \tau_0^l, \phi_0^l, \mathbf{e}) = \mathrm{d}_1(j - \otimes_{i-1} \mid A(e_{[i-1]}), B(f_j))$$

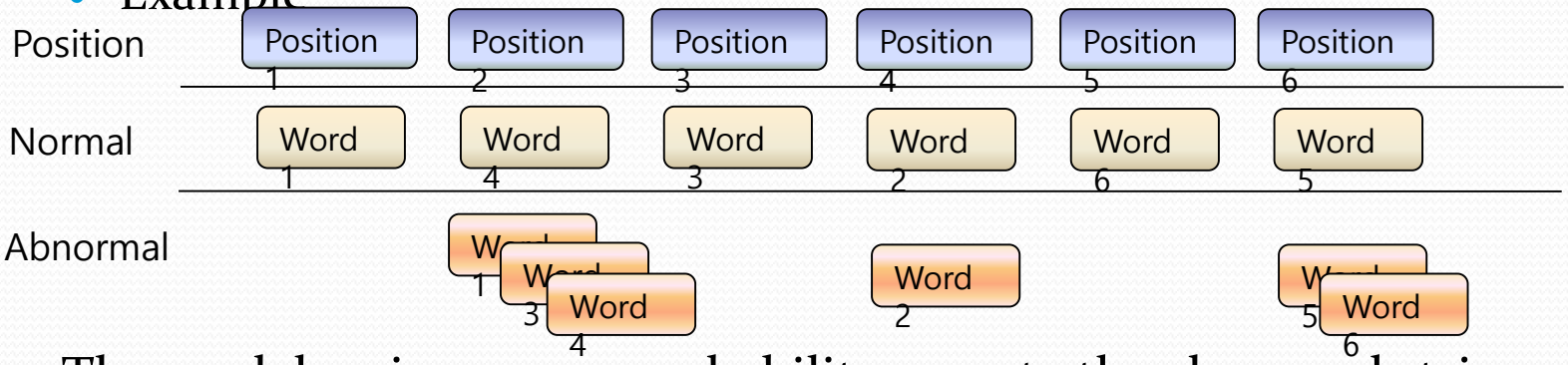
    - A parameter for remaining part of each cept

$$\Pr(\Pi_{[i]k} = j \mid \pi_{[i]1}^{k-1}, \pi_1^{[i]-1}, \tau_0^l, \phi_0^l, \mathbf{e}) = \mathrm{d}_{>1}(j - \pi_{[i]k-1} \mid B(f_j))$$

    - Terms
      - [i] : The position in the English string of the i[th] one-word cept.
      - : Ceiling of the average value of the positions in the French string of the words from i[th] tablet
      - A(.),B(.) : function that changes words into some class of vocabulary
    - cept :
      - Roughly, a unit of concepts , it can be a translation unit
      - A set of English words connected to a French word in a particular alignment

# Deficiency

- A problem with distortion probability (for Model 3, 4)
  - We assumed $\Pr(\pi_{ik} \mid \pi_{i1}^{k-1}, \pi_1^{i-1}, \tau_0^l, \phi_0^l, \mathbf{e})$ depends only on j, i, m, and l
  - The distortion probabilities for assigning positions to later words do not depend on the positions assigned to earlier words
    - Multiple words can have same position
    - Empty position also possible
    - Example




  - The model assigns some probability mass to the abnormal string
- For this problem we say that it is "deficient"

# Model-5

- Model 3 and Model 4 are deficient
- Model 5 remove the deficiency
- Restriction to assigning position
  - We need to avoid unavailable positions when we assign the positions
    - Model 3 and 4 do not consider this point
  - First, we define $v_j$
    - the number of available position up to j including $j$

$$v_j = v(j, \tau_1^{[i]-1}, \tau_{[i]1}^{k-1})$$

0, if $j$ is not available
1, if $j$ is available

  - Rewrite the distortion probabilities of model 4
    - For head $\mathrm{d}_1(v_j \mid B(f_j), v_{\oplus i=1}, v_m - \phi_{[i]} + 1)(1 - \delta(v_j, v_{j-1}))$
    - For remaining parts

$$\mathrm{d}_{>1}(v_j - v_{\pi[i]k-1} \mid B(f_j), v_m - v_{\pi[i]k-1} - \phi_{[i]} + k)(1 - \delta(v_j, v_{j-1}))$$

# Summary of IBM Model 1~5

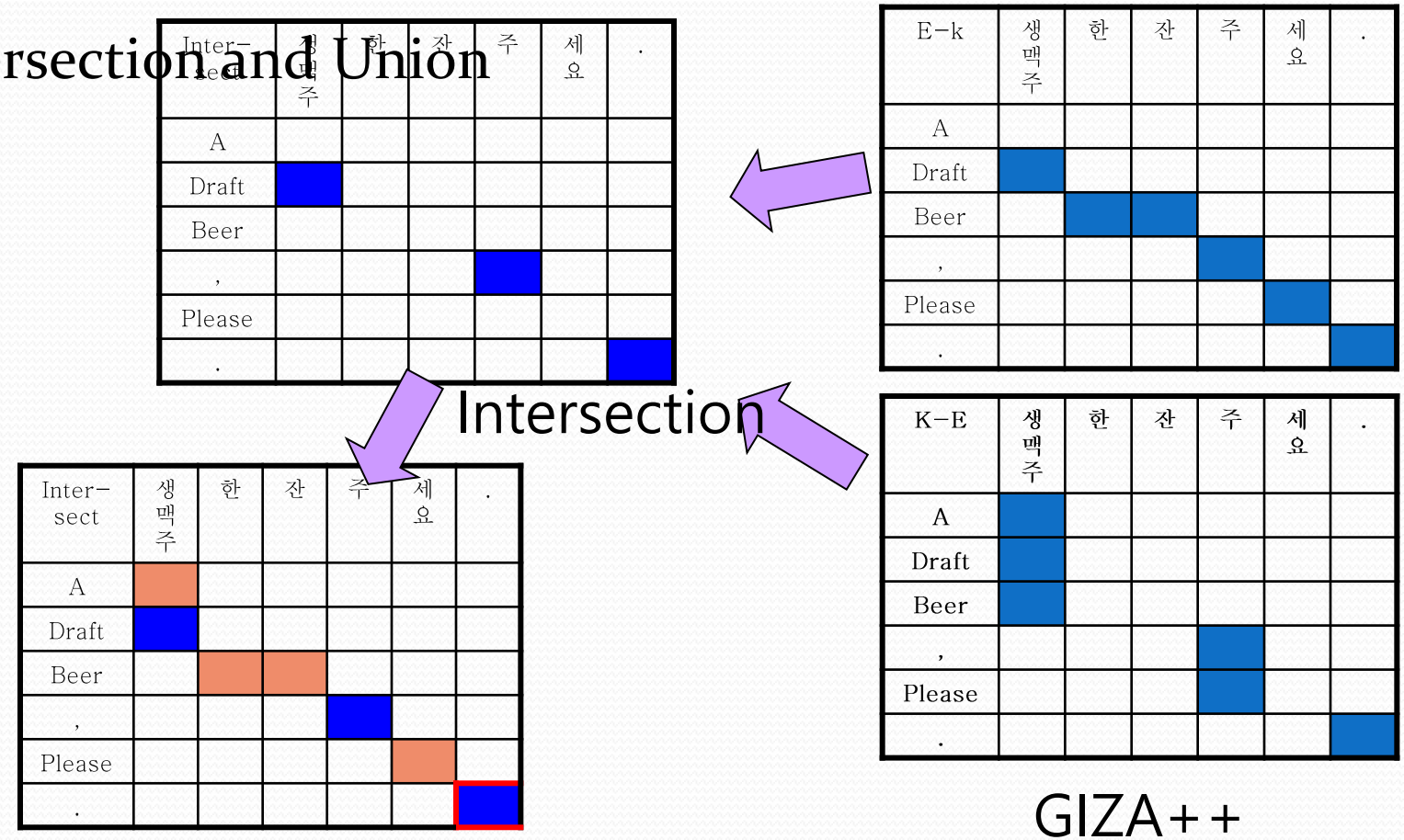
| | Probability models | | | Other features |
|---|---|---|---|---|
| | Translation | Alignment / Distortion | Length/ Fertility | |
| Model 1 | $t(f_j \mid e_{a_j})$ | $(l+1)^{-1}$ | Constant | Unique maxima |
| Model 2 | " | $a(a_j \mid j,m,l)$ | Constant | - |
| Model 3 | " | $d(j \mid i,m,l)$<br>$(\phi_0 - k)^{-1}$ | $n(\phi \mid e_i)$<br>$_{m-\phi_0}C_{\phi_0} p_0^{m-2\phi_0} p_1^{\phi_0}$ | Introduced fertility |
| Model 4 | " | $d_{>1}(j - \pi_{[i]k-1} \mid B(f_j))$<br>$d_1(j - \odot_{i-1} \mid A(e_{[i-1]}), B(f_j))$<br>$(\phi_0 - k)^{-1}$ | " | Phrase property |
| Model 5 | " | $d_1(v_j \mid B(f_j), v_{\oplus i=1}, v_m - \phi_{[i]} + 1)(1-\delta(v_j, v_{j-1}))$<br>$d_{>1}(v_j - v_{\pi[i]k-1} \mid B(f_j), v_m - v_{\pi[i]k-1} - \phi_{[i]} + k)(1-\delta(v_j, v_{j-1}))$<br>$(\phi_0 - k)^{-1}$ | " | Removed deficiency |

