Machine Translation (Part 1)

CS 585, Fall 2015
Introduction to Natural Language Processing
http://people.cs.umass.edu/~brenocon/inlp2015/

Brendan O'Connor
College of Information and Computer Sciences
University of Massachusetts Amherst

[Some slides borrowed from J&M + mt-class.org]

NLP news of the day

The system helps Mountain View, California-based Google deal with the 15 percent of queries a day it gets which its systems have never seen before, he said. For example, it's adept at dealing with ambiguous queries, like, "What's the title of the consumer at the highest level of a food chain?" And RankBrain's usage of AI means it works differently than the other technologies in the search engine.

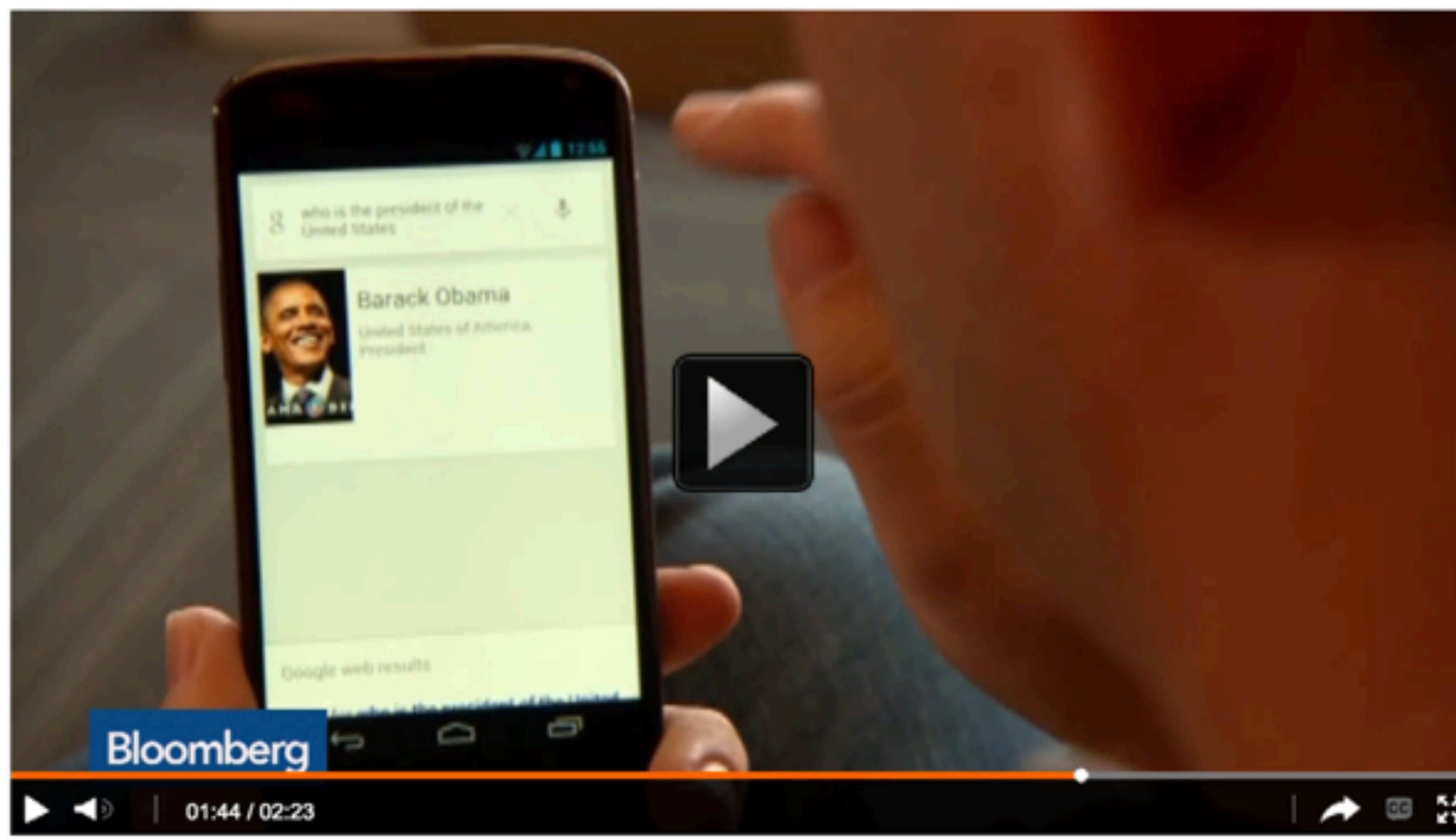
"The other signals, they're all based on discoveries and insights that people in information retrieval have had, but there's no learning," Corrado said.

BloombergBusiness | News | Markets | Insights | Video

Google Turning Its Lucrative Web Search Over to AI Machines

by Jack Clark

October 26, 2015 – 5:00 AM EDT



http://www.bloomberg.com/news/articles/2015-10-26/google-turning-its-lucrative-web-search-over-to-ai-machines

Graders at work

- Midterms back on Thursday
- Project feedback: by tomorrow
- HW2 still underway (sorry!)

