Machine Translation 1: Word alignments

Computational Linguistics

Alexander Koller

18 December 2018

slides contain material from mt-class.org

Google Translate

EL PAÍS | PORTADA | INTERNACIONAL | POI

DEPORTES

FÚTBOL | BALONCESTO | TENIS | CICLISMO | FÓRMULA 1 | MOTOS | GOLF | OTROS

ESTÁ PASANDO | Liga: 37ª jornada | Giro Italia | F-1: GP de España | Tenis: Masters de

MUNDIAL 2014 | ALEMANIA »

Löw deja fuera del Mundial a Ter Stegen y cuenta con Khedira

- El técnico de Alemania anuncia la lista de 30 preseleccionados para Brasil, en la que no cuenta con el futuro portero del Barcelona y sí con el medio del Madrid

EL PAÍS | Madrid | 8 MAY 2014 - 14:04 CET 4

Archivado en: Selección Fútbol Alemania Joachim Löw Selecciones deportivas Fútbol Deportes

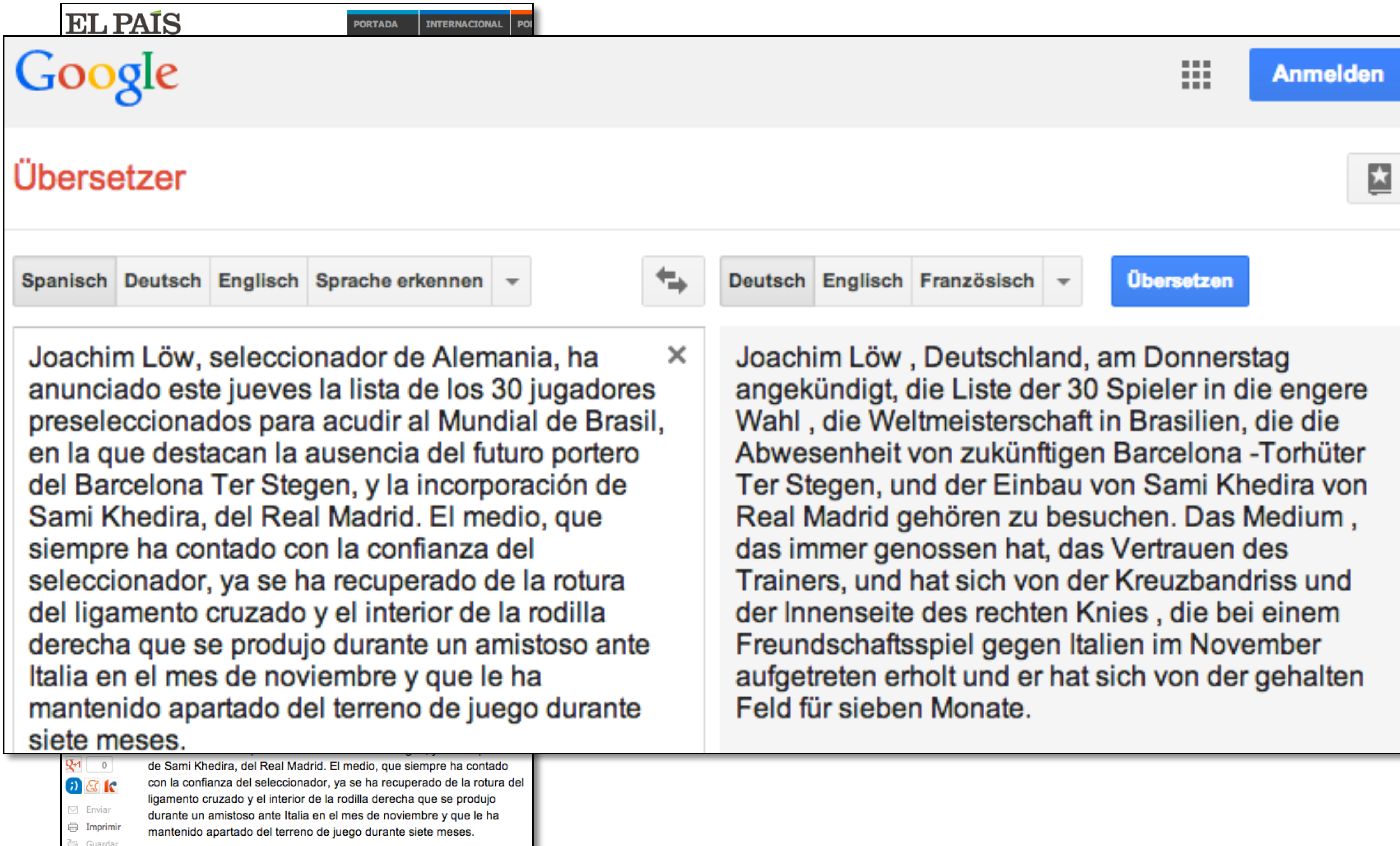
Khedira se duele tras su lesión / GIUSEPPE CACACE (AFP)