# Phrase Extraction

- Phrase level alignment
- Pharaoh's process
  - Get word alignments in both directions
    - From the GIZA++, IBM Model 4
    - Bi-directional word alignment (source to target, target to source)
  - Intersect the word alignments
  - Expand the intersection to the union
    - Use some heuristic to resolve conflict
    - Pharaoh presents 6 heuristics
  - Extract all possible phrase pairs which consistent with word alignment
  - Assign probabilities to the phrase pairs
    - Count the phrase co-occurrences
    - Divide it by count of occurrence of phrase e

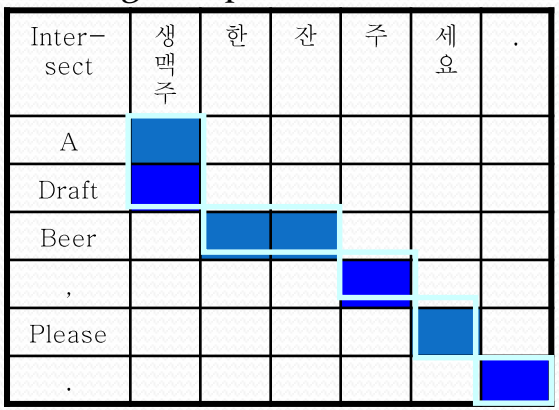
# Bidirectional alignment

- Intersection and Union



# Phrase Extraction

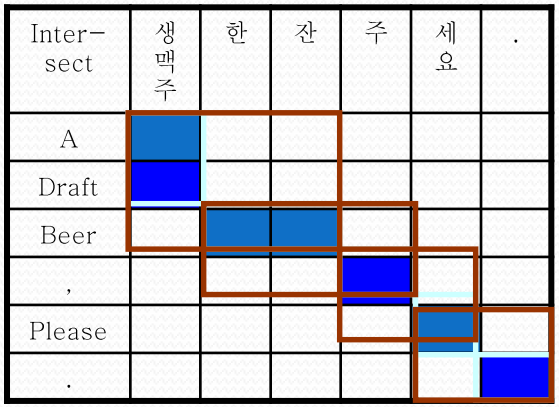
- Learning all phrase pairs that are consistent with the word alignment
  - One should limit the maximum length of phrases




- (A Draft | 생맥주) ( Beer | 한 잔 ) (, | 주) (Please | 세요) (. | .)

# Phrase Extraction

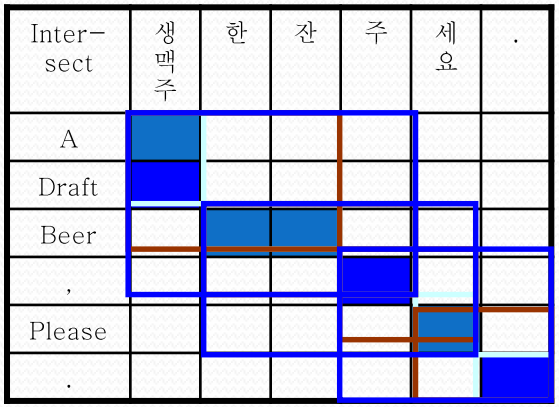
- Learning all phrase pairs that are consistent with the word alignment




- (A Draft | 생맥주) ( Beer | 한 잔 ) (, | 주) (Please | 세요) (. | .)
- (A Draft Beer | 생맥주 한 잔) (Beer , | 한 잔 주) (, Please | 주 세요) (Please . | 세요 .)

# Phrase Extraction

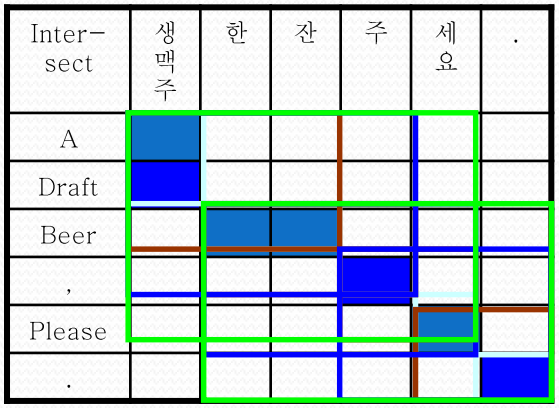
- Learning all phrase pairs that are consistent with the word alignment




- (A Draft | 생맥주) ( Beer | 한 잔 ) (, | 주) (Please | 세요) (. | .)
- (A Draft Beer | 생맥주 한 잔) (Beer , | 한 잔 주) (, Please | 주 세요) (Please . | 세요 .)
- (A Draft Beer , | 생맥주 한 잔 주 ) ( Beer , Please | 한 잔 주 세요 ) ( , Please | 주 세요 .)

# Phrase Extraction

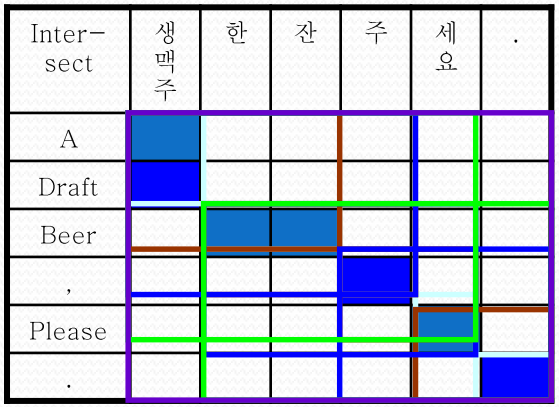
- Learning all phrase pairs that are consistent with the word alignment




- (A Draft | 생맥주) ( Beer | 한 잔 ) (, | 주) (Please | 세요) (. | .)
- (A Draft Beer | 생맥주 한 잔) (Beer , | 한 잔 주) (, Please | 주 세요) (Please . | 세요 .)
- (A Draft Beer , | 생맥주 한 잔 주 ) ( Beer , Please | 한 잔 주 세요 ) ( , Please | 주 세요 .)
- (A Draft Beer , Please | 생맥주 한 잔 주 세요 ) ( Beer , Please . | 한 잔 주 세요 .)

# Phrase Extraction

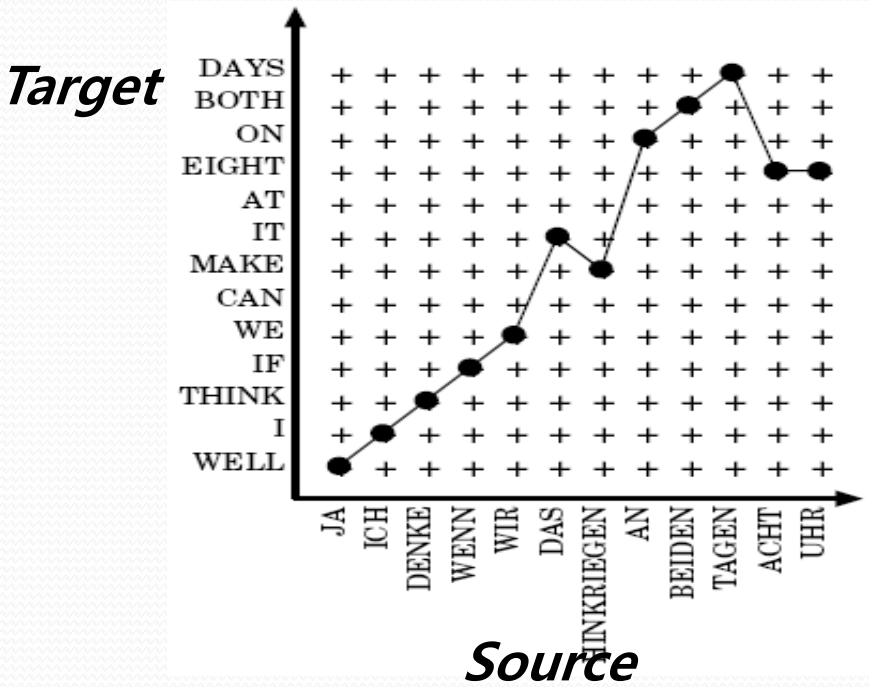
- Learning all phrase pairs that are consistent with the word alignment




- (A Draft | 생맥주) ( Beer | 한 잔 ) (, | 주) (Please | 세요) (. | .)
- (A Draft Beer | 생맥주 한 잔) (Beer , | 한 잔 주) (, Please | 주 세요) (Please . | 세요 .)
- (A Draft Beer , | 생맥주 한 잔 주 ) ( Beer , Please | 한 잔 주 세요 ) ( , Please | 주 세요 .)
- (A Draft Beer , Please | 생맥주 한 잔 주 세요 ) ( Beer , Please . | 한 잔 주 세요 .)
- (A Draft Beer , Please . | 생맥주 한 잔 주 세요 . )