Machine translation

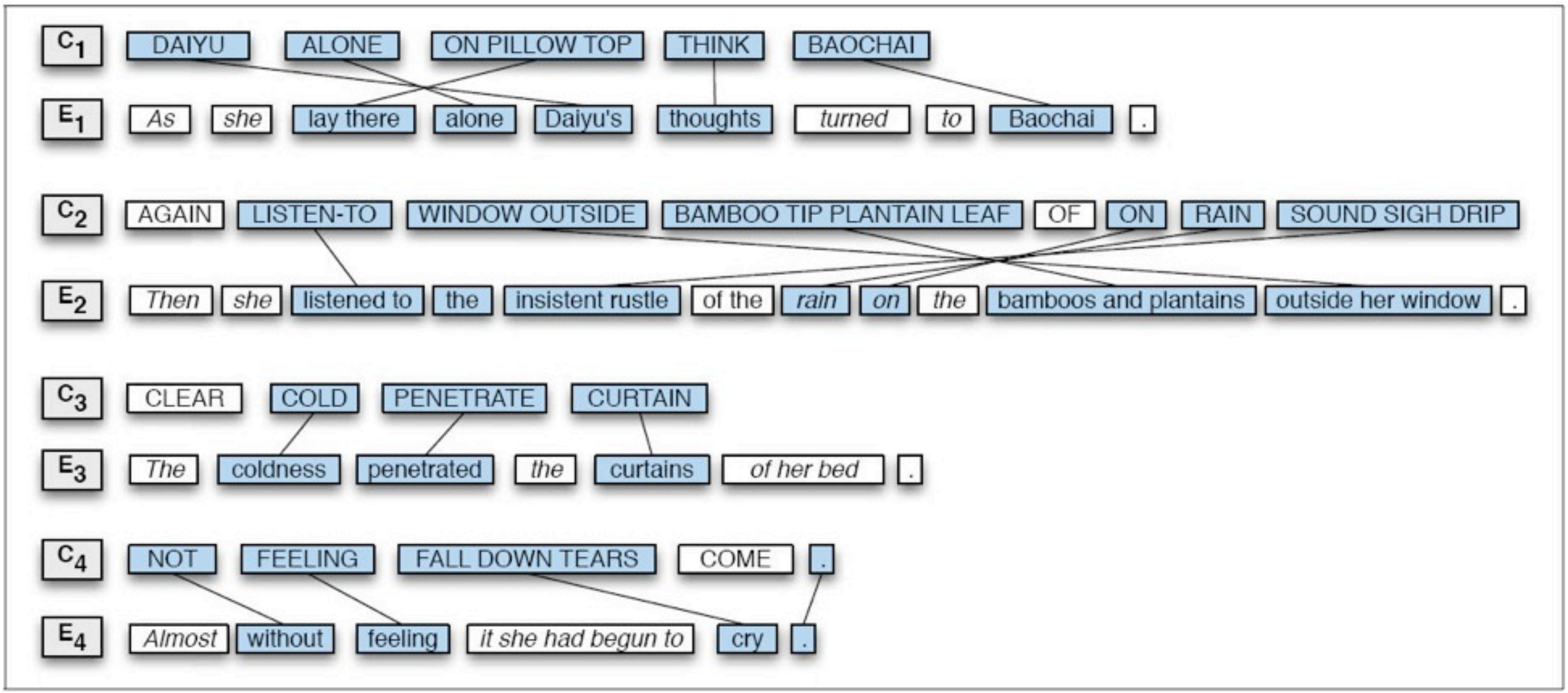
- Intro
- Classic MT
- Statistical MT

- Training
- Evaluation

MT is amazing

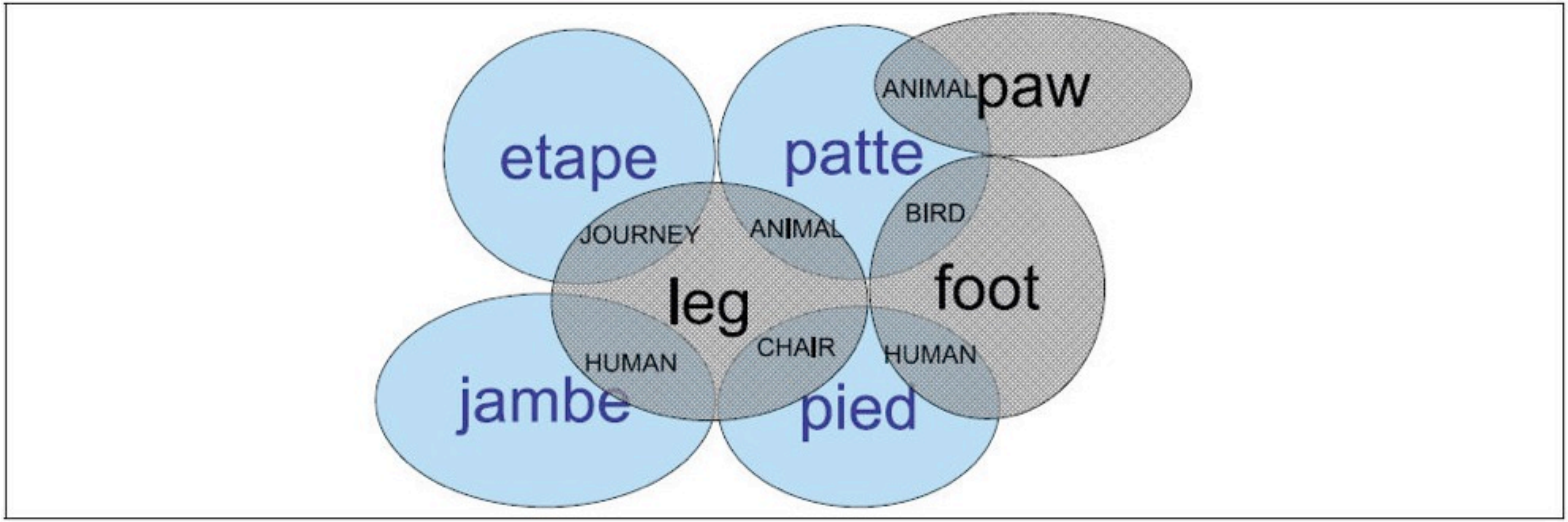
MT is hard

- Word order, word meanings



MT is hard

- Word meaning:
  many-to-many and context dependent



- *Translation* itself is hard: metaphors, cultural references, etc.

MT goals

- Motivation: Human translation is expensive

- High precision translation
- Rough translation
- Assistance for human translators
  - Comparison: bilingual dictionary

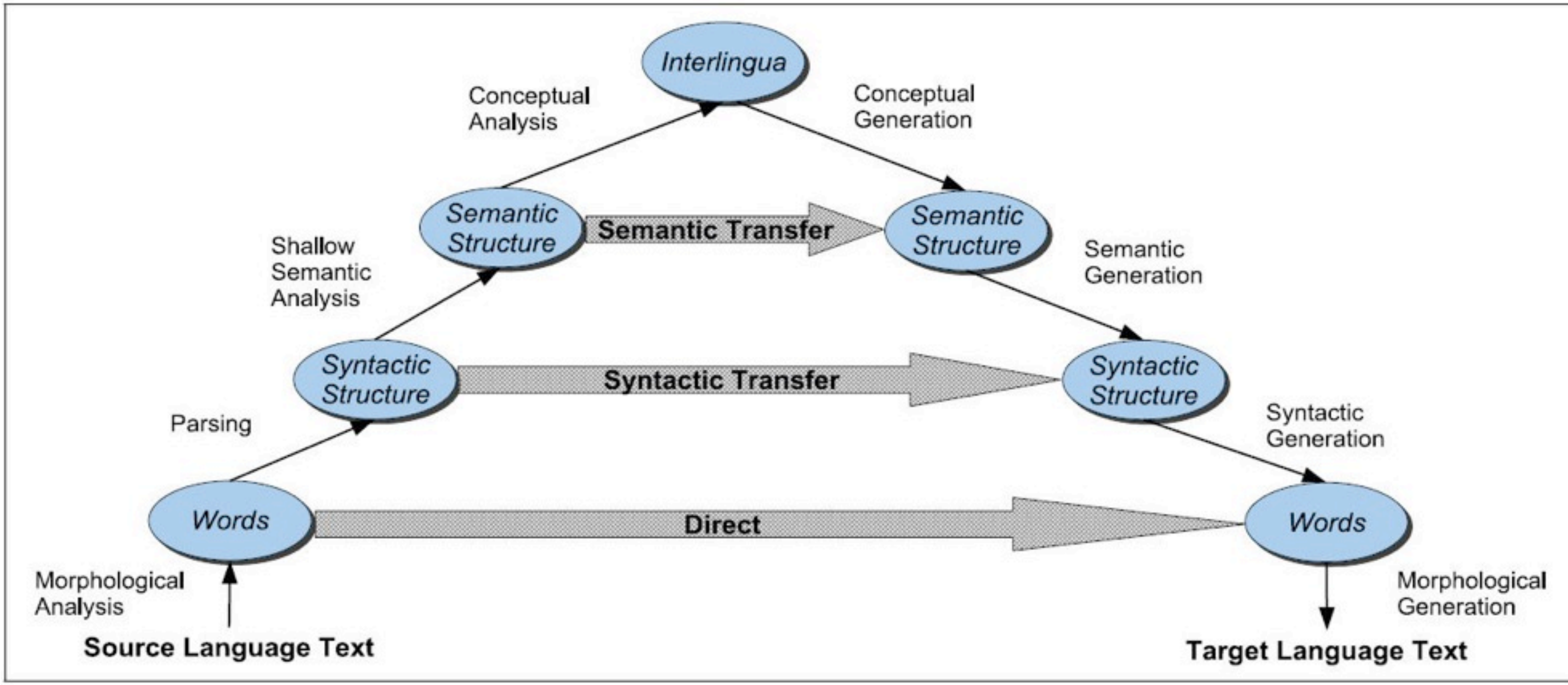
MT: major types

- Rule-based transfer
  - Manually program lexicons/rules
  - SYSTRAN (AltaVista Babelfish)

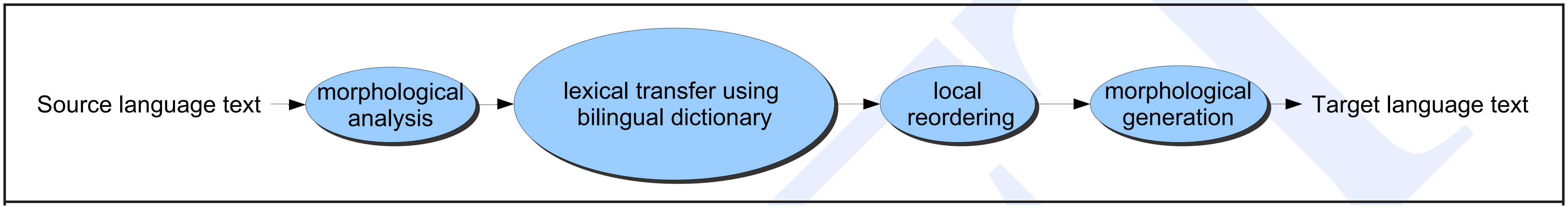
- Statistical MT:
  - Learn translation rules from data,
    search for high-scoring translation outputs
    - Phrase or syntactic transformations
  - Key research in the early 90s
  - Google Translate (mid 00s)
  - Moses, cdec (open-source)

- [Active current work: Semantic MT? Neural MT?]

Vauquois Triangle



Direct (word-based) transfer



Input:	Mary didn't slap the green witch
After 1: Morphology	Mary DO-PAST not slap the green witch
After 2: Lexical Transfer	Maria PAST no dar una bofetada a la verde bruja
After 3: Local reordering	Maria no dar PAST una bofetada a la bruja verde
After 4: Morphology	Maria no dió una bofetada a la bruja verde

Syntactic transfer

Nominal		⇒	Nominal	
Adj	Noun		Noun	Adj

Figure 25.10 A simple transformation that reorders adjectives and nouns

	English to Spanish:		
1.	NP $\rightarrow$ Adjective$_1$ Noun$_2$	$\Rightarrow$	NP $\rightarrow$ Noun$_2$ Adjective$_1$
	Chinese to English:		
2.	VP $\rightarrow$ PP[+Goal] V	$\Rightarrow$	VP $\rightarrow$ V PP[+Goal]
	English to Japanese:		
3.	VP $\rightarrow$ V NP	$\Rightarrow$	VP $\rightarrow$ NP V
4.	PP $\rightarrow$ P NP	$\Rightarrow$	PP $\rightarrow$ NP P
5.	NP $\rightarrow$ NP$_1$ Rel. Clause$_2$	$\Rightarrow$	NP $\rightarrow$ Rel. Clause$_2$ NP$_1$

Interlingua

"Mary did not slap the green witch"

$$\begin{bmatrix} \text{EVENT} & \text{SLAPPING} \\ \text{AGENT} & \text{MARY} \\ \text{TENSE} & \text{PAST} \\ \text{POLARITY} & \text{NEGATIVE} \\ \text{THEME} & \begin{bmatrix} \text{WITCH} & \\ \text{DEFINITENESS} & \text{DEF} \\ \text{ATTRIBUTES} & \begin{bmatrix} \text{HAS-COLOR} & \text{GREEN} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

- More like classic logic-based AI
- Works in narrow domains
- Broad domain currently fails
    - Coverage: Knowledge representation for all possible semantics?
    - Can you parse to it?
    - Can you generate from it?