13 | 133 | 0 | 0

Enviar
Imprimir
Guardar

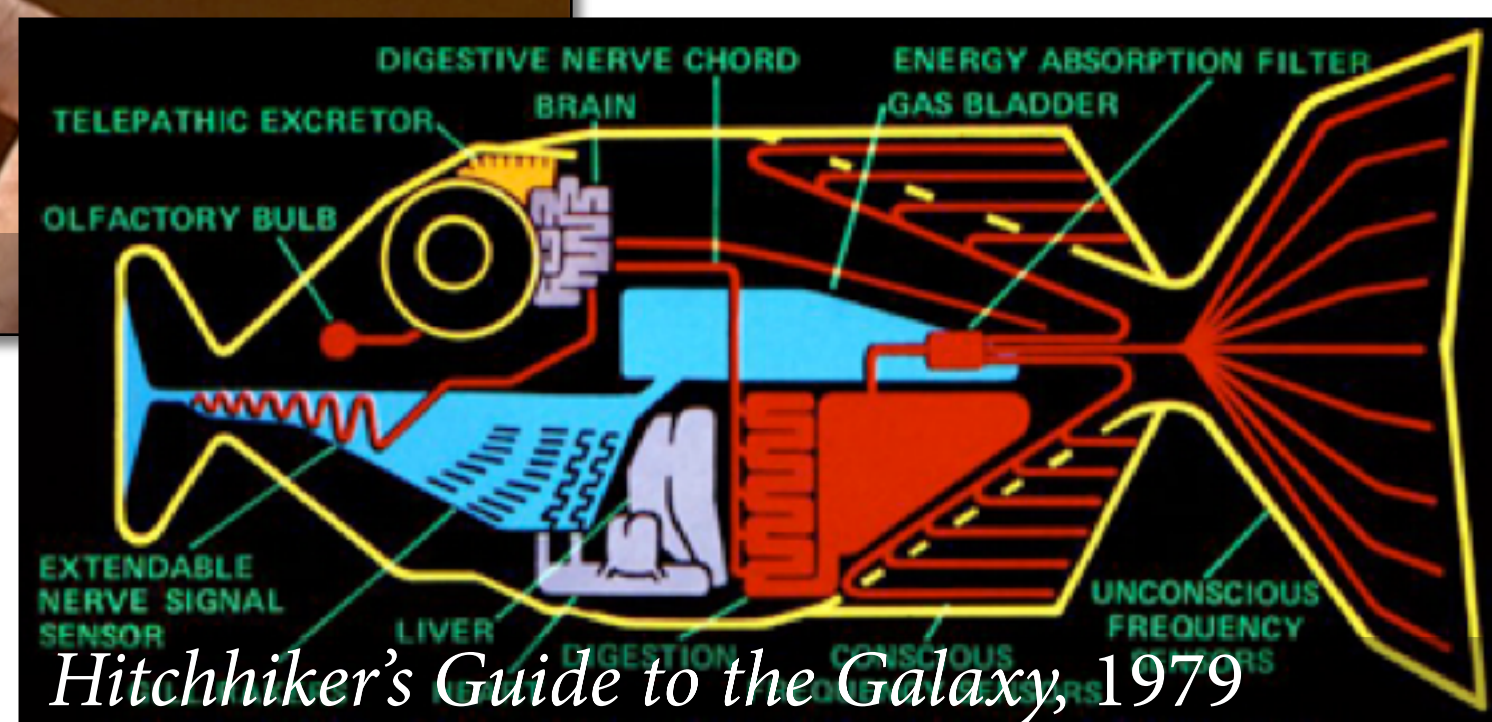
Joachim Löw, seleccionador de Alemania, ha anunciado este jueves la lista de los 30 jugadores preseleccionados para acudir al Mundial de Brasil, que comenzará el próximo 12 de junio, en la que destacan la ausencia del futuro portero del Barcelona Ter Stegen, y la incorporación de Sami Khedira, del Real Madrid. El medio, que siempre ha contado con la confianza del seleccionador, ya se ha recuperado de la rotura del ligamento cruzado y el interior de la rodilla derecha que se produjo durante un amistoso ante Italia en el mes de noviembre y que le ha mantenido apartado del terreno de juego durante siete meses.

Google Translate



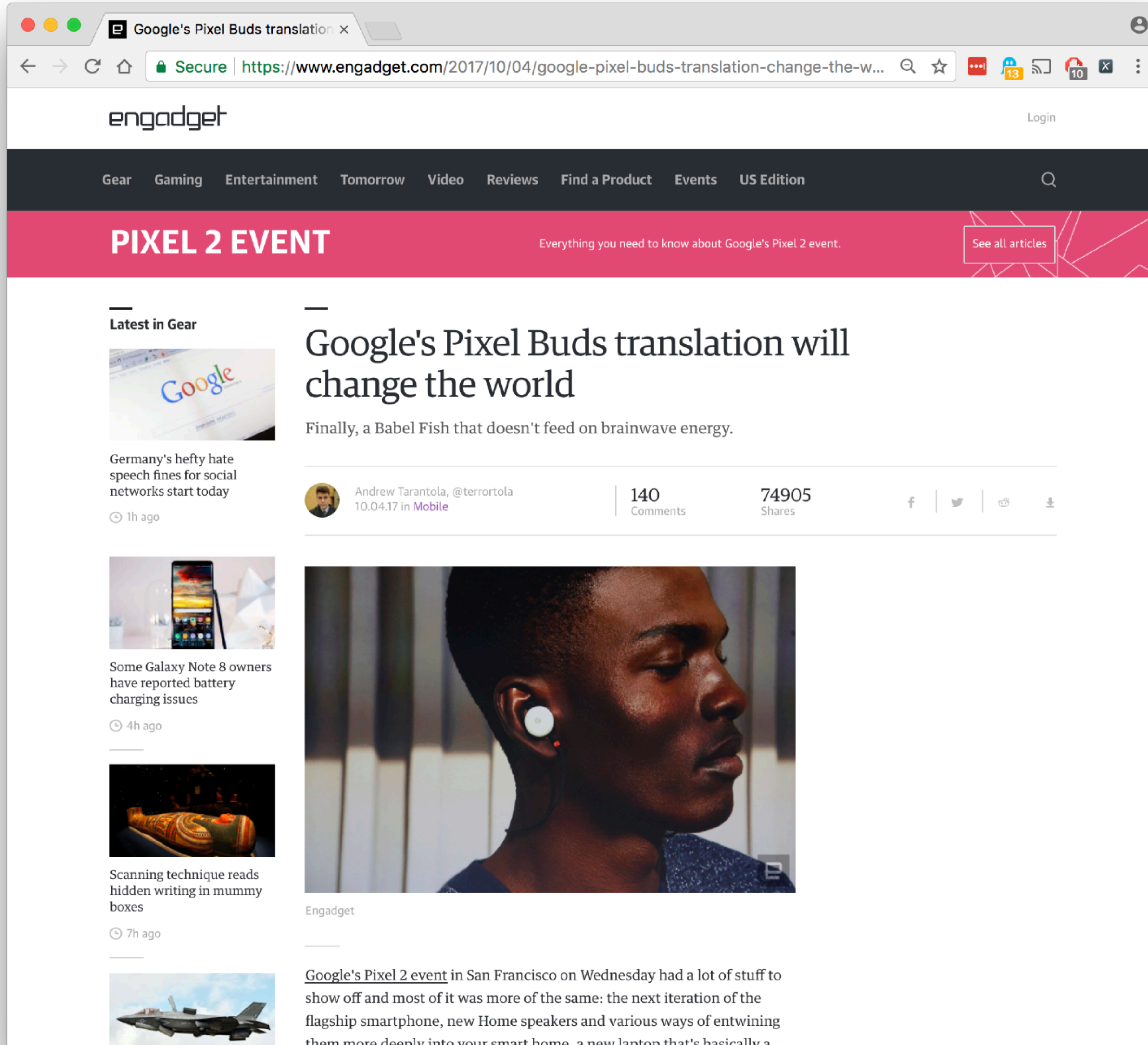
Automatic Translation

Star Trek, 1966



Hitchhiker's Guide to the Galaxy, 1979

Google Pixel Buds, 2017



Google's Pixel Buds translatio ×

Secure https://www.engadget.com/2017/10/04/google-pixel-buds-translation-change-the-w...

engadget

Login

Gear Gaming Entertainment Tomorrow Video Reviews Find a Product Events US Edition

PIXEL 2 EVENT

Everything you need to know about Google's Pixel 2 event.