# HMM Alignment Model

- Goal : Improve IBM Model 1-2
- Idea : relative position model



# HMM Alignment Model

- The alignment model $\Pr(f_1^J, a_1^J \mid e_1^I)$ can be structured without loss of generality as follows: (IBM Model 1,2)

$$\Pr(f_1^J, a_1^J \mid e_1^I) = \Pr(J \mid e_1^I) \prod_{j=1}^{J} \Pr(f_j, a_j \mid f_1^{j-1}, a_1^{j-1}, e_1^I)$$

$$= \Pr(J \mid e_1^I) \prod_{j=1}^{J} \Pr(a_j \mid f_1^{j-1}, a_1^{j-1}, e_1^I) \Pr(f_j \mid f_1^{j-1}, a_1^{j-1}, e_1^I)$$

- In HMM alignment model
  - A first-order dependence for the alignments

$$\Pr(a_j \mid f_1^{j-1}, a_1^{j-1}, e_1^I) = \Pr(a_j \mid a_{j-1}, I)$$

  - The lexicon probability depends only on the word at position $a_j$

$$\Pr(f_j \mid f_1^{j-1}, a_1^{j-1}, e_1^I) = \Pr(f_j \mid e_{a_j})$$

# HMM Alignment Model

- Alignment probability:
  - A simple length model $p(J \mid e_1^I) = p(J \mid I)$

$$\Pr(f_1^J \mid e_1^I) = p(J \mid I) \sum_{a_1^J} \prod_{j=1}^{J} p(a_j \mid a_{j-1}, I)\, p(f_j \mid e_{a_j})$$

  - Alignment depends on relative position

$$p(i \mid i', I) = \frac{c(i - i')}{\sum_{i''=1}^{I} c(i'' - i')}$$

- Maximum approximation:

$$\Pr(f_1^J \mid e_1^I) \cong p(J \mid I) \max_{a_1^J} \prod_{j=1}^{J} p(a_j \mid a_{j-1}, I)\, p(f_j \mid e_{a_j})$$

# Other Alignment Method

- Heuristic Method
  - Dictionary Look up
  - Transliteration and string similarity
  - Nearest aligned neighbor (alignment locality)
  - POS affinities
  - …
- Hybrid Method
  - Combing two or more methods
  - Intersection, Union, Voting, …
- Variants of IBM Models and HMM Model

# Contents

- What Samsung requires to survey
  - Rule-base and Statistical approach
  - Alignment Model
  - Decoding Algorithms
  - Open Sources
  - Evaluation Methods
  - Using Syntax Info. in SMT

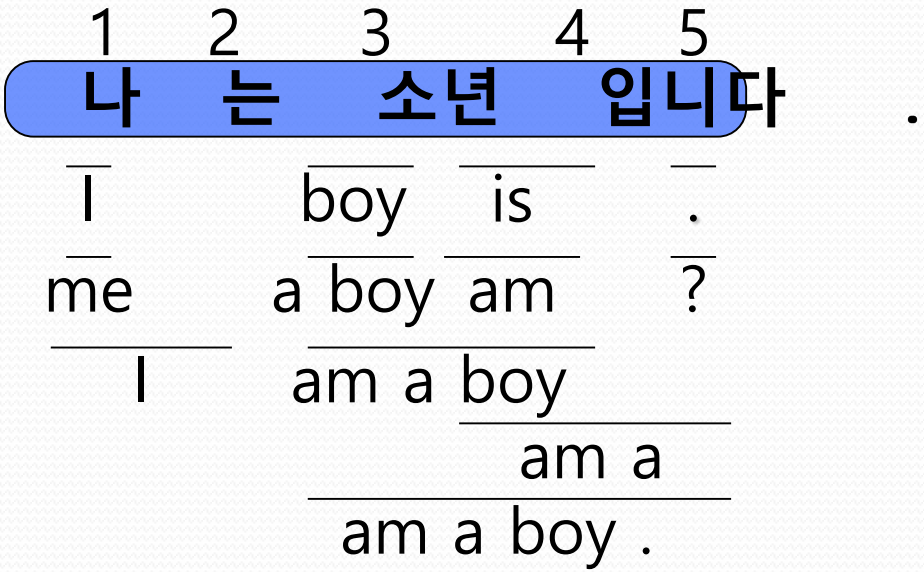
# Decoding Algorithms

- Beam Search Style
  - Phrase-based Systems
  - Pharaoh, Moses and its variants

- CFG Parsing style
  - Syntax-based Systems, SMT by parsing
  - Hiero, GenPar, …

# Pharaoh Decoding

- Translation options
  - In a sentence of length n, there are $\sum_{i=1}^{n} i$ phrases
  - A translation option is a possible translation of a phrase

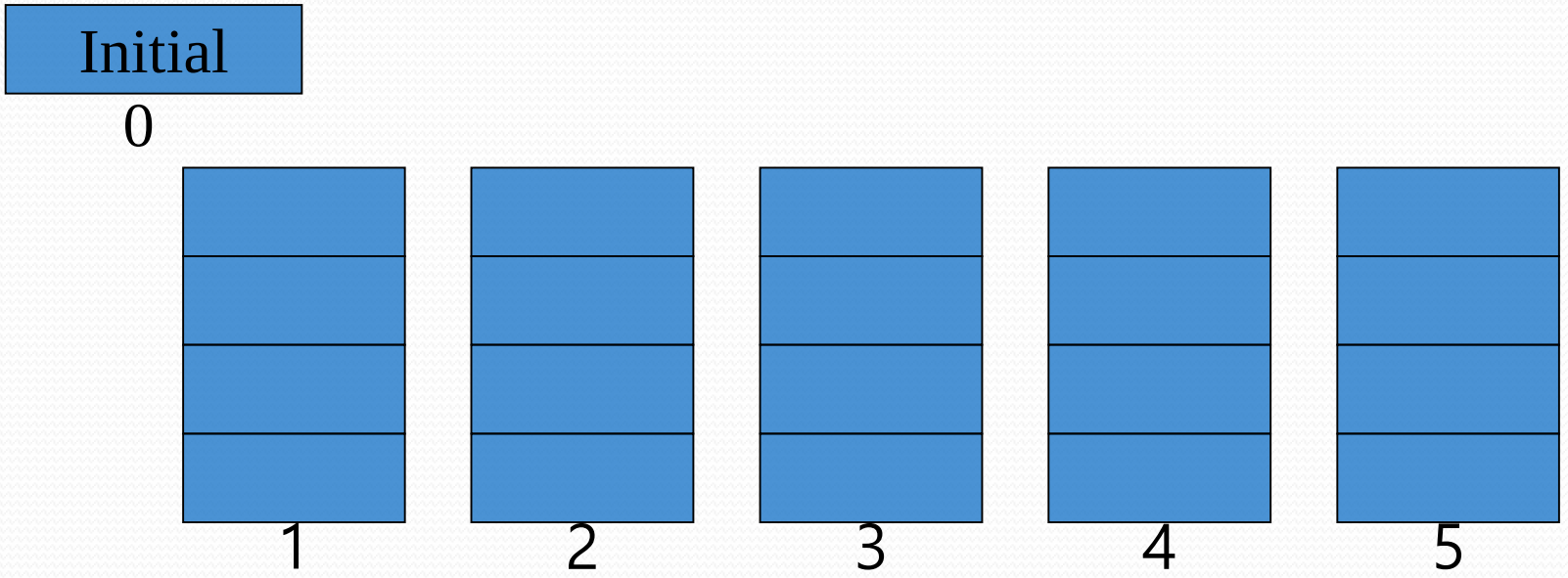


# Pharaoh Decoding

- Hypothesis
  - Hypothesis is a Partial translation taken by applying some translation options
  - Contents (data structure)
    - A back link to the previous state
    - The foreign words covered so far
    - The last two native words generated
    - The end of the last foreign phrase covered
    - The last added native phrase
    - The cost so far
    - An estimate of the future cost