Rules are hard

- Coverage
- Complexity (context dependence)
- Maintenance

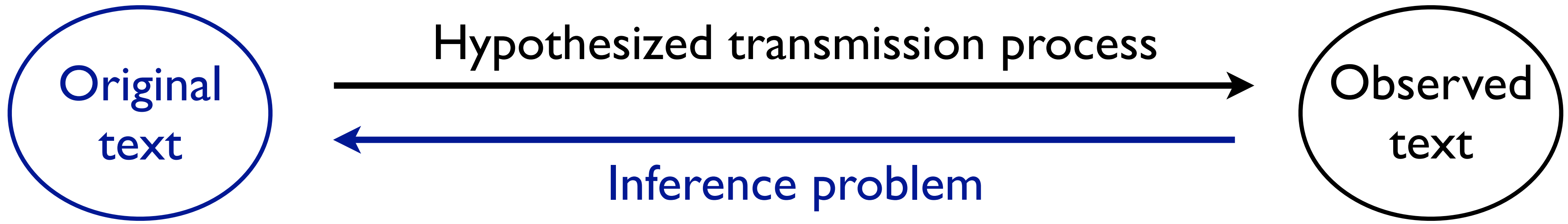
function DIRECT_TRANSLATE_MUCH/MANY(word) **returns** Russian translation

if preceding word is *how* **return** *skol'ko*
else if preceding word is *as* **return** *stol'ko zhe*
else if word is *much*
  if preceding word is *very* **return** nil
  else if following word is a noun **return** *mnogo*
else /* word is many */
  if preceding word is a preposition and following word is a noun **return** *mnogii*
  else return *mnogo*

Statistical MT

- MT as ML: Translation is something people do naturally. Learn rules from data?
- Parallel data: (source, target) text pairs
  - E.g. 20 million words of European Parliament proceedings
    http://www.statmt.org/europarl/

Noisy channel model



Codebreaking

P(plaintext | encrypted text) $\propto$ P(encrypted text | plaintext) P(plaintext)

Speech recognition

P(text | acoustic signal) $\propto$ P(acoustic signal | text) P(text)

Optical character recognition

P(text | image) $\propto$ P(image | text) P(text)

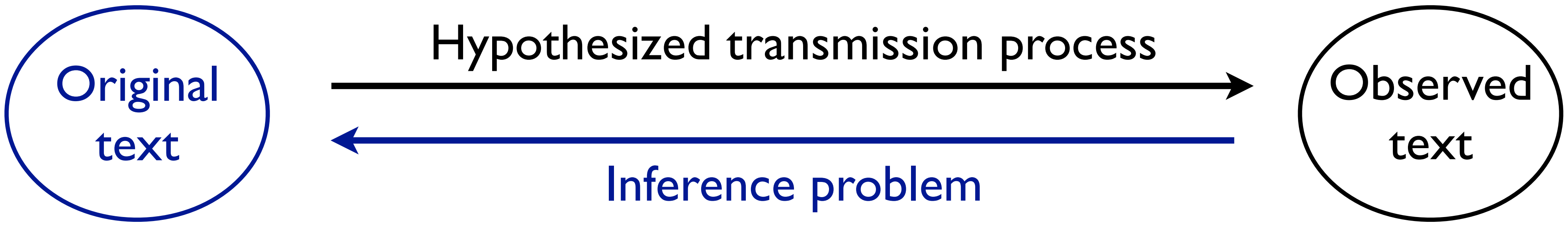
Machine translation

P(target text | source text) $\propto$ P(source text | target text) P(target text)

Spelling correction

P(target text | source text) $\propto$ P(source text | target text) P(target text)

Noisy channel model



Codebreaking

P(plaintex ... ext)

Speech

P(text | ac

Optical

P(text | im

> One naturally wonders if the problem of translation could conceivably be treated as a problem in cryptography. When I look at an article in Russian, I say: 'This is really written in English, but it has been coded in some strange symbols. I will now proceed to decode.'
>
> -- Warren Weaver (1955)

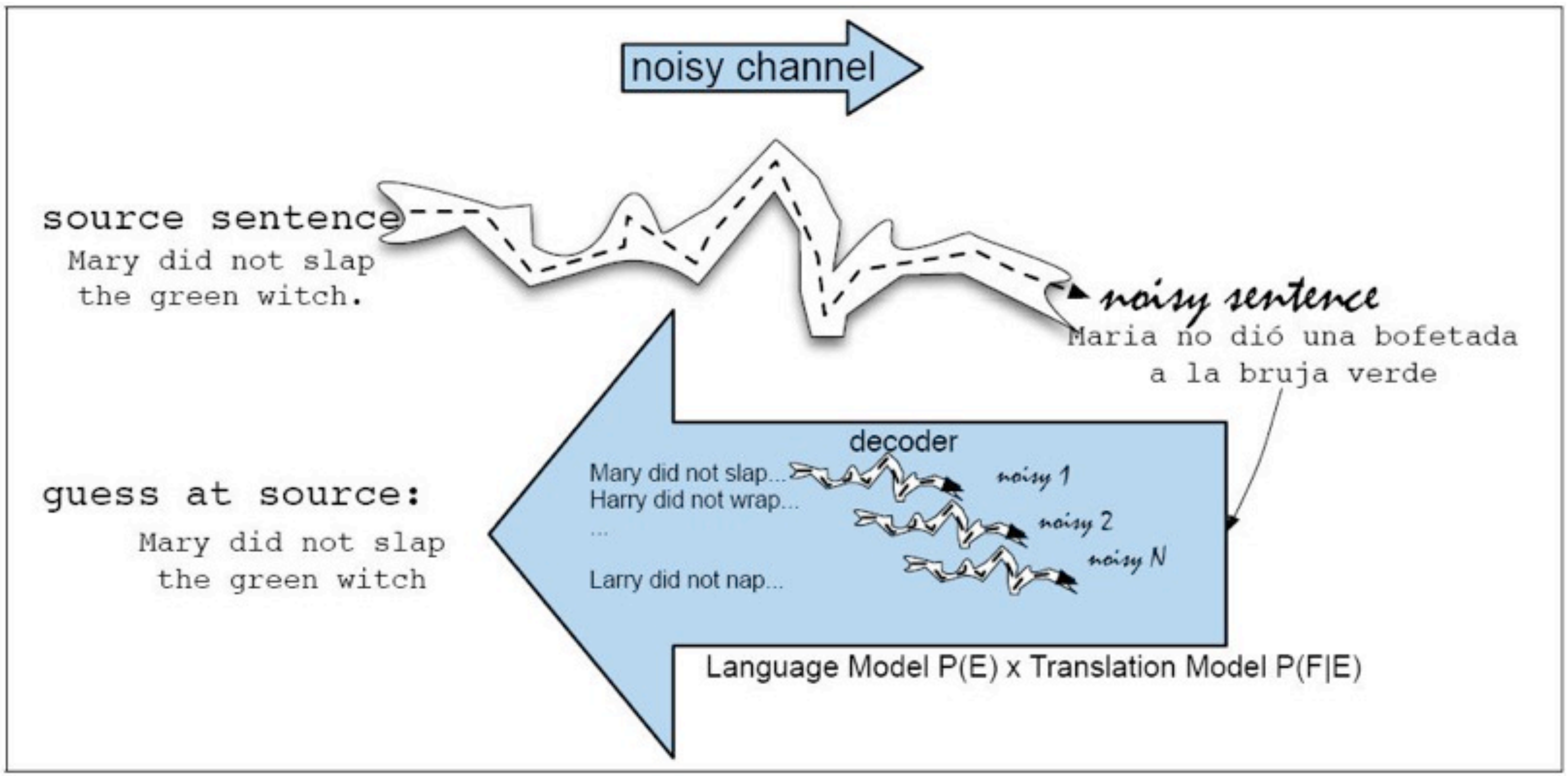
Machine translation

P(target text | source text) $\propto$ P(source text | target text) P(target text)

Spelling correction

P(target text | source text) $\propto$ P(source text | target text) P(target text)

Statistical MT



Statistical MT

- Pioneered at IBM, early 1990s
  (Forerunner of 90s-era statistical revolution in NLP)

The COLING Paper Review

The validity of statistical (information theoretic) approach to MT has indeed been recognized, as the authors mention, by Weaver as early as 1949. And was universally recognized as mistaken by 1950. (cf. Hutchins, MT: Past, Present, Future, Ellis Horwood, 1986, pp. 30ff. and references therein) The crude force of computers is not science. The paper is simply beyond the scope of COLING.