See all articles

Latest in Gear

Germany's hefty hate speech fines for social networks start today

1h ago

Some Galaxy Note 8 owners have reported battery charging issues

4h ago

Scanning technique reads hidden writing in mummy boxes

7h ago

Google's Pixel Buds translation will change the world

Finally, a Babel Fish that doesn't feed on brainwave energy.

Andrew Tarantola, @terrortola
10.04.17 in Mobile

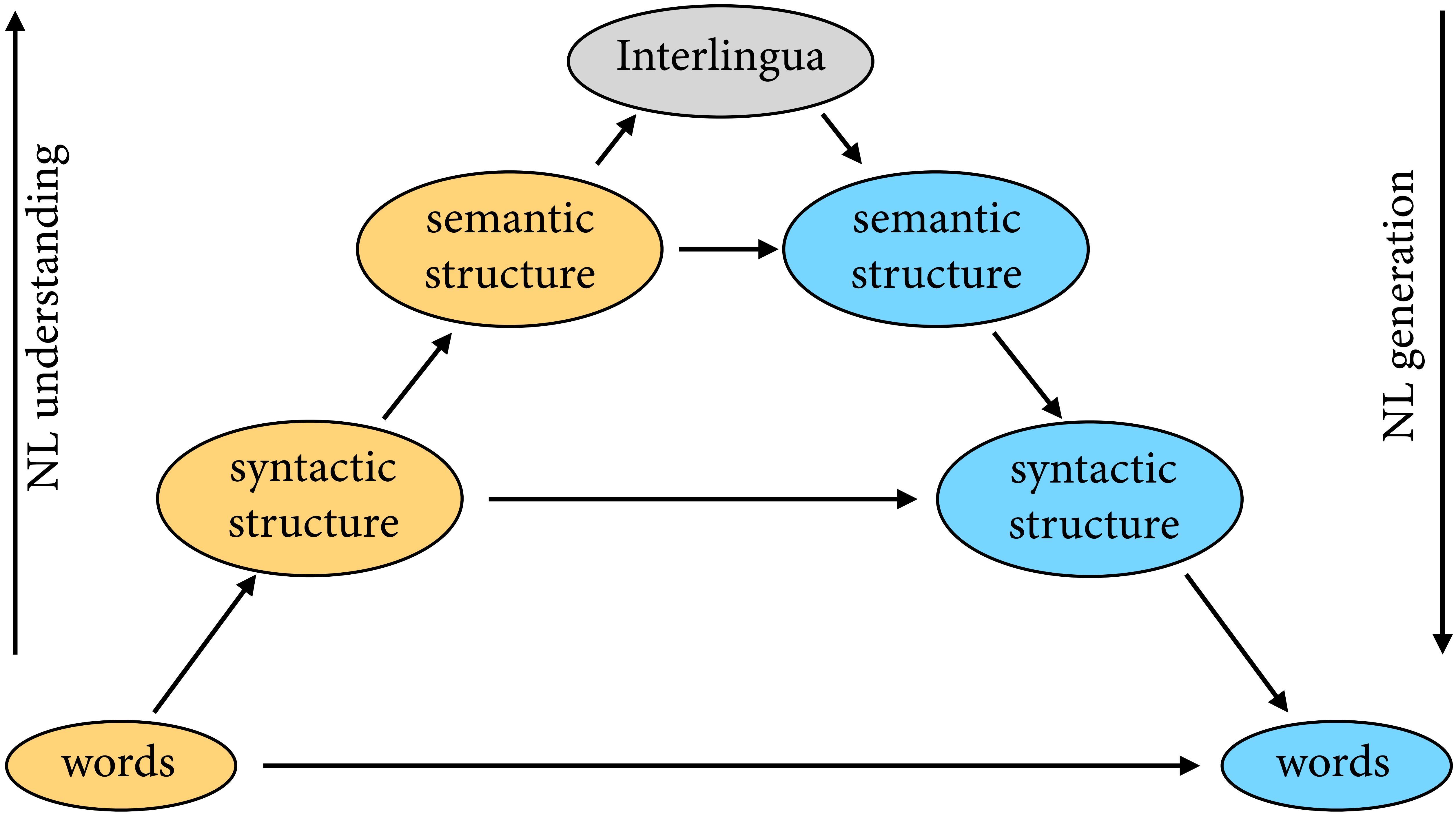
140 Comments

74905 Shares

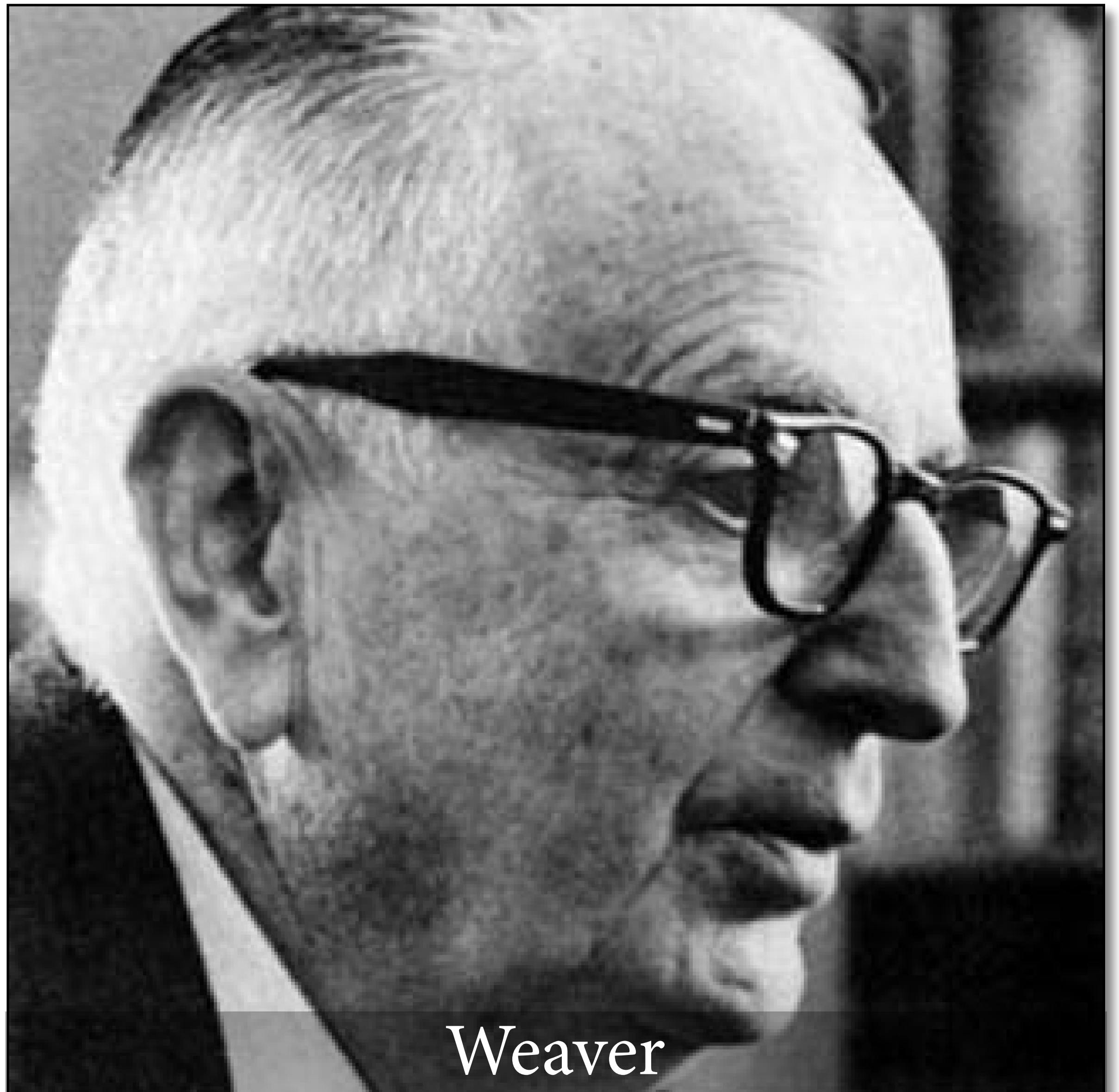
Engadget