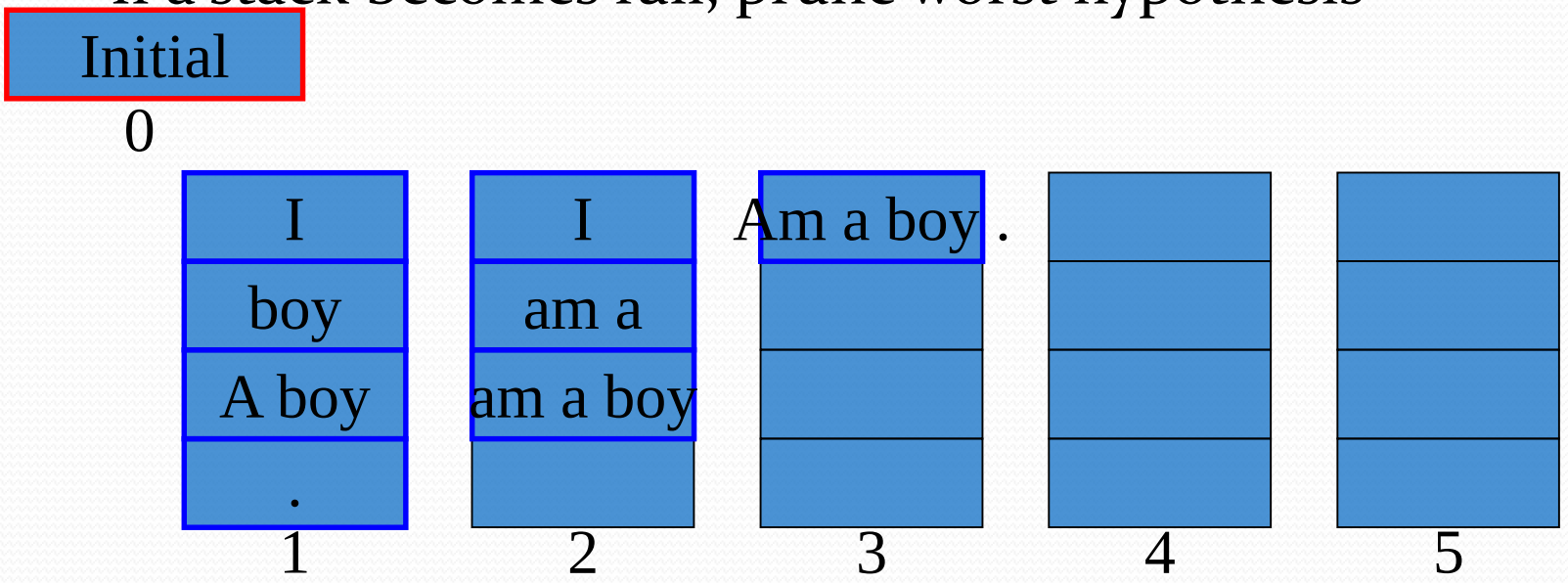
# Pharaoh Decoding

- Decoding
  - Initialize hypothesis stack
  - Create initial hypothesis



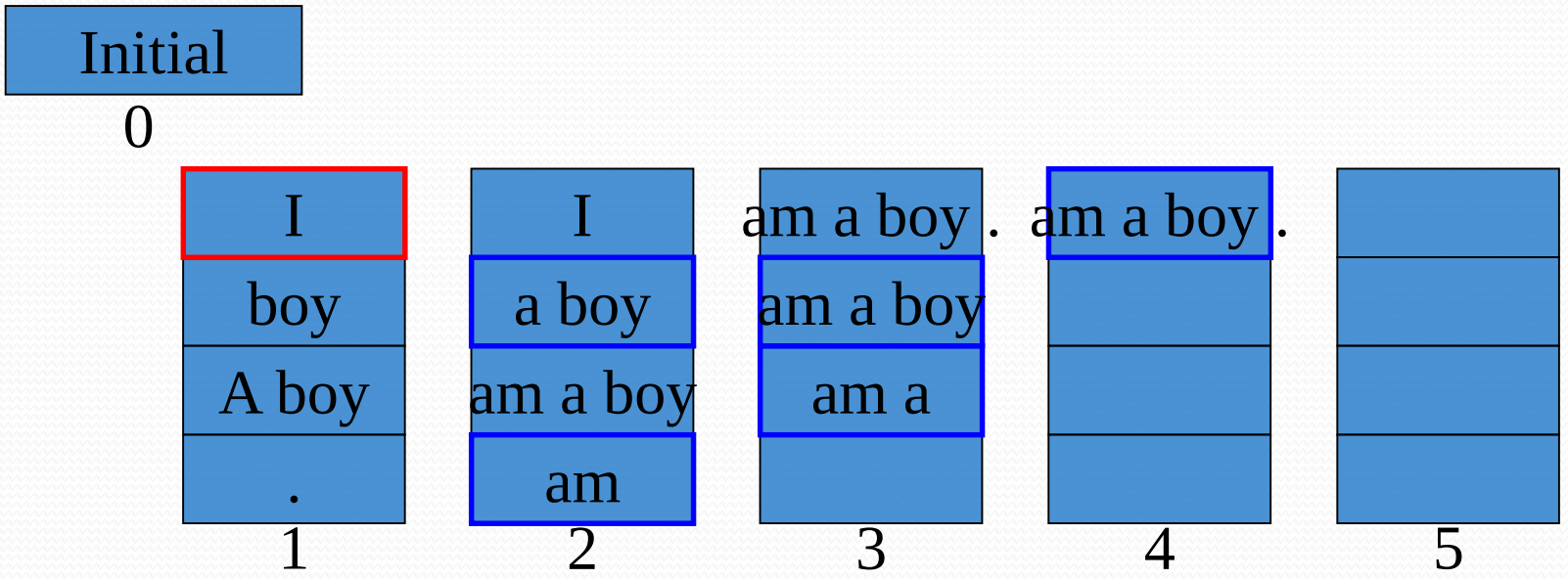
# Pharaoh Decoding

- Decoding
  - Derive new hypothesis from previous hypothesis by applying possible translation options
  - If a stack becomes full, prune worst hypothesis



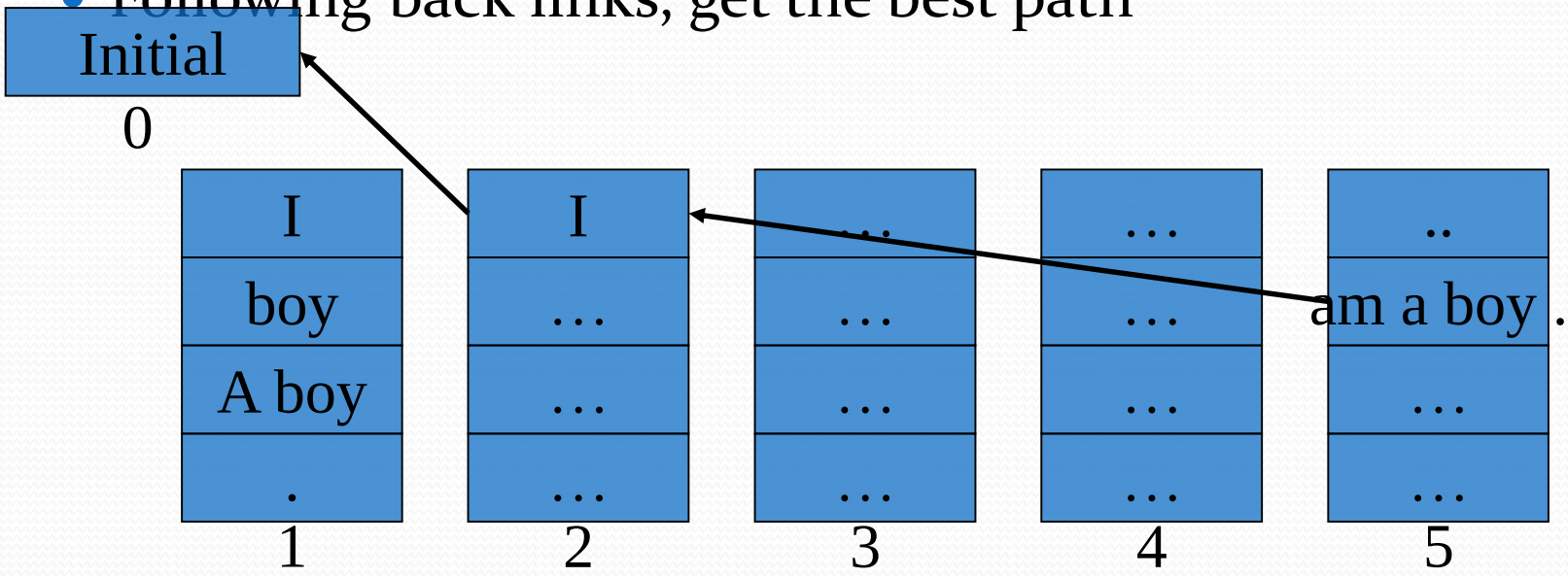
# Pharaoh Decoding

- Decoding
  - Apply translation options for each hypothesis in stack1~4.



# Pharaoh Decoding

- Decoding
  - After processing last element of stack4
  - Fine the best hypothesis in the stack5
  - Following back links, get the best path