Historical notes: http://cs.jhu.edu/~post/bitext/

51

Statistical MT

- Pioneered at IBM, early 1990s
  (Forerunner of 90s-era statistical revolution in NLP)
- Noisy channel model borrowed from speech recognition processing

"Every time I fire a linguist,
the performance of the speech recognizer goes up"
[Fred Jelinek]

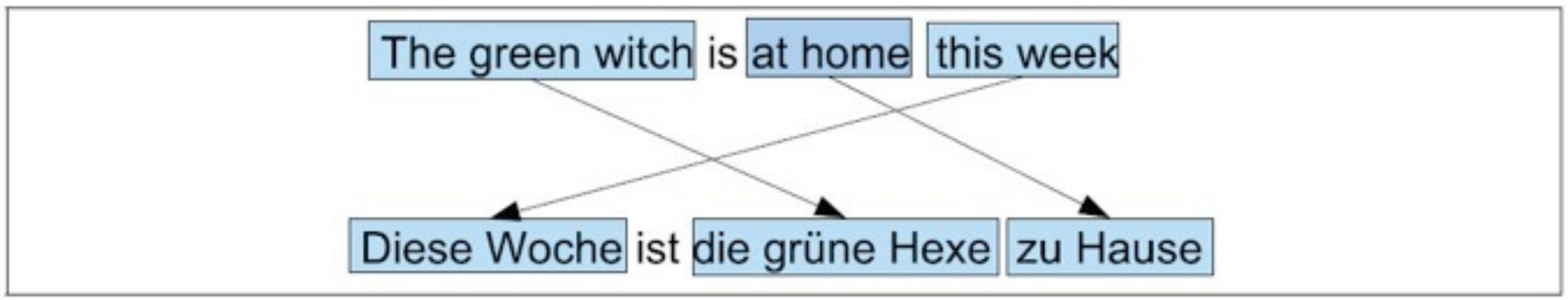
Problem formulation

best-translation $\hat{T}$ = argmax$_T$ faithfulness(T,S) fluency(T)

$$\hat{E} = \underset{E \in \text{English}}{\text{argmax}} \overbrace{P(F|E)}^{\text{translation model}} \overbrace{P(E)}^{\text{language model}}$$

decoder
(search algo)

Phrase-based model



- Learning P(F | E) phrase translation tables: Assume aligned corpus. Then count

- Today: lexical translation model (IBM Model 1)

Lexical Translation

- How do we translate a word? Look it up in the dictionary

$$\textit{Haus} : \textit{house}, \textit{home}, \textit{shell}, \textit{household}$$

- Multiple translations
    - Different word senses, different registers, different inflections (?)
    - *house, home* are common
    - *shell* is specialized (the Haus of a snail is a shell)

How common is each?

Translation	Count
house	5000
home	2000
shell	100
household	80

Maximum Likelihood Estimation: count ratios

$$\hat{p}_{\mathrm{MLE}}(e \mid \texttt{Haus}) = \begin{cases} 0.696 & \text{if } e = \texttt{house} \\ 0.279 & \text{if } e = \texttt{home} \\ 0.014 & \text{if } e = \texttt{shell} \\ 0.011 & \text{if } e = \texttt{household} \\ 0 & \text{otherwise} \end{cases}$$

Could learn ***if*** we had translation frequencies.

Lexical Translation

- Goal: a model $p(\mathbf{e} \mid \mathbf{f}, m)$
- where $\mathbf{e}$ and $\mathbf{f}$ are complete English and Foreign sentences

$$\mathbf{e} = \langle e_1, e_2, \ldots, e_m \rangle \qquad \mathbf{f} = \langle f_1, f_2, \ldots, f_n \rangle$$

Lexical Translation

- Goal: a model $p(\mathbf{e} \mid \mathbf{f}, m)$
- where $\mathbf{e}$ and $\mathbf{f}$ are complete English and Foreign sentences
- Lexical translation makes the following ***assumptions***:
    - Each word in e_i in $\mathbf{e}$ is generated from exactly one word in $\mathbf{f}$
    - Thus, we have an *alignment* a_i that indicates which word e_i "came from", specifically it came from f_{a_i}.
    - Given the alignments $\mathbf{a}$, translation decisions are conditionally independent of each other and depend *only* on the aligned source word f_{a_i}.

Lexical Translation

$$\mathbf{e} = \langle e_1, e_2, ...e_m \rangle \qquad \mathbf{f} = \langle f_1, f_2, ...f_n \rangle$$

$$\mathbf{a} = \langle a_1, a_2, ...a_m \rangle \quad \text{each } a_i \in \{0, 1, ..., n\}$$

Modeling assumptions

$$p(\mathbf{e} \mid \mathbf{f}, m) = \sum_{\mathbf{a} \in \{0,1,..,n\}^m} p(\mathbf{a} \mid \mathbf{f}, m) \times \prod_{i=1}^{m} p(e_i \mid f_{a_i})$$

[Alignment] x [Translation | Alignment]

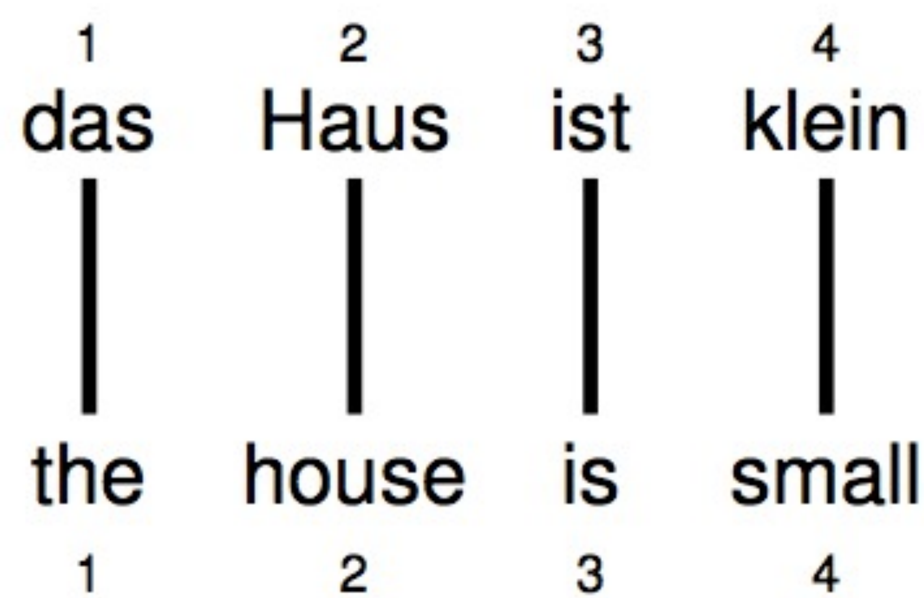
Alignment

$$p(\mathbf{a} \mid \mathbf{f}, m)$$

Most of the action for the first 10 years of MT was here. Words weren't the problem, word *order* was hard.

Alignment

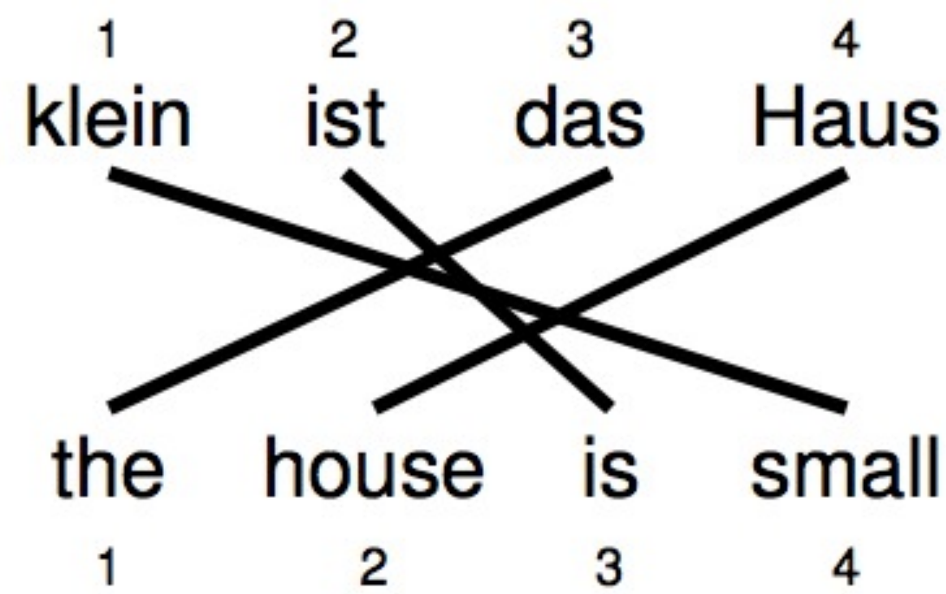
- Alignments can be visualized in by drawing links between two sentences, and they are represented as vectors of positions:



$$\mathbf{a} = (1, 2, 3, 4)^\top$$

Reordering

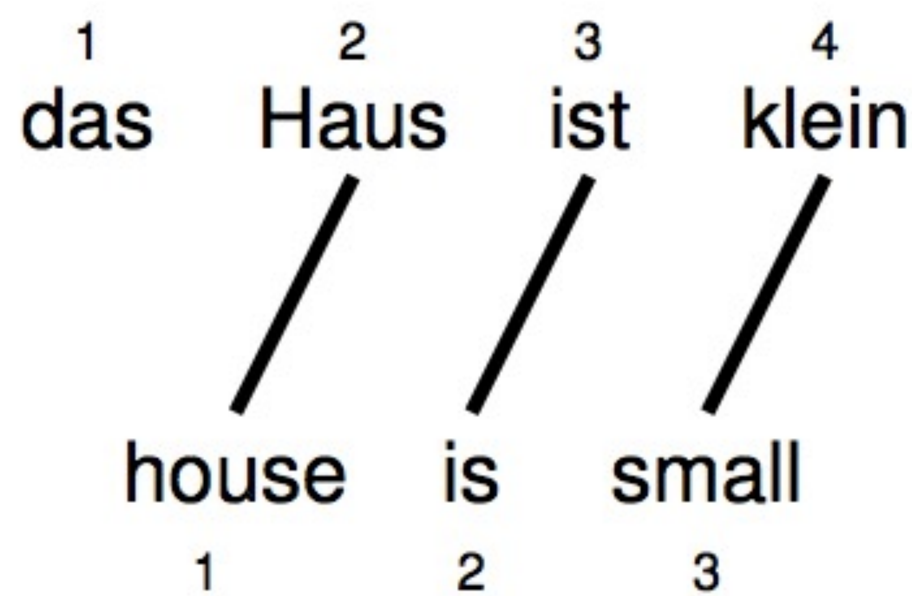
- Words may be reordered during translation.