Google's Pixel 2 event in San Francisco on Wednesday had a lot of stuff to show off and most of it was more of the same: the next iteration of the flagship smartphone, new Home speakers and various ways of entwining them more deeply into your smart home, a new laptop that's basically a

Classical view on translation



Early History

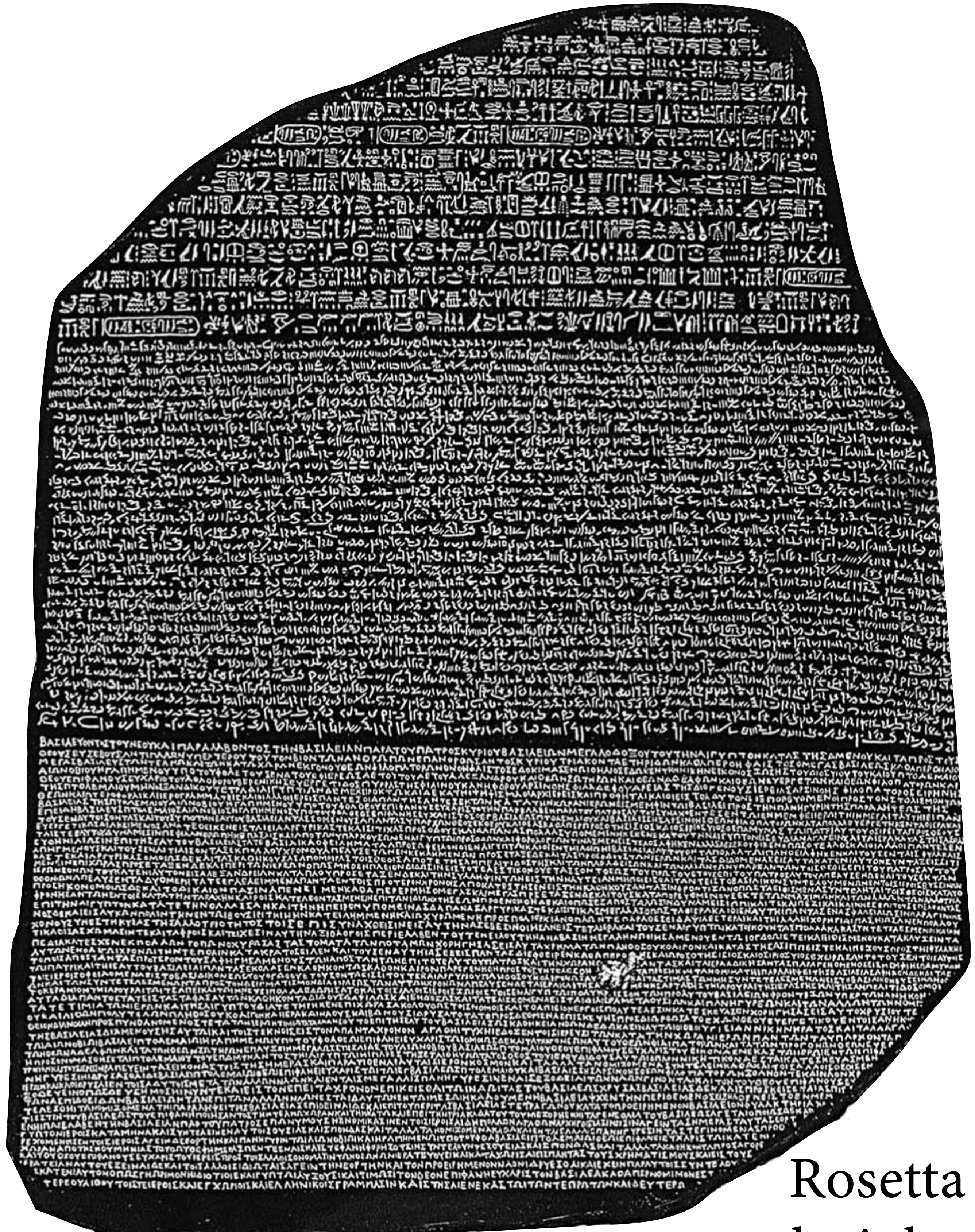


One naturally wonders if the problem of translation could conceivably be treated as a problem in cryptography.

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."