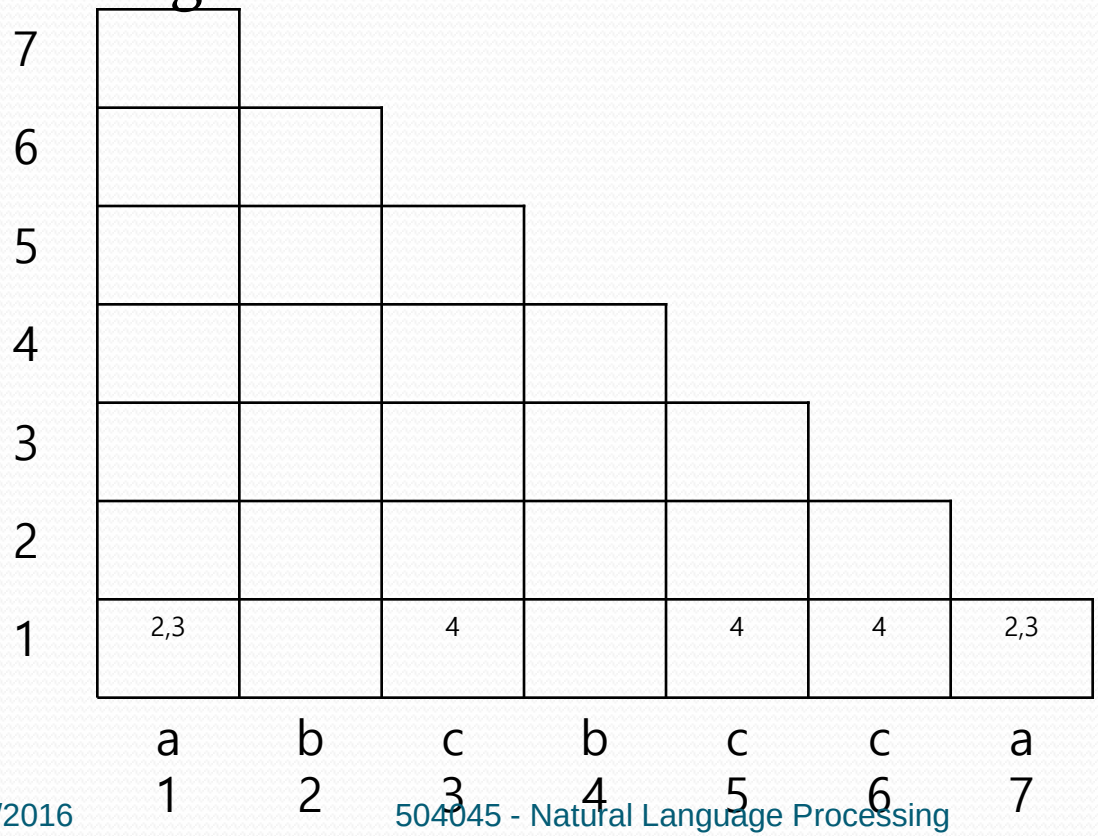


# Hiero Decoding

- Goal
  - CKY Algorithm
    - Reject or Accept a string for given grammar rules
  - Decoding for Translation
    - Ultimate goal: Get Most probable string
    - Practical goal: Get Most probable derivation

# Hiero Decoding

- Example parsing
  - length 1

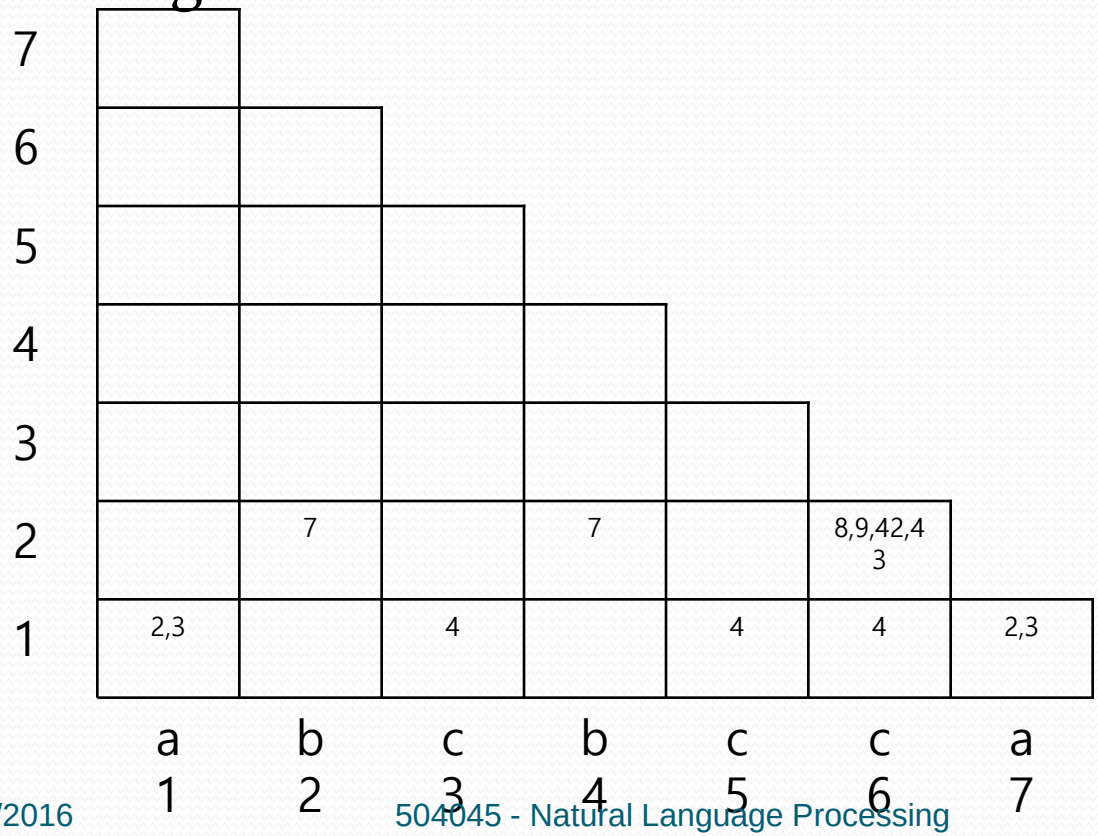



Example grammar

R0:S→(X,X) :1.0
R1:S→(SX,SX) :0.7
R2:X→(a,¬) : 0.4
R3:X→(a, ⊏) : 0.3
R4:X→(c,∟) :0.7
R5:X→(abXb,X¬⊏):0.1
R6:X→(bcX,¬X⊏) :0.2
R7:X→(bc,¬):0.3

# Hiero Decoding

- Example parsing
  - length 2

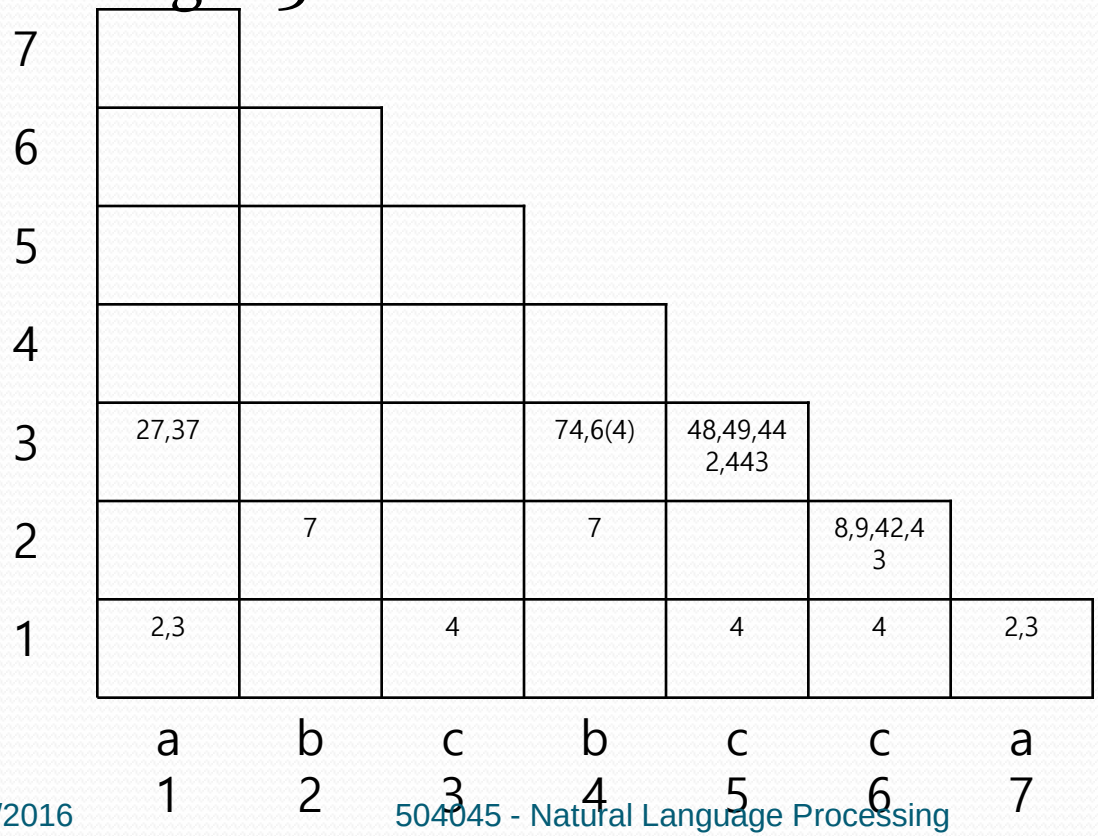



Example grammar

R0:S→(X,X) :1.0
R1:S→(SX,SX) :0.7
R2:X→(a,ㄱ) : 0.4
R3:X→(a, ㄷ) : 0.3
R4:X→(c,ㄴ) :0.7
R5:X→(abXb,Xㄱㄷ):0.1
R6:X→(bcX,ㄱXㄷ) :0.2
R7:X→(bc,ㄱ):0.3