$$\mathbf{a} = (3, 4, 2, 1)^{\top}$$

Word Dropping

- A source word may not be translated at all



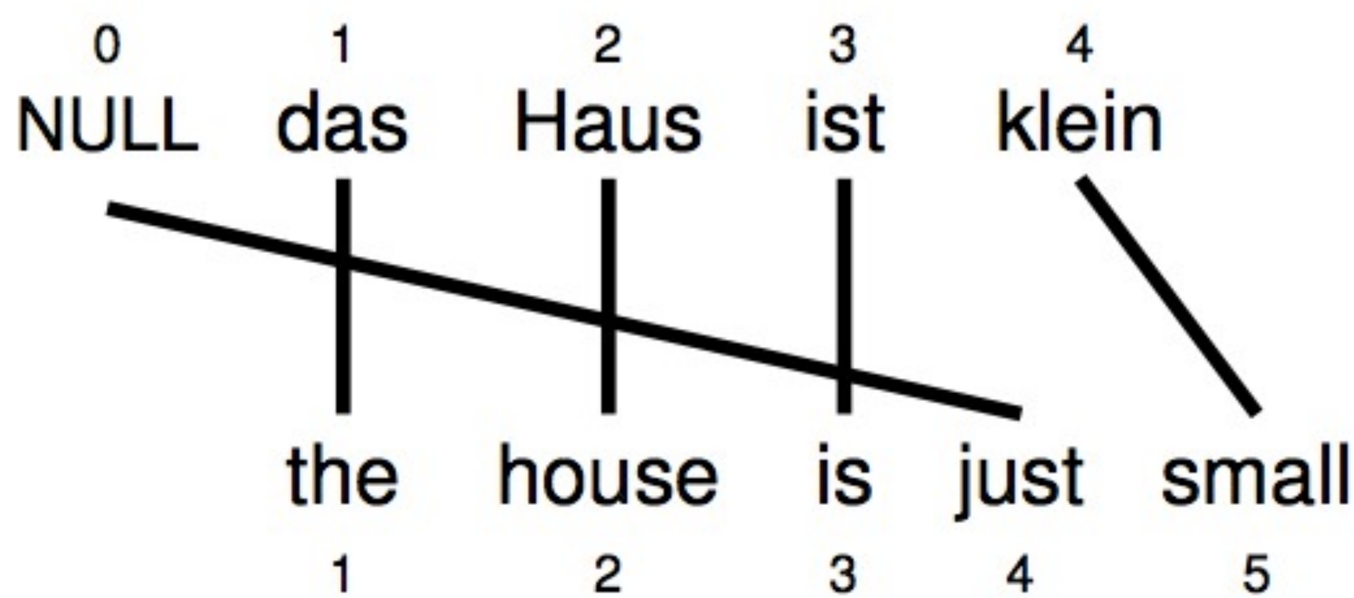
$$\mathbf{a} = (2, 3, 4)^\top$$

Word Insertion

- Words may be inserted during translation

English just does not have an equivalent

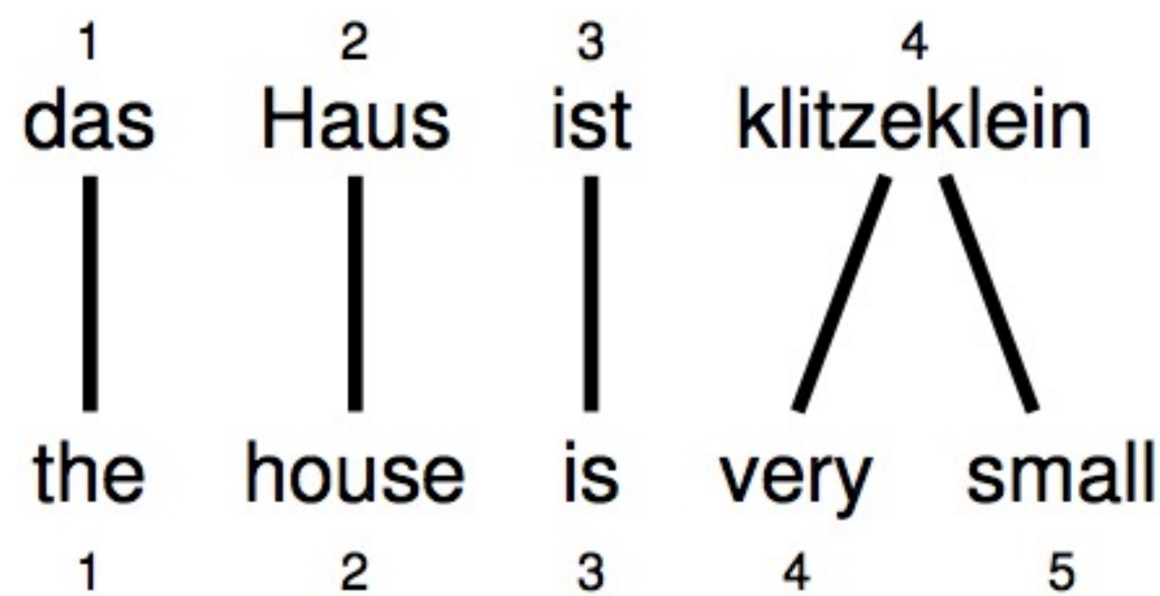
But it must be explained - we typically assume every source sentence contains a NULL token



$$\mathbf{a} = (1, 2, 3, 0, 4)^\top$$

One-to-many Translation

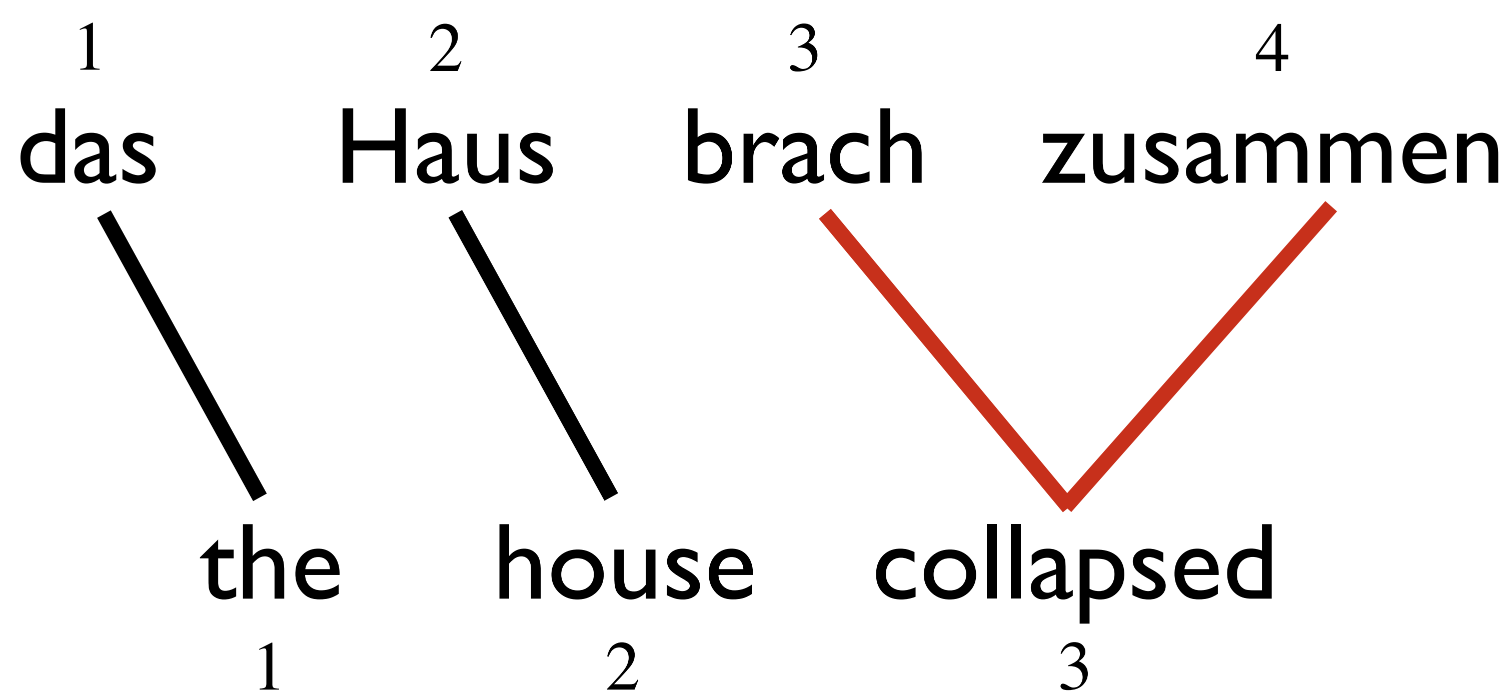
- A source word may translate into **more than one** target word



$$\mathbf{a} = (1, 2, 3, 4, 4)^\top$$

Many-to-one Translation

- **More than one source word** may **not** translate as a unit in lexical translation



$\mathbf{a} = ???$

[IBM Model 1 can't do this]

IBM Model 1

- Simplest possible lexical translation model
- Additional assumptions
  - The m alignment decisions are independent
  - The alignment distribution for each a_i is uniform over all source words and NULL

$$\begin{aligned}
&\text{for each } i \in [1, 2, \ldots, m] \\
&\qquad a_i \sim \mathrm{Uniform}(0, 1, 2, \ldots, n) \\
&\qquad e_i \sim \mathrm{Categorical}(\boldsymbol{\theta}_{f_{a_i}})
\end{aligned}$$