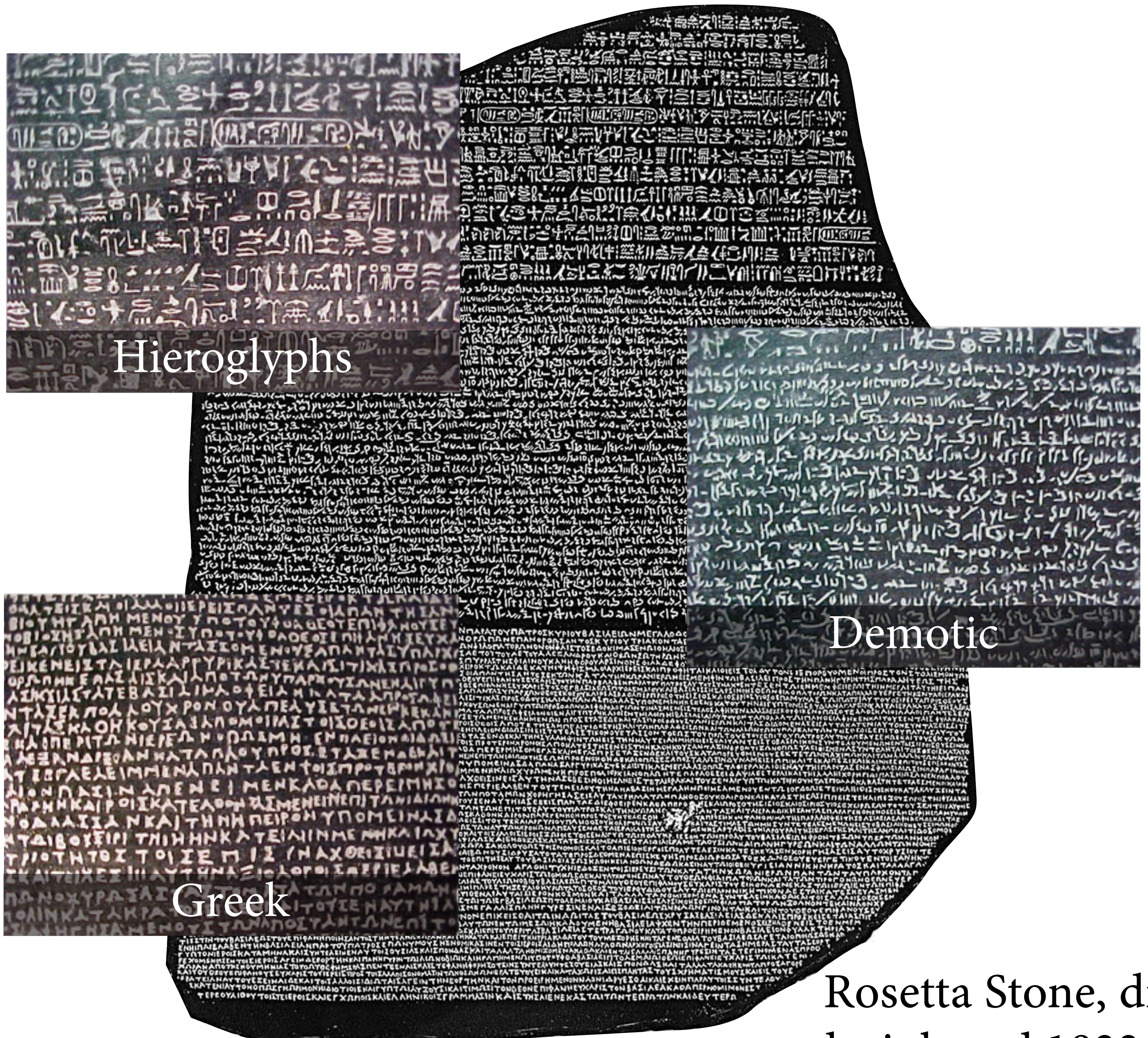
Warren Weaver to Norbert Wiener (1947)

Really Early History



Rosetta Stone, discovered 1799;
deciphered 1822 (Champollion)

Really Early History



Rosetta Stone, discovered 1799; deciphered 1822 (Champollion)

Types of MT systems

- What's it for?
  - fully automatic translation
  - support for human translators
- How does it work?
  - rule-based
  - statistical
  - neural
- Neural methods: see Language Technology II
  Here: elementary statistical methods