# Hiero Decoding

- Example parsing
  - length 3

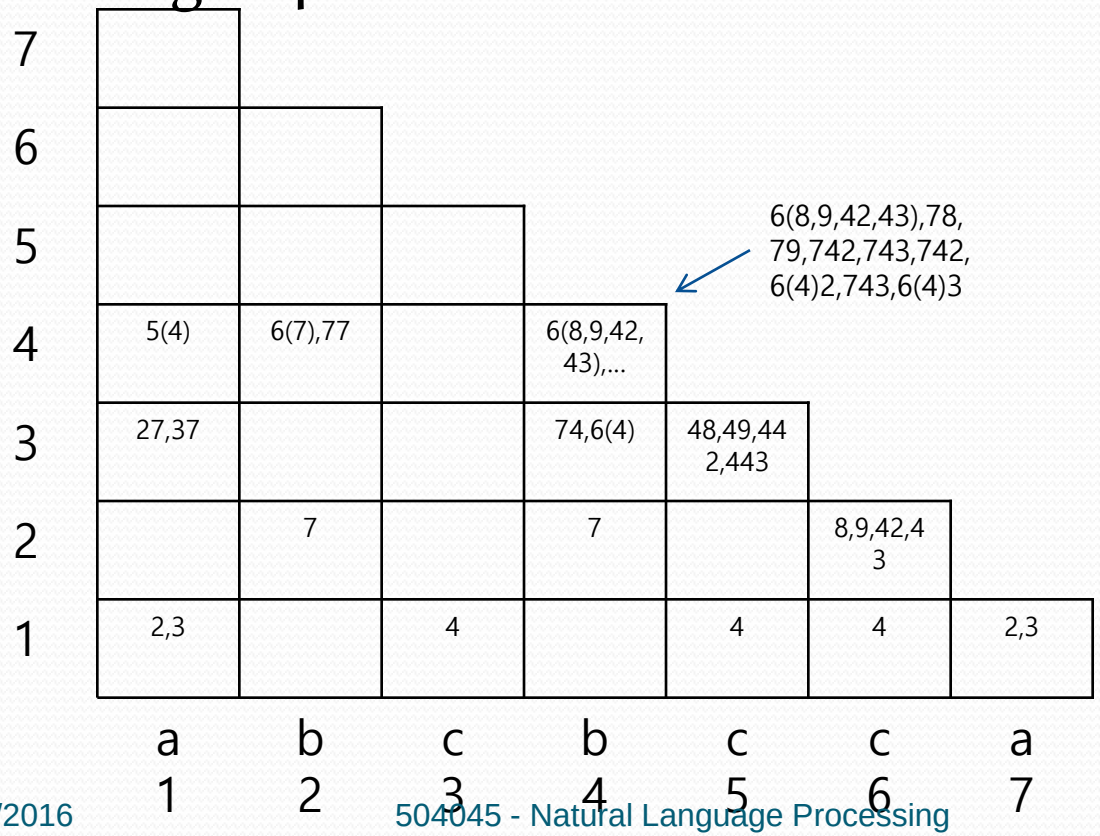



Example grammar

R0:S→(X,X) :1.0
R1:S→(SX,SX) :0.7
R2:X→(a,¬) : 0.4
R3:X→(a, ⊏) : 0.3
R4:X→(c,∟) :0.7
R5:X→(abXb,X¬⊏):0.1
R6:X→(bcX,¬X⊏) :0.2
R7:X→(bc,¬):0.3

# Hiero Decoding

- Example parsing
  - length 4

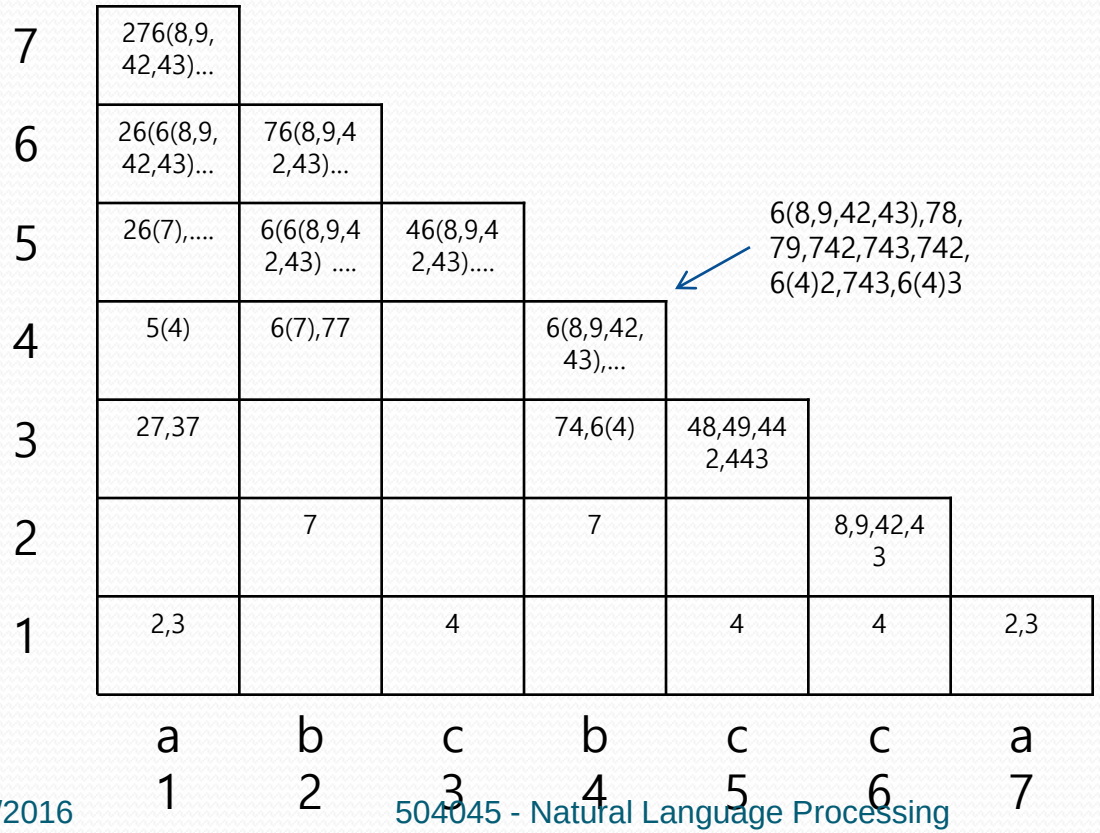



Example grammar

R0:S→(X,X) :1.0
R1:S→(SX,SX) :0.7
R2:X→(a,¬) : 0.4
R3:X→(a, ⊏) : 0.3
R4:X→(c,∟) :0.7
R5:X→(abXb,X¬⊏):0.1
R6:X→(bcX,¬X⊏) :0.2
R7:X→(bc,¬):0.3

# Hiero Decoding

- Example parsing
  - End




Example grammar

R0:S→(X,X) :1.0
R1:S→(SX,SX) :0.7
R2:X→(a,¬) : 0.4
R3:X→(a, ⊏) : 0.3
R4:X→(c,∟) :0.7
R5:X→(abXb,X¬⊏):0.1
R6:X→(bcX,¬X⊏) :0.2
R7:X→(bc,¬):0.3

# Hiero Decoding

- Read Out the first cell
  - 276(8,9,42,43), ….
    - 276(8)
      - string: ㄱㄱㄱㄷㄱㄷ
      - score: 0.4*0.3*0.2*0.4*LM Score
    - 276(9)
      - string:ㄱㄱㄱㄷㄱㄴㄷ
    - 276(42)
      - string:ㄱㄱㄱㄴㄱㄷ
    - 276(43)
      - string:ㄱㄱㄱㄴㄷㄷ
  - Other derivations

Example grammar

R0:S→(X,X) :1.0
R1:S→(SX,SX) :0.7
R2:X→(a,ㄱ) : 0.4
R3:X→(a, ㄷ) : 0.3
R4:X→(c,ㄴ) :0.7
R5:X→(abXb,Xㄱㄷ):0.1
R6:X→(bcX,ㄱXㄷ) :0.2
R7:X→(bc,ㄱ):0.3

# Contents

- What Samsung requires to survey
  - Rule-base and Statistical approach
  - Alignment Model
  - Decoding Algorithms
  - Open Sources
  - Evaluation Methods
  - Using Syntax Info. in SMT

# Open Sources

- GIZA++
  - Franz Josef Och, 2000
  - Most SMT researchers use GIZA++
  - Much research on alignment start from IBM Model and HMM alignment model
  - A C++ Implementation of
    - IBM model 1~5
    - HMM alignment model
    - Smoothing for fertility, distortion/alignment parameters
    - Some improvements of IBM and HMM models
  - License : GPL
  - http://www.fjoch.com/GIZA++.html

# Open Sources

- Sri-LM
  - A. Stolcke, 2002
  - Implements State-of-the-art LM techniques
    - The latest update 1.5.3 (2007.07)
  - A C++ Implementation of
    - N-gram language modeling
    - Kneser-Ney discounting
    - Witten-Bell discounting
    - ….
  - License: SRILM Research Community License
  - http://www.speech.sri.com/projects/srilm/

# Open Sources

- Moses
  - Philipp Koehn et. al. 2007
  - State-of-the art SMT system
  - C++ & Perl implementation of
    - Phrase-based SMT ( Pharaoh )
    - Factor phrase-based decoder
    - Minimum error rate training
    - Translation Model training
  - License : LGPL
  - http://www.statmt.org/moses/

# Open Sources

- GenPar Toolkit
  - C++ implementation
  - Translation Performance is not so good
  - SMT by parsing
    - Training by parallel parsing on target and source language
    - Decoding by CKY style parsing algorithm
  - License : GPL 2.0 or later
  - http://nlp.cs.nyu.edu/GenPar/

# Open Sources

- Phramer
  - Marian Olteanu, 2006
  - Java implementation of
    - Phrase-based machine translation
    - MERT training of MT
  - License : Free 
  - http://www.utdallas.edu/~mgo031000/phramer/

# Contents

- Part1: What Samsung requires to survey
  - Rule-base and Statistical approach
  - Alignment Model
  - Decoding Algorithms
  - Open Sources
  - Evaluation Methods
  - Using Syntax Info. in SMT

# Automatic Evaluation

- Advantages of automatic evaluation
  - Fast, Low Cost
  - Objective
- Evaluation methods
  - BLEU Score: Bi-Lingual Evaluation Understudy Score
    - Geometric mean of modified n-gram precision
  - NIST Score:
    - Arithmetic mean of modified n-gram precision
  - METEOR Score: Metric for Evaluation of Translation With Explicit Ordering
  - WER :Word Error Rate
  - PER :Position independent word Error Rate
  - TER :Translation Error Rate
  - Others ..