IBM Model 1

$$\boxed{\text{for each } i \in [1, 2, \ldots, m]}$$

$$a_i \sim \text{Uniform}(0, 1, 2, \ldots, n)$$

$$e_i \sim \text{Categorical}(\boldsymbol{\theta}_{f_{a_i}})$$

$$p(\mathbf{e}, \mathbf{a} \mid \mathbf{f}, m) = \prod_{i=1}^{m}$$

IBM Model 1

$$\text{for each } i \in [1, 2, \ldots, m]$$
$$\boxed{a_i \sim \text{Uniform}(0, 1, 2, \ldots, n)}$$
$$e_i \sim \text{Categorical}(\boldsymbol{\theta}_{f_{a_i}})$$

$$p(\mathbf{e}, \mathbf{a} \mid \mathbf{f}, m) = \prod_{i=1}^{m} \frac{1}{1+n}$$

IBM Model 1

$$\text{for each } i \in [1, 2, \ldots, m]$$

$$a_i \sim \text{Uniform}(0, 1, 2, \ldots, n)$$

$$\boxed{e_i \sim \text{Categorical}(\boldsymbol{\theta}_{f_{a_i}})}$$

$$p(\mathbf{e}, \mathbf{a} \mid \mathbf{f}, m) = \prod_{i=1}^{m} \frac{1}{1+n}\, p(e_i \mid f_{a_i})$$

Example

0	1	2	3	4
NULL	das	Haus	ist	klein

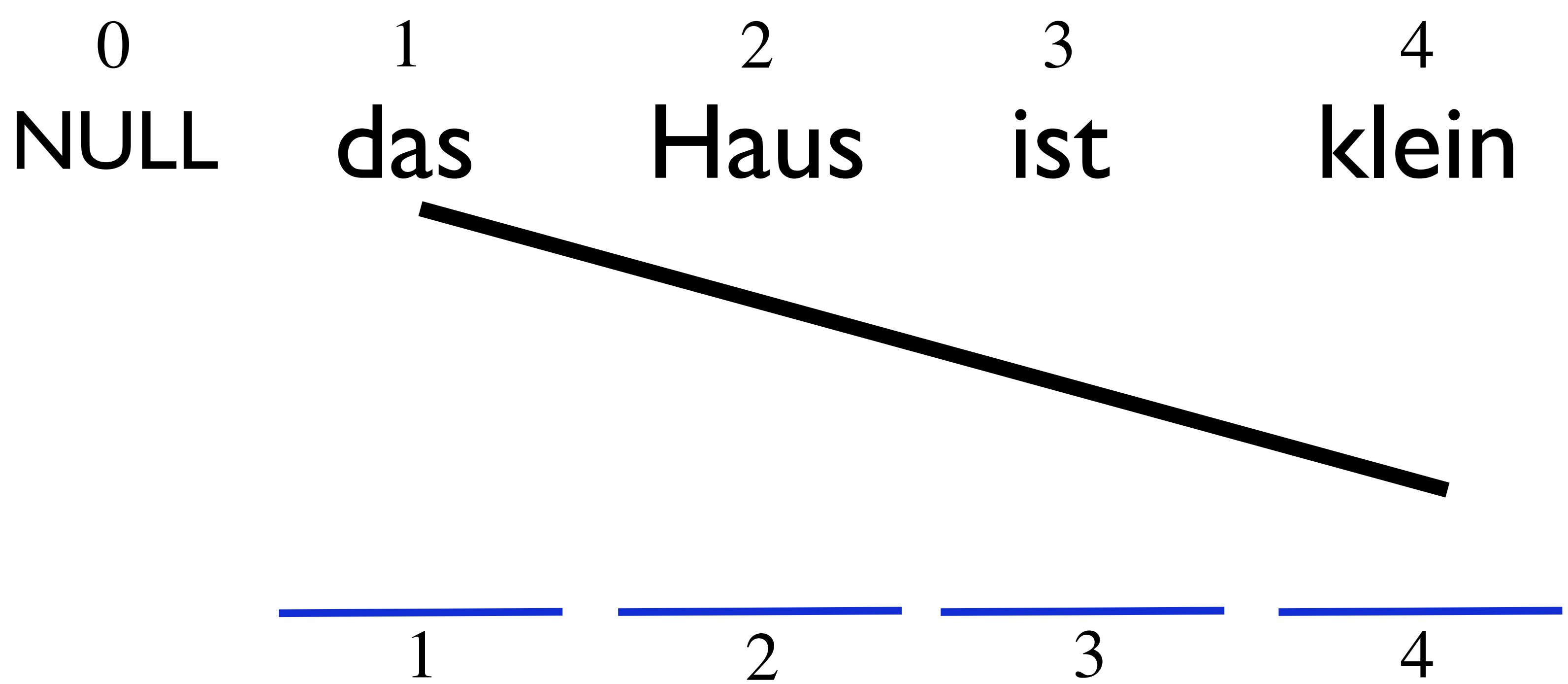
______ ______ ______ ______

1 2 3 4

$p(\mathbf{e} \mid \mathbf{f})$: Assume a foreign sentence and target length.

$= p(\mathbf{a} \mid \mathbf{f}) \; p(\mathbf{e} \mid \mathbf{a}, \mathbf{f})$

Example



p(**e** | **f**): Assume a foreign sentence and target length.

= p(**a** | **f**) p(**e** | **a**, **f**)

Example

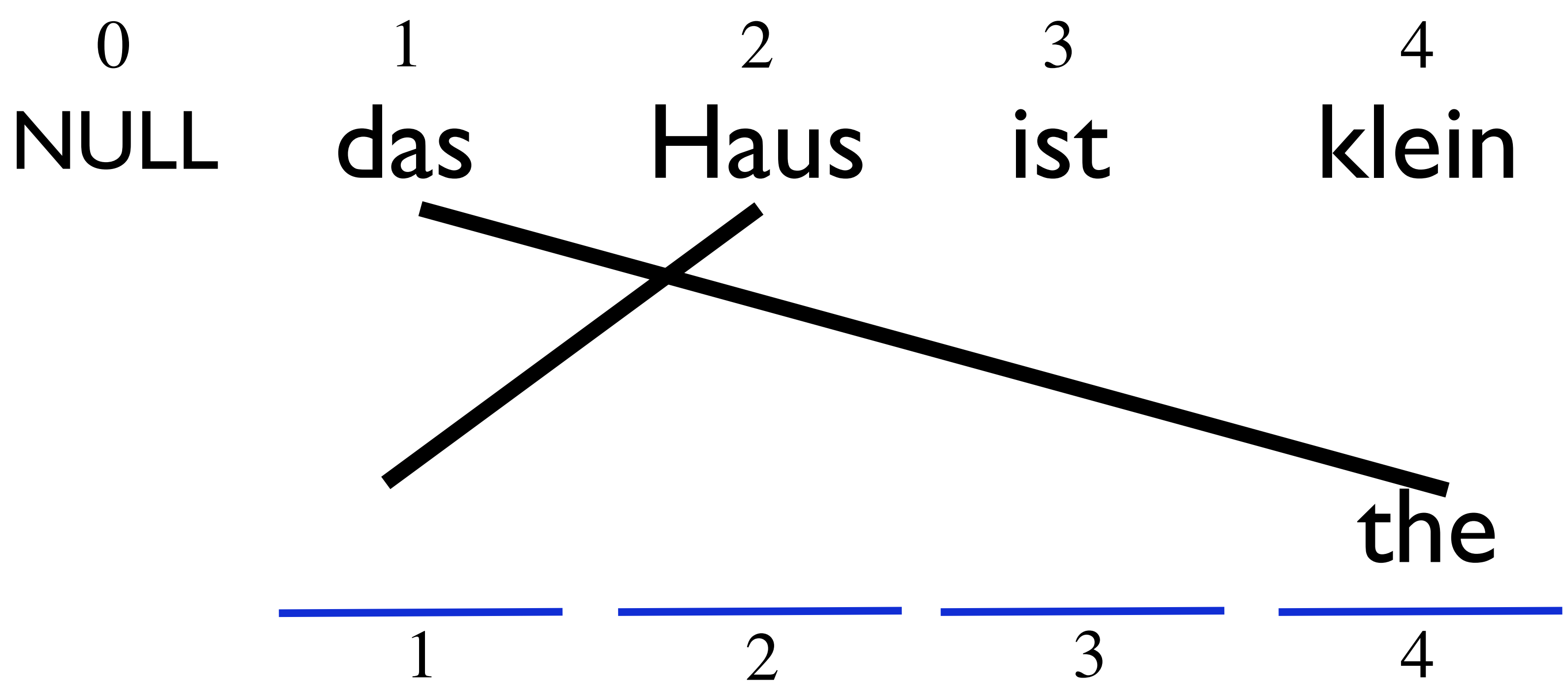
0	1	2	3	4
NULL	das	Haus	ist	klein

			the
1	2	3	4

p(**e** | **f**): Assume a foreign sentence and target length.