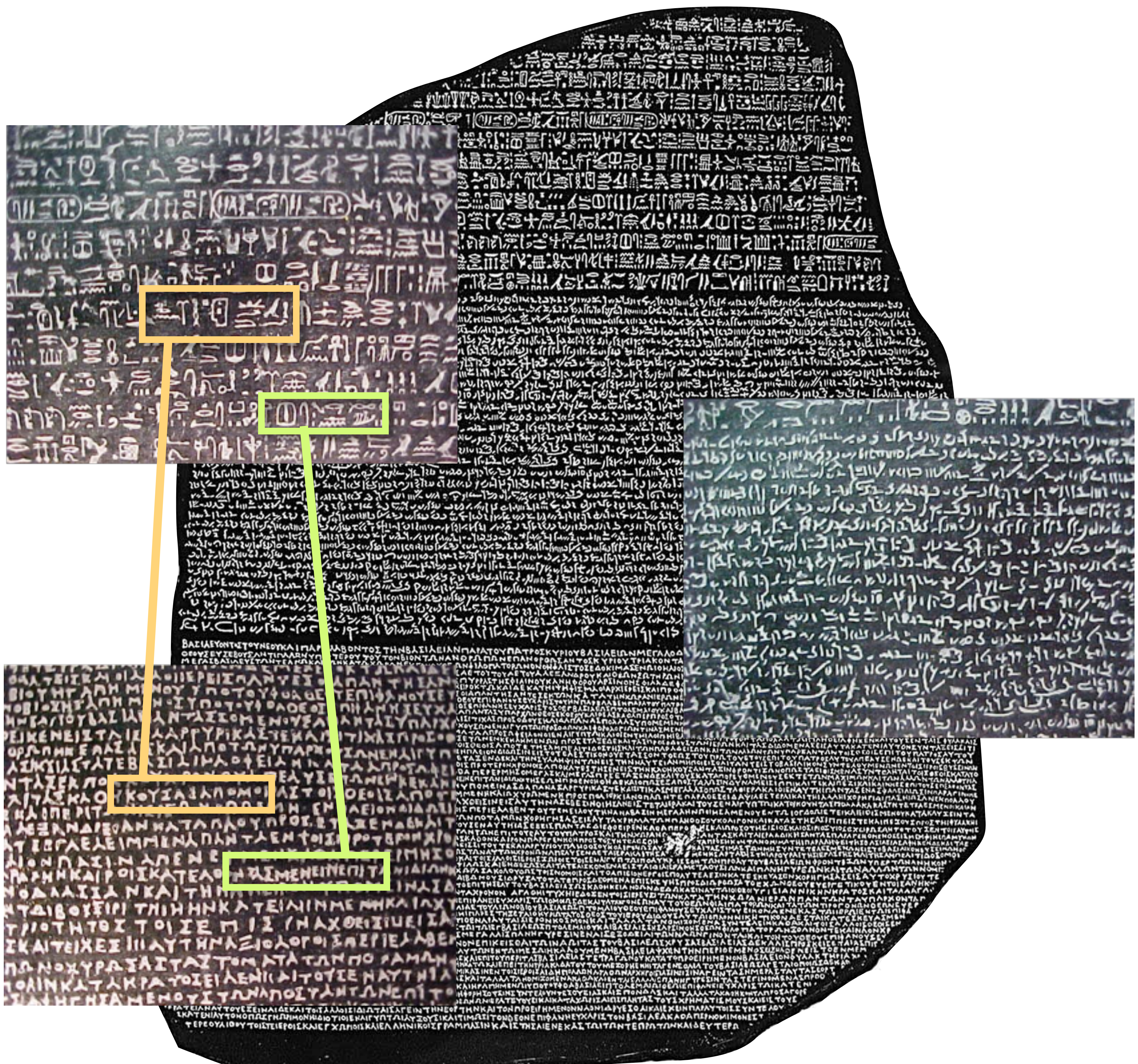
Corpora

- Learning translation models requires *parallel corpora:* text in one language with its translation in another.

- Popular parallel corpora:
  - Hansards (Canadian parliament): English/French
  - Europarl (European parliament): EU member languages
  - Literary texts with their translations (e.g. bible)

Step 1: Lexical Alignment

Step 1: Lexical Alignment



Lexical Translation

- We want to learn a model P(e | f):
  - e = "English" word (target language)
  - f = "French" word (original, foreign language)

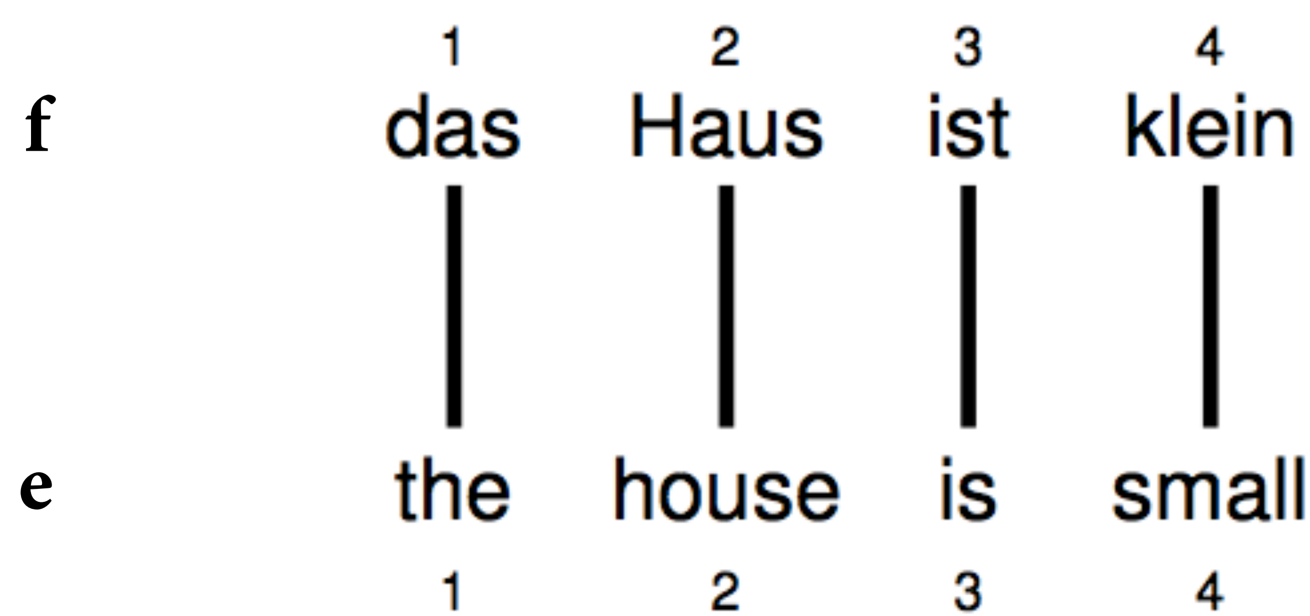
- Gives a naive translation model for P(**e** | **f**). (Boldface **e**, **f** are English, Foreign sentences.)

- Linked to idea of *word alignments*.
  - alignments often independently useful (e.g. parse tree projection)

Word alignments

Alignment

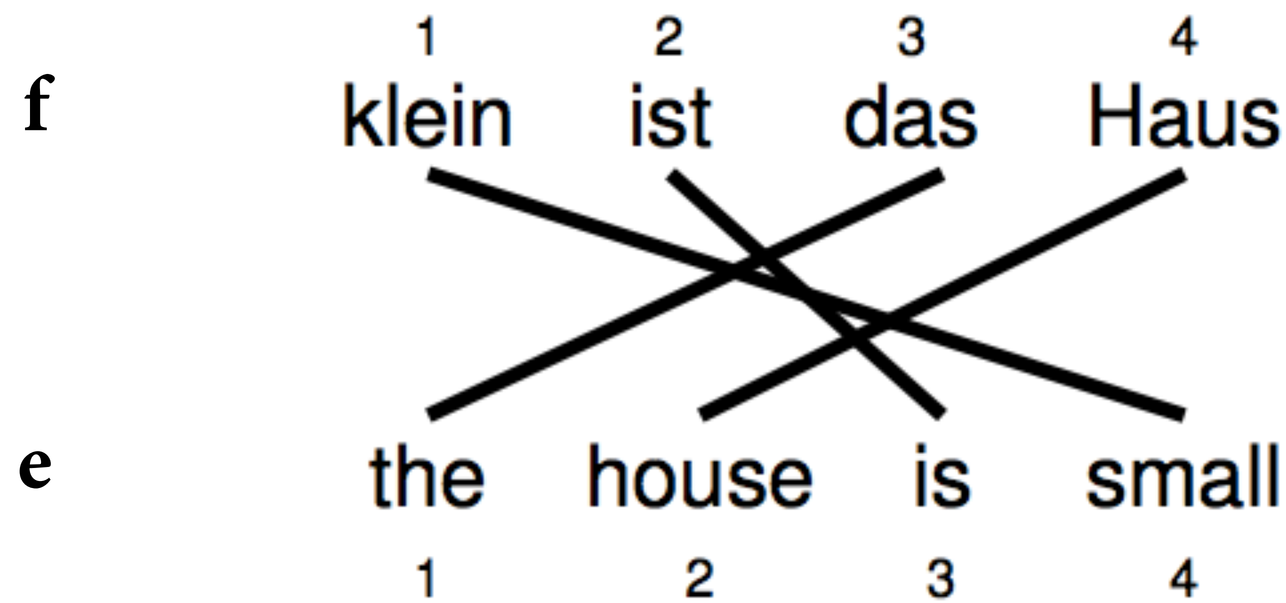
- Alignments can be visualized by drawing links between two sentences, and they are represented as vectors of positions:



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

Reordering

- Words may be reordered during translation.

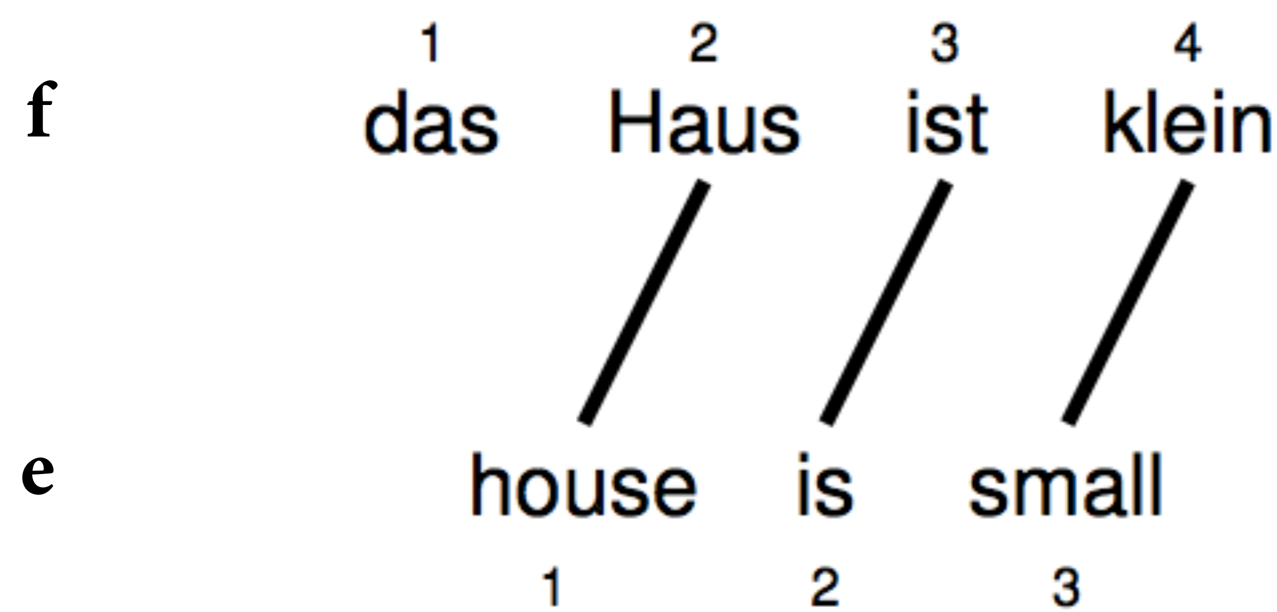


English word #1
aligned with Foreign word #3

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

Word Dropping

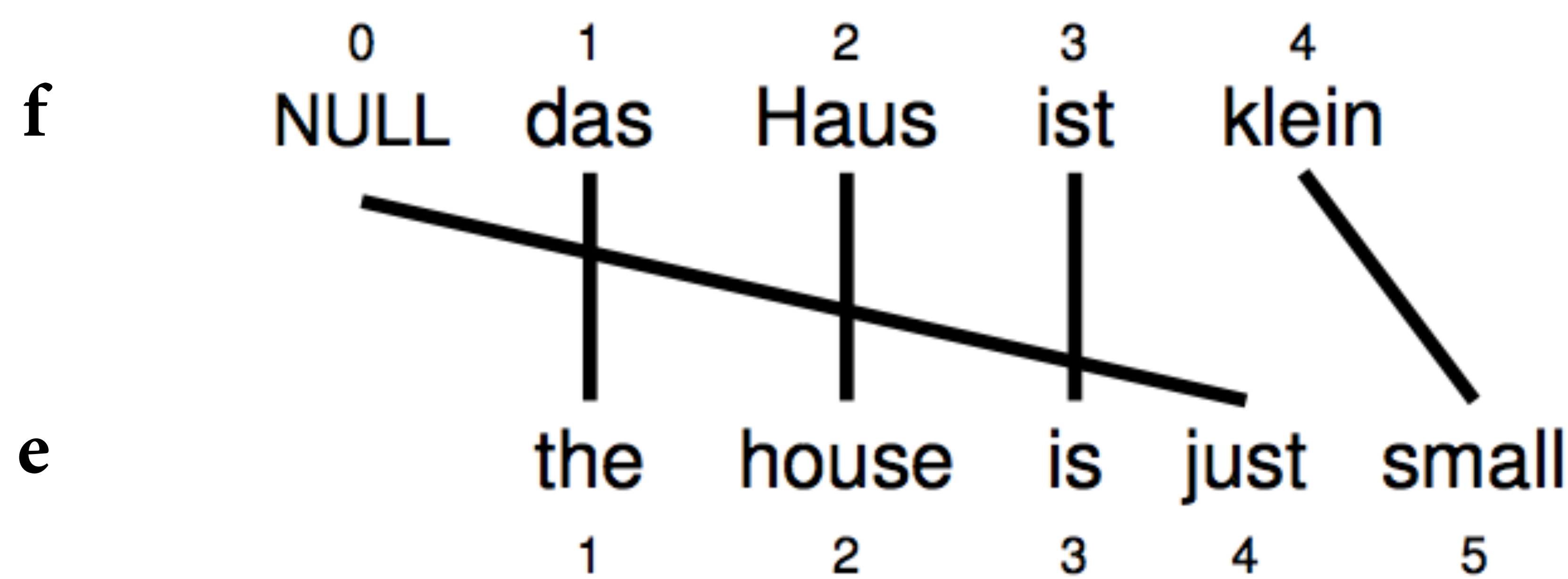
- A source word may not be translated at all
  ("1" does not occur as a_i for any English position i)