# Automatic Evaluation: examples

- BLEU score
  - Most famous metric
  - Range 0~1.
  - Higher score means better translation
  - Typically, consider up to 4-gram
    - denote BLEU-4 score

$$\mathrm{BP} = \begin{cases} 1 & \text{if } c > r \\ e^{(1-r/c)} & \text{if } c \le r \end{cases}$$

c : length of candidate translation

r : length of reference sentence

$$\mathrm{BLEU} = \mathrm{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right)$$

BP: factor related to the length of candidate translation

$p_n$: n-gram precision, ignoring duplicate count

N: maximum order of n-gram

# Automatic Evaluation: examples

- NIST score
  - Similar to BLEU metric
  - Higher score means better translation
  - Arithmetic mean of n-gram precisions
  - β is chosen to make the BP =0.5 when $L_{sys}/L_{ref} = 2/3$

$$BP = \exp(\beta \log_2 \min(\frac{L_{sys}}{L_{ref}}, 1))$$

$L_{sys}$ : length of candidate translation

$L_{ref}$ :average length of reference sentences

$$Info(w_1 \dots w_n) = \log_2 (\frac{\text{the \# of occurrences of } w_1 \dots w_n}{\text{the \# of occurrences of } w_1 \dots w_n})$$

$$NIST = BP \cdot \sum_{n=1}^{N} \frac{\sum Info(w_1 \dots w_n)}{\sum_{\text{words in output}} 1}$$

# Automatic Evaluation: examples

- METEOR Score
  - Not very popular
  - Based on uni-gram precision and recall
  - Chunks : A sequence of uni-grams those are adjacent in both reference and system output

$$\text{F-mean} = \frac{10\,\text{Precision}}{\text{Recall} + 9\,\text{Precision}}$$

$$\text{Penalty} = 0.5\left(\frac{\#\text{ of chunks}}{\#\text{ of unigrams matched}}\right)^3$$

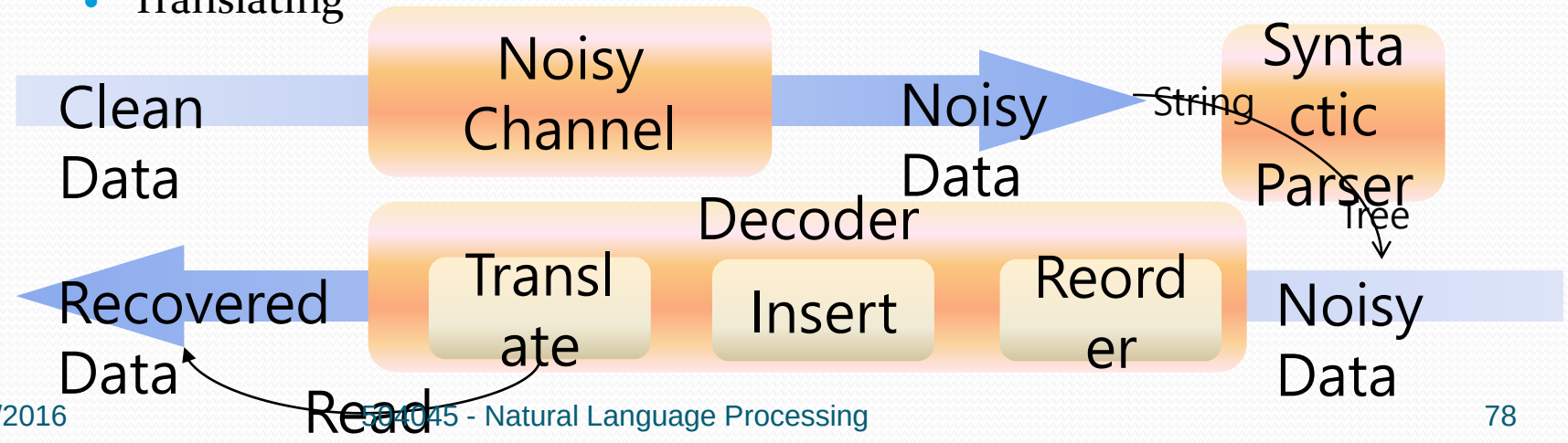
$$\text{Meteor} = \text{F-mean}\,(1 - \text{Penalty})$$

# Contents

- What Samsung requires to survey
  - Rule-base and Statistical approach
  - Alignment Model
  - Decoding Algorithms
  - Open Sources
  - Evaluation Methods
  - Using Syntax Info. in SMT

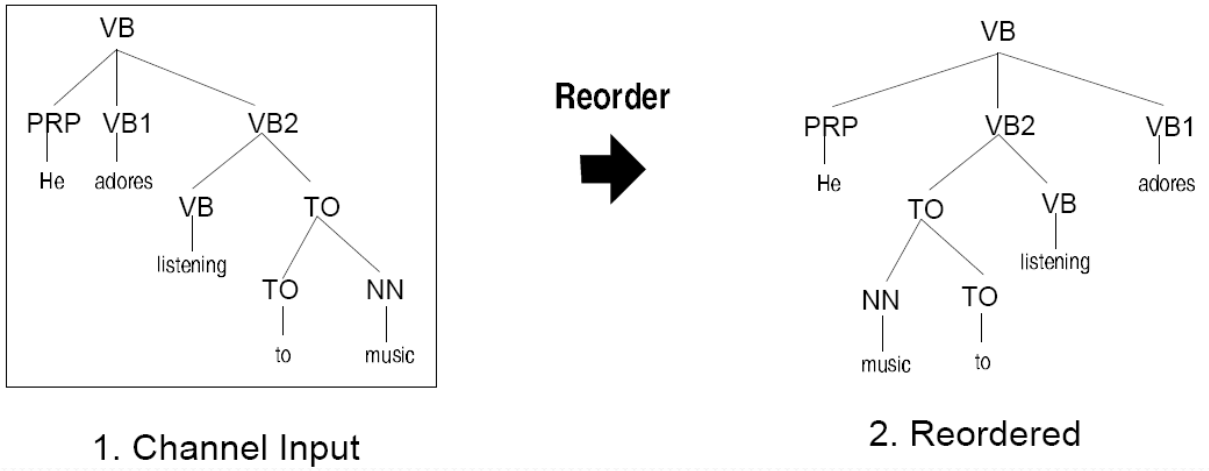
# Syntax-based Statistical Translation

- K. Yamada and K. Knight [2001] proposed a method
- Modified source-channel model
  - Input
    - Sentences → Parse trees
    - Input sentences are preprocessed by a syntactic parser
  - Channel operation
    - Reordering
    - Inserting
    - Translating



# Syntax-based MT: Process

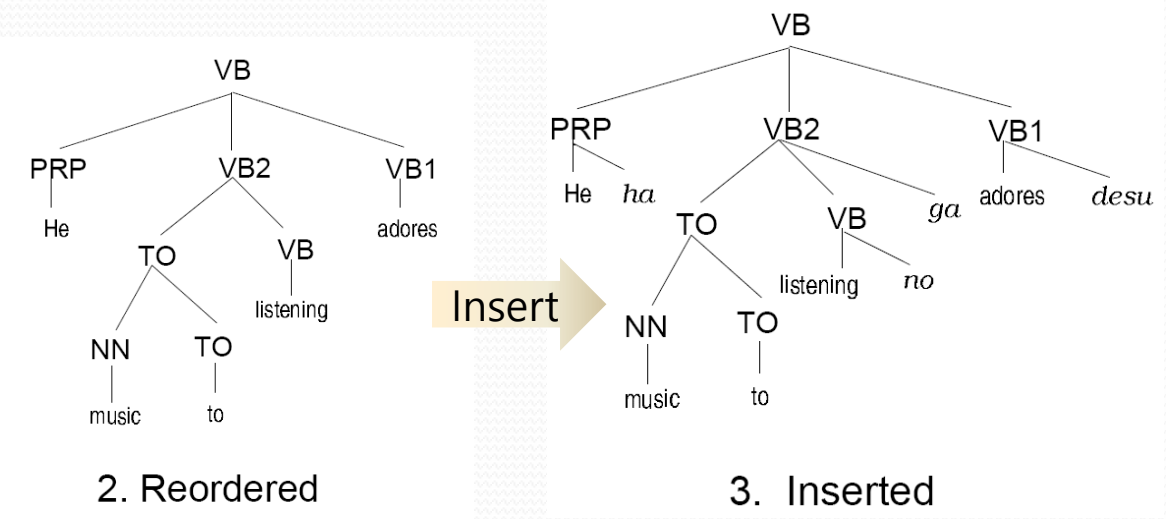
- Original sentence is processed by a syntactic parser
- Start with a parse tree
- Step1: Reordering
  - Assume that only the sequence of child node labels influences the reordering
  - The probability of reordering is given by r-table ( reordering model)
    - e.g. 0.723(PRP VB1 VB2→PRP VB2 VB1) * 0.749(VB TO→TO VB)* 0.893(TO NN → NN TO) = 0.484




| Original order | Reordering | P(reorder) |
|---|---|---|
| PRP VB1 VB2 | PRP VB1 VB2<br>PRP VB2 VB1<br>VB1 PRP VB2<br>VB1 VB2 PRP<br>VB2 PRP VB1<br>VB2 VB1 PRP | 0.074<br>0.723<br>0.061<br>0.037<br>0.083<br>0.021 |
| VB TO | VB TO<br>TO VB | 0.251<br>0.749 |
| TO NN | TO NN<br>NN TO | 0.107<br>0.893 |
| … | … | … |