= p(**a** | **f**) p(**e** | **a**, **f**)

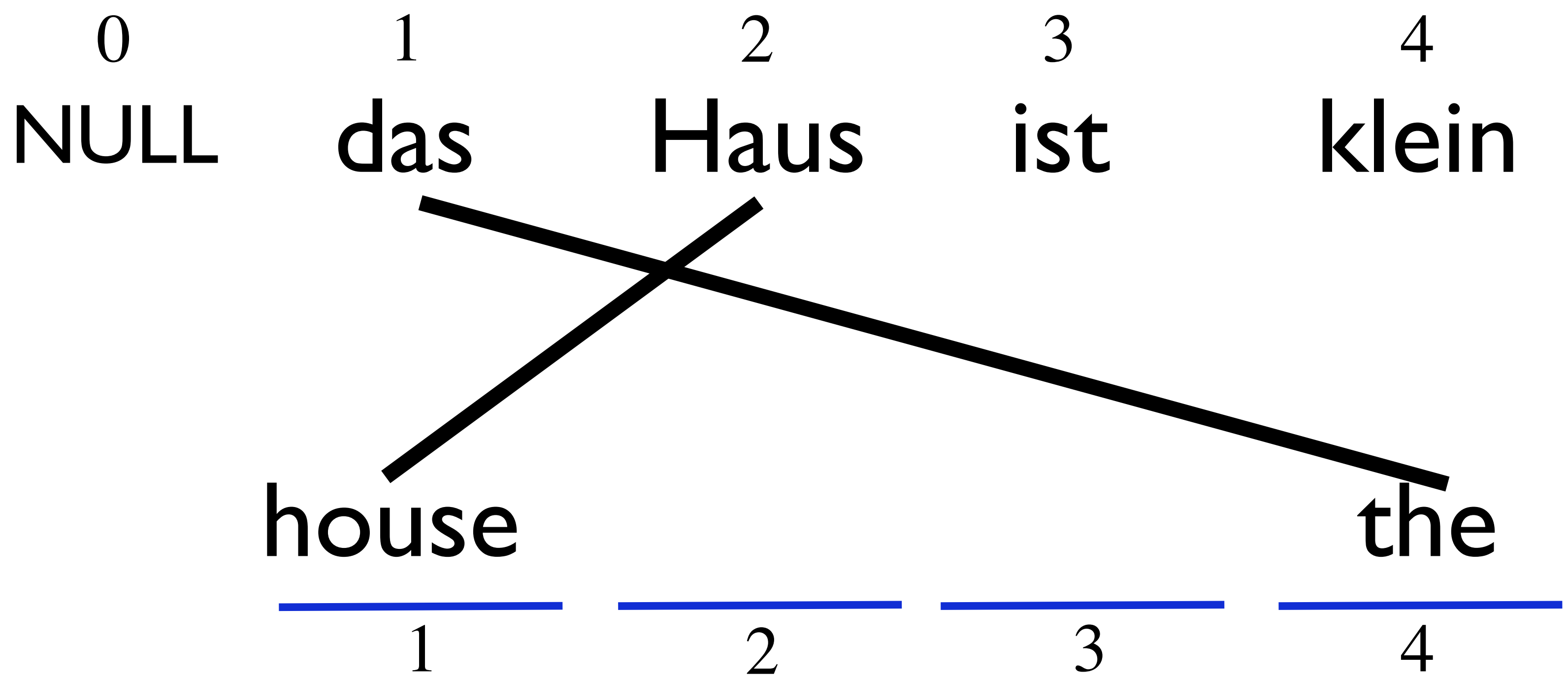
Example



p(**e** | **f**): Assume a foreign sentence and target length.

= p(**a** | **f**) p(**e** | **a**, **f**)

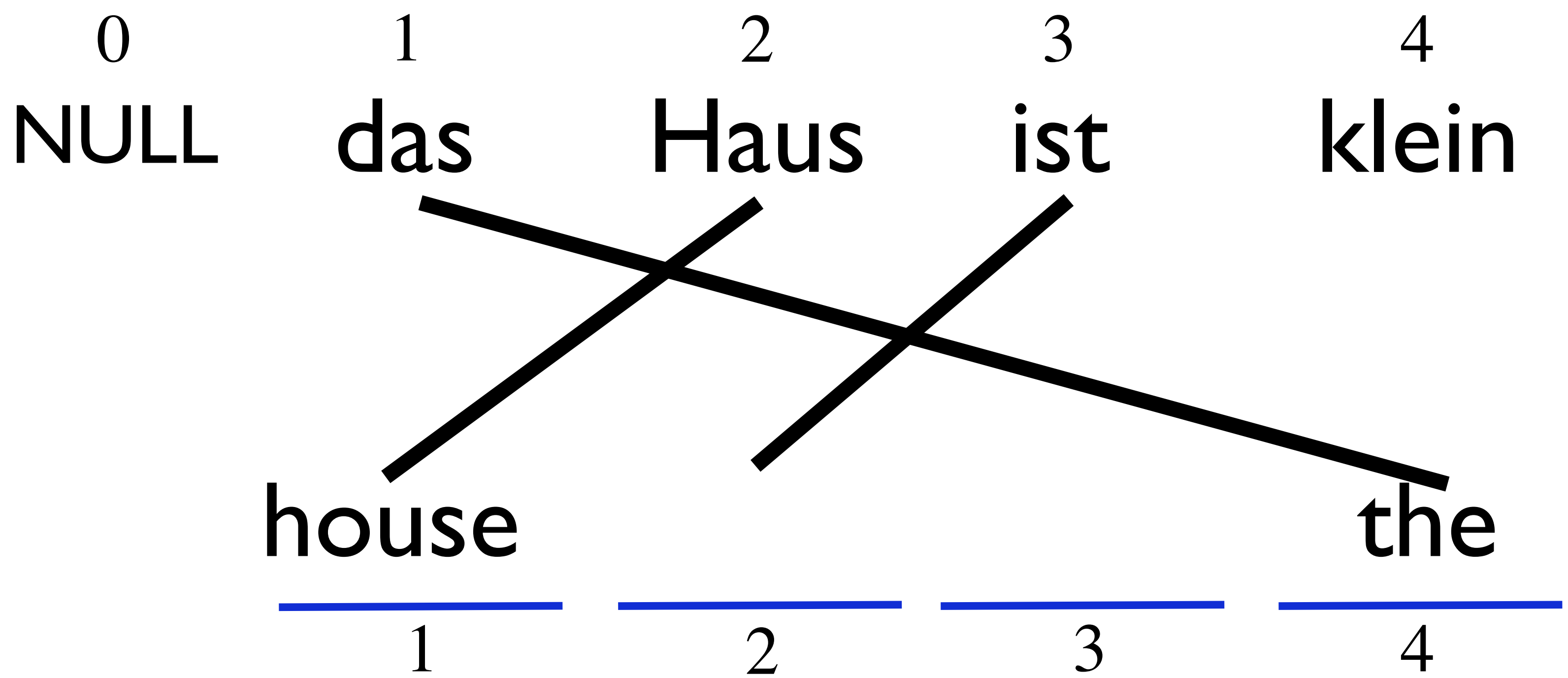
Example



$p(\mathbf{e} \mid \mathbf{f})$: Assume a foreign sentence and target length.

$= p(\mathbf{a} \mid \mathbf{f}) \;\; p(\mathbf{e} \mid \mathbf{a}, \mathbf{f})$

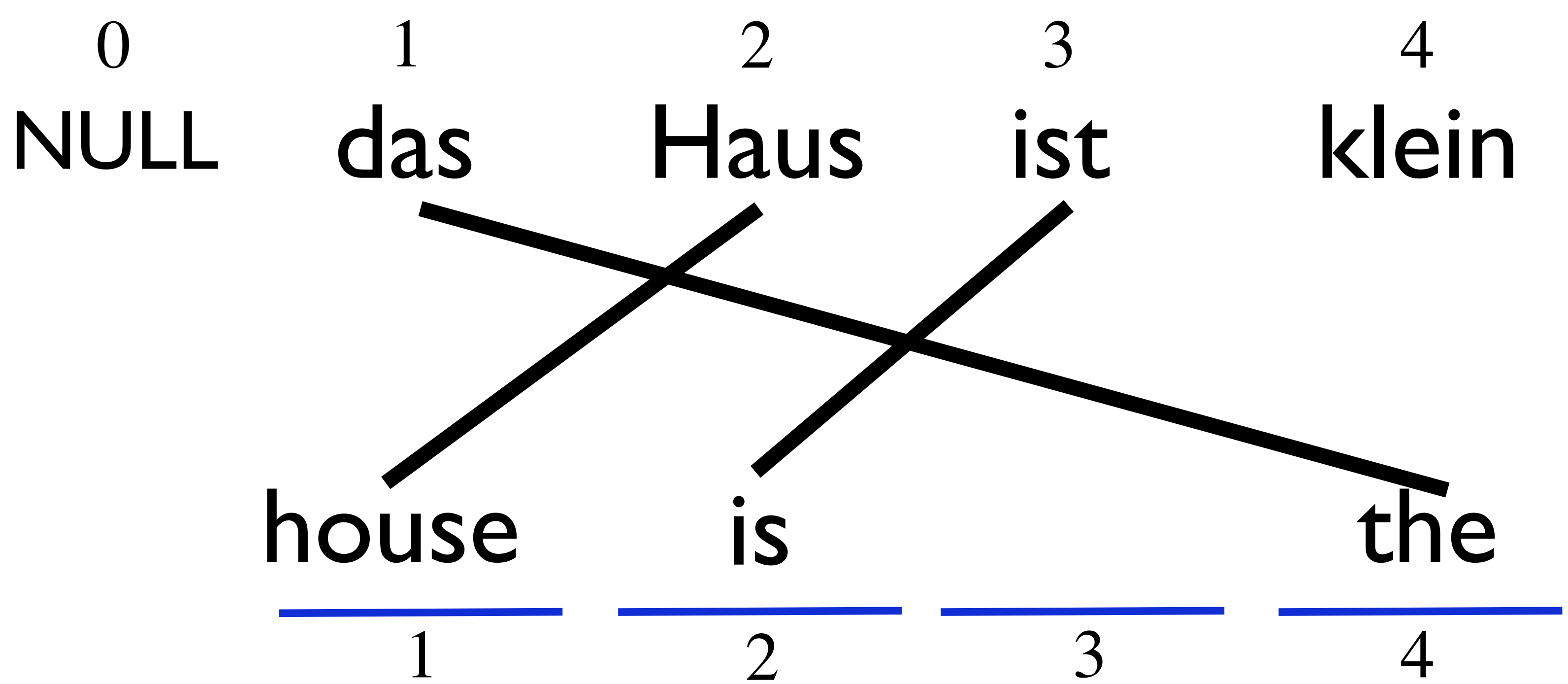
Example



p(**e** | **f**): Assume a foreign sentence and target length.