These figures came from the original paper [Yamada and Knight, "A Syntax Based Translation Model",2001]

# Syntax-based MT: Process




## Calculation of this example

(0.652*0.219)*(0.252*0.094)(0.252*.0.62)*(0.252*0.0007)*(0.735*0.709)*(0.900*0.800) =3.498e-9

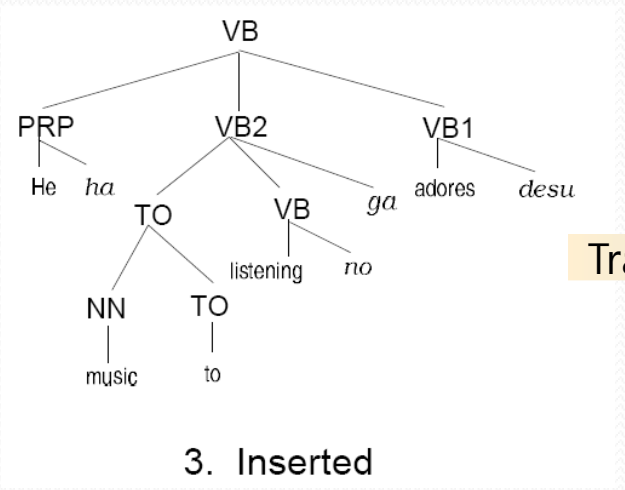
- Insertion probability is defined by n-table
- n-table is divided into two
  - Table for position
  - Table for word identity

| Parent | TOP | VB | VB | VB | TO | TO | ... |
|---|---|---|---|---|---|---|---|
| Node | VB | VB | PRP | TO | TO | NN | ... |
| P(None) | 0.735 | 0.687 | 0.344 | 0.709 | 0.900 | 0.800 | ... |
| P(Left) | 0.004 | 0.061 | 0.004 | 0.030 | 0.003 | 0.096 | ... |
| P(Right) | 0.260 | 0.252 | 0.652 | 0.261 | 0.007 | 0.104 | ... |

| W | P(ins-w) |
|---|---|
| ha | 0.219 |
| ta | 0.131 |
| wo | 0.099 |
| no | 0.094 |
| ni | 0.080 |
| te | 0.078 |
| ga | 0.062 |
| ... | ... |
| desu | 0.0007 |
| ... | ... |

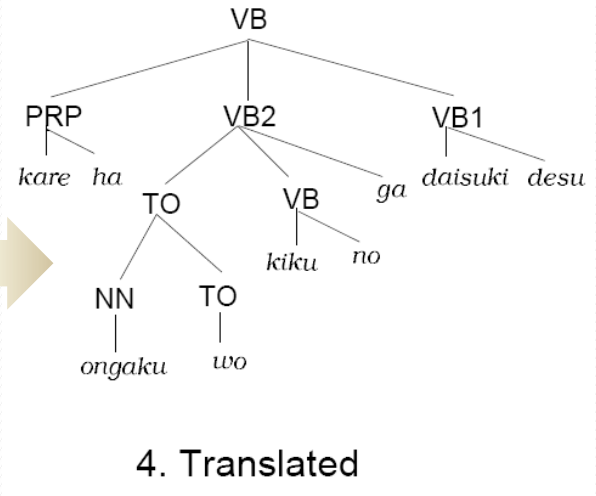
These figures came from the original paper[Yamada and Knight, "A Syntax Based Translation Model",2001]

# Syntax-based MT: Process




3. Inserted

Trans




4. Translated

Calculation of this example

0.952*0.900*.0038*0.333*1.000=0.0108

- The translate operation is applied to each leaf
- This operation is dependent only on the word itself

| E | adores | | he | | I | | listening | | Music | | to | | ... |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| J | daisuki | 1.000 | kare | 0.95 | NUL | 0.471 | kiku | 0.333 | ongaku | 0.900 | ni | 0.216 | ... |
| | | | NULL | 0.016 | L | 0.111 | kii | 0333 | naru | 0.100 | NULL | 0.204 | |
| | | | nani | 0.005 | watas | 0.055 | mi . | 0333 | | | to | 0.133 | |
| | | | da | 0.003 | i | 0.021 | | | | | no | 0.046 | |
| | | | shi | 0.003 | kare | 0.020 | | | | | wo | 0.038 | |
| | | | ... | ... | shi | | | | | | ... ... | | |
| | | | | | nani | | | | | | | | |
| | | | | | ... ... | | | | | | | | |

These figures came from the original paper[Yamada and Knight, "A Syntax Based Translation Model",2001]

# Hierarchical Modeling

- Hierarchical organization of Natural Language
  - A sentence is derived by recursive application of some production rules
  - S yields NP and VP
  - VP may yield another NP
  - …

- Traditional Statistical Systems
  - A sentence is generated by sequentially concatenating some phrases

  - We need to model the Hierarchical property of language

# Hiero

- Hiero
  - A Hierarchical Phrase based Statistical Machine Translation System
  - Automatically extracts production rules from un-annotated parallel texts
  - Finds the best derivation for a given sentence using Modified CKY beam search decoder
  - Grammar
    - Form of synchronous CFG
    - Can be Automatically extracted from parallel texts
  - Model
    - Use log-linear model
    - Assign a weight for each rule
    - Goal is findi total weight

$$w(X \rightarrow \langle \gamma, \alpha \rangle) = \prod_i \phi_i(X \rightarrow \langle \gamma, \alpha \rangle)^{\lambda_i}$$

# Synchronous Grammar

- A synchronous CFG
  - Consists of a pair of CFG rules with aligned non-terminal symbols
  - Derivation starts with a pair of start symbols

- A Partial Derivation

Grammar example

| | |
|---|---|
| 1 | $S \rightarrow \langle S_{[1]} X_{[2]}, S_{[1]} X_{[2]} \rangle$ |
| 2 | $S \rightarrow \langle X_{[1]}, X_{[1]} \rangle$ |
| 3 | $X \rightarrow \langle$ yu $X_{[1]}$ you $X_{[2]}$, have $X_{[2]}$ with $X_{[1]} \rangle$ |
| 4 | $X \rightarrow \langle X_{[1]}$ de $X_{[2]}$, the $X_{[2]}$ that $X_{[1]} \rangle$ |
| 5 | $X \rightarrow \langle X_{[1]}$ zhiyi, one of $X_{[1]} \rangle$ |
| 6 | $X \rightarrow \langle$ Aozhou, Australia $\rangle$ |
| 7 | $X \rightarrow \langle$ shi, is $\rangle$ |
| 8 | $X \rightarrow \langle$ shaoshu guojia, few countries $\rangle$ |
| 9 | $X \rightarrow \langle$ bangjiao, diplomatic relations $\rangle$ |
| 10 | $X \rightarrow \langle$ Bei Han, North Korea $\rangle$ |

$\langle S_{[1]}, S_{[1]} \rangle$
$\overset{1}{\Rightarrow} \langle S_{[2]} X_{[3]}, S_{[2]} X_{[3]} \rangle$
$\overset{1}{\Rightarrow} \langle S_{[4]} X_{[5]} X_{[3]}, S_{[4]} X_{[5]} X_{[3]} \rangle$
$\overset{2}{\Rightarrow} \langle X_{[6]} X_{[5]} X_{[3]}, X_{[6]} X_{[5]} X_{[3]} \rangle$
$\overset{6}{\Rightarrow} \langle$ Aozhou $X_{[5]} X_{[3]}$, Australia $X_{[5]} X_{[3]} \rangle$
$\overset{7}{\Rightarrow} \langle$ Aozhou shi $X_{[3]}$, Australia is $X_{[3]} \rangle$
$\overset{5}{\Rightarrow} \langle$ Aozhou shi $X_{[7]}$ zhiyi, Australia is one of $X_{[7]} \rangle$
$\overset{4}{\Rightarrow} \langle$ Aozhou shi $X_{[8]}$ de $X_{[9]}$ zhiyi, Australia is one of the $X_{[9]}$ that $X_{[8]} \rangle$
$\overset{3}{\Rightarrow} \langle$ Aozhou shi yu $X_{[1]}$ you $X_{[2]}$ de $X_{[9]}$ zhiyi, Australia is one of the $X_{[9]}$ that have $X_{[2]}$ with $X_{[1]} \rangle$