= p(**a** | **f**) p(**e** | **a**, **f**)

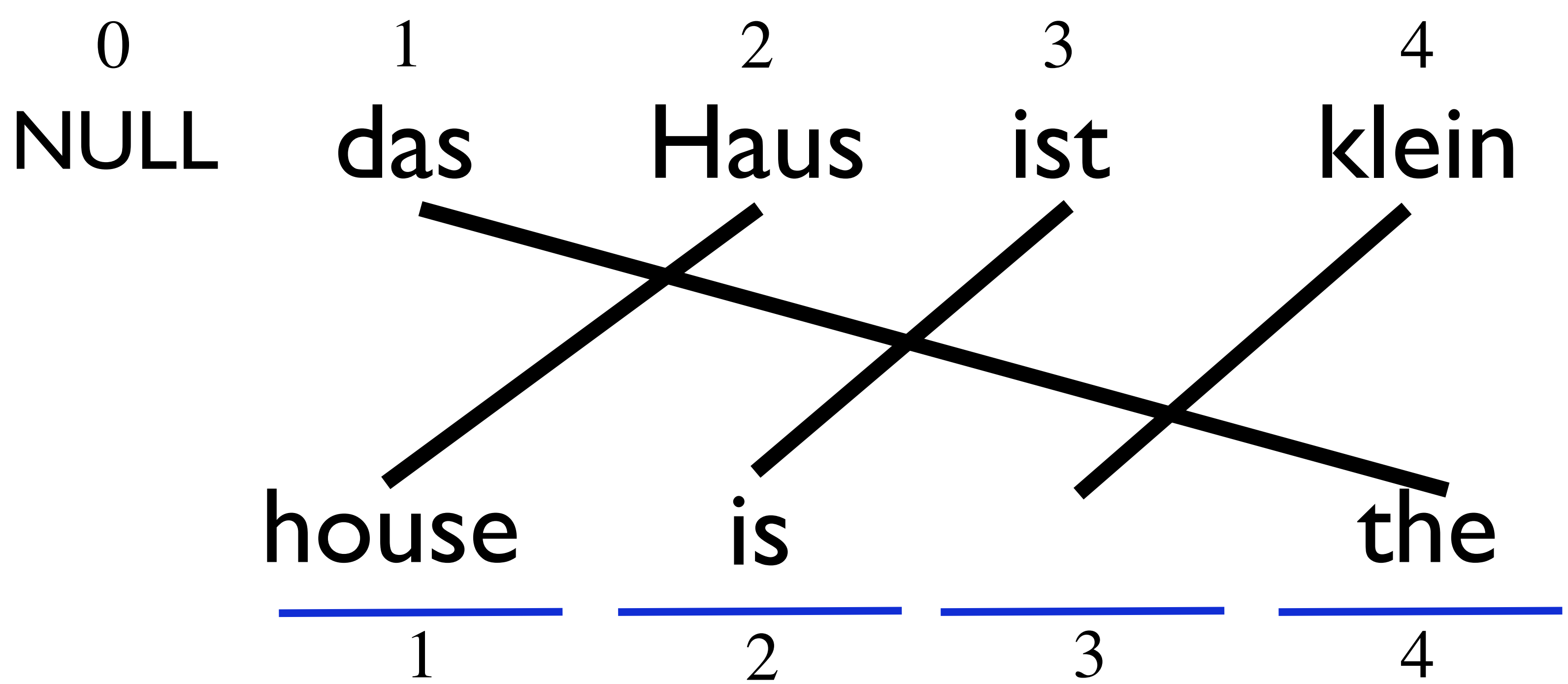
Example



$p(\mathbf{e} \mid \mathbf{f})$: Assume a foreign sentence and target length.

$= p(\mathbf{a} \mid \mathbf{f}) \;\; p(\mathbf{e} \mid \mathbf{a}, \mathbf{f})$

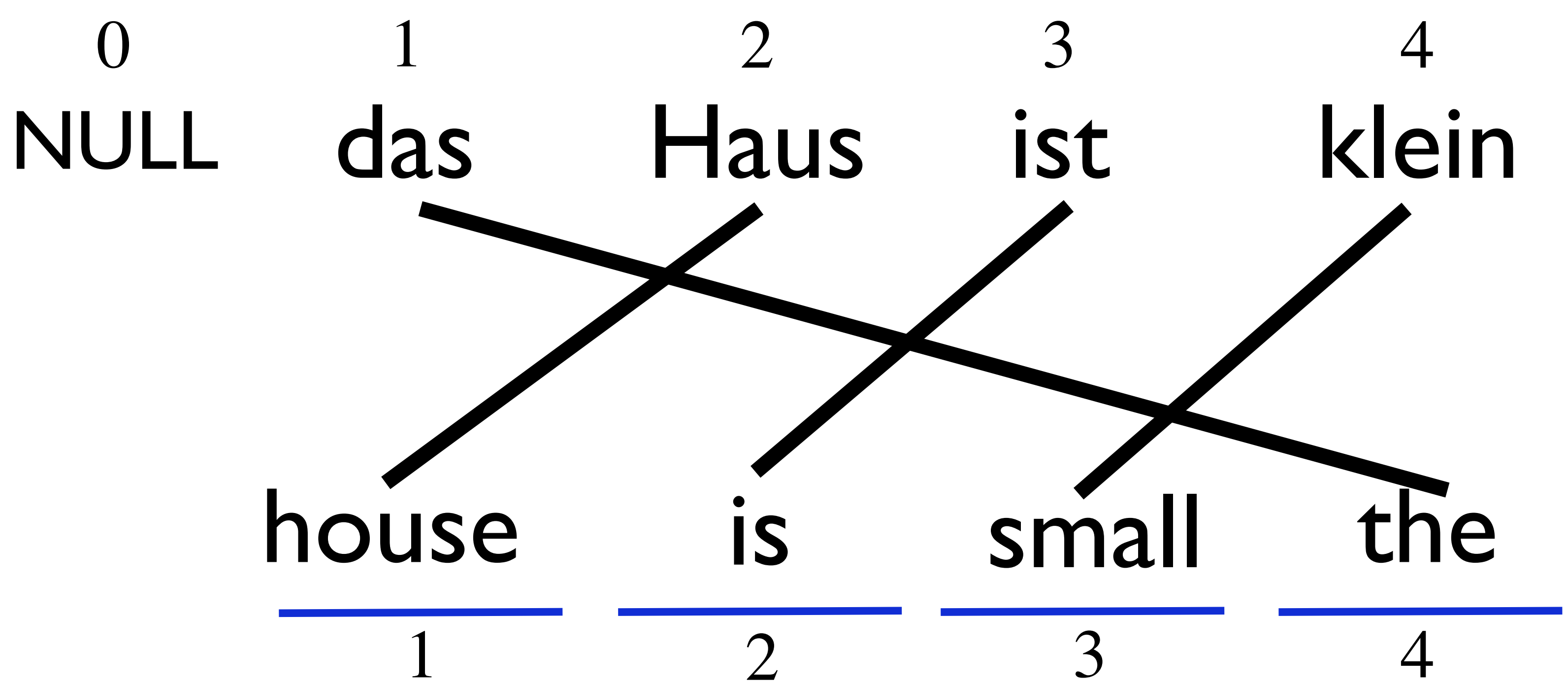
Example



$p(\mathbf{e} \mid \mathbf{f})$: Assume a foreign sentence and target length.

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

Example



p(**e** | **f**): Assume a foreign sentence and target length.

= p(**a** | **f**) p(**e** | **a**, **f**)

IBM Model 1: Inference and learning

- Alignment inference:
  Given lexical translation probabilities,
  infer posterior or Viterbi alignment

$$\arg\max_{\mathbf{a}} p(\mathbf{a} \mid \mathbf{e}, \mathbf{f}, \theta)$$

- Translation: incorporate into noisy channel
  (this model isn't good at this)

$$\arg\max_{e} p(\mathbf{e} \mid \mathbf{f}, \theta)\ p(\mathbf{e})$$

- How do we learn translation parameters?
  EM Algorithm (Thursday)

$$\arg\max_{\theta} p(\mathbf{e} \mid \mathbf{f}, \theta)$$

- Chicken and egg problem:
  If we knew alignments, translation
  parameters would be trivial (just counting)