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

Word Insertion

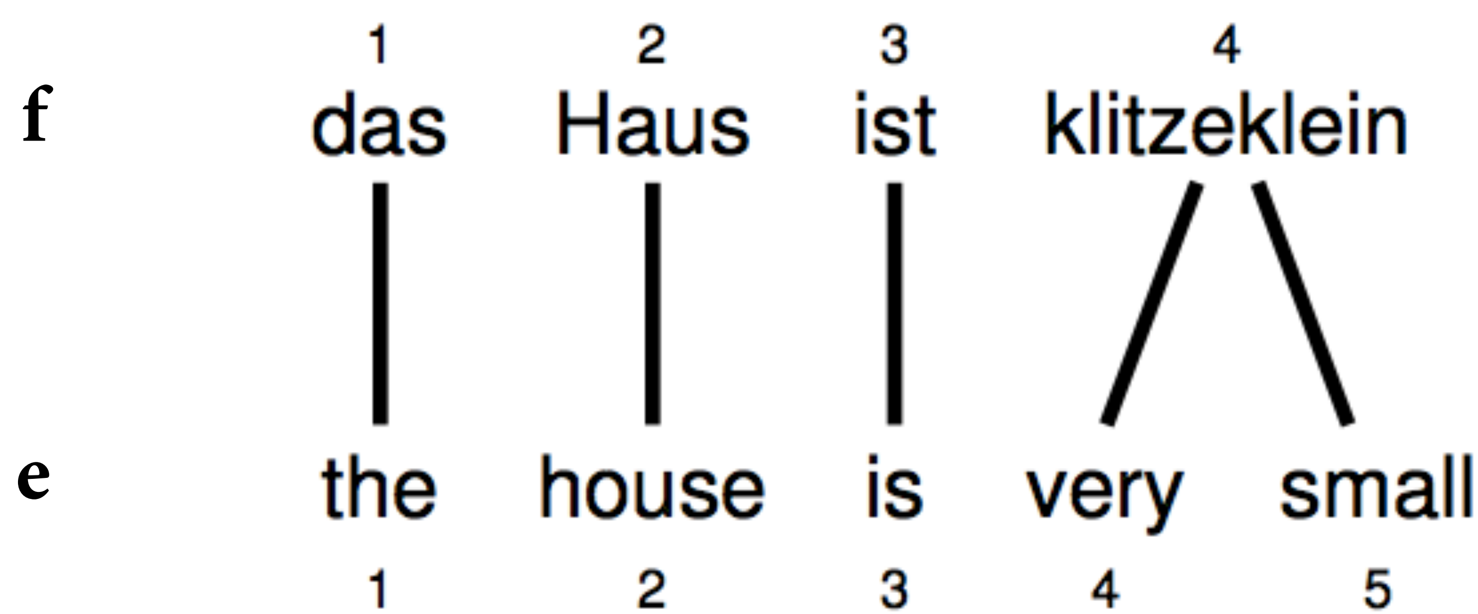
- Words may be inserted during translation
  - English "just" does not have an equivalent
  - record this by aligning with special NULL token at "position 0"



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

One-to-many Translation

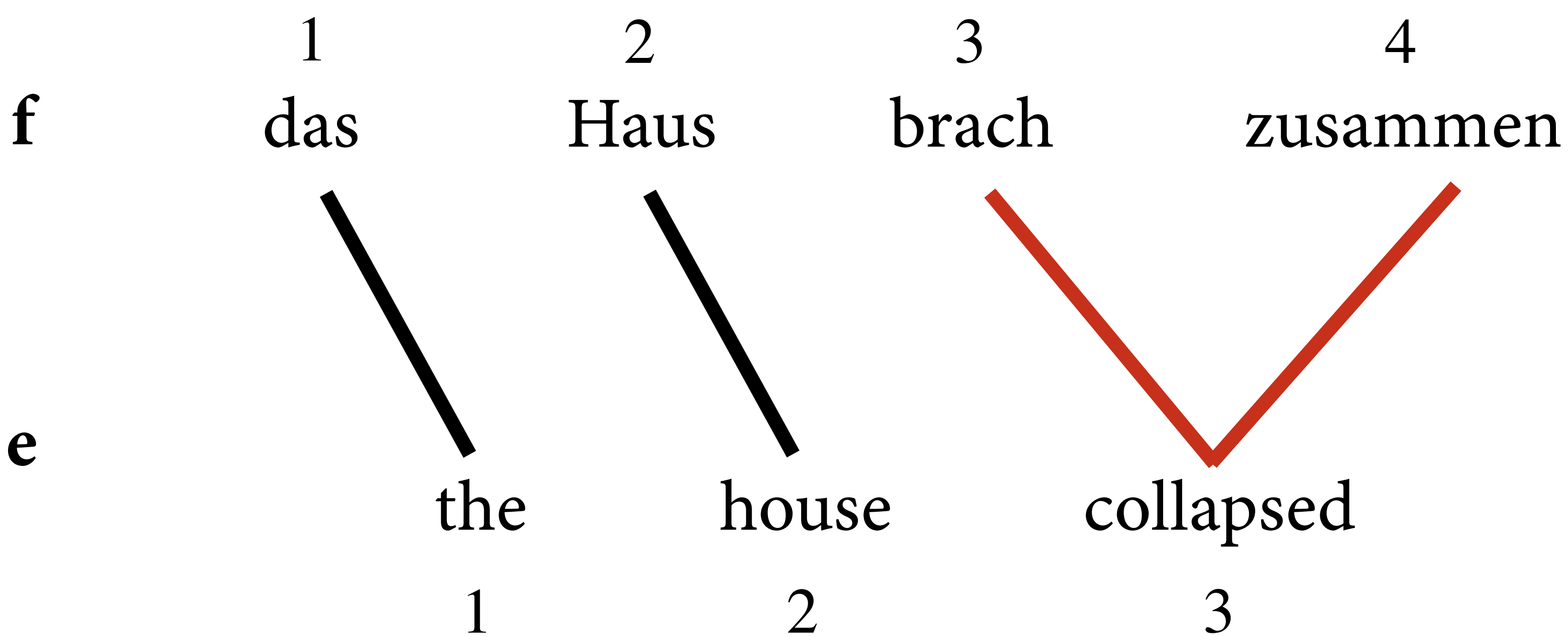
- A Foreign word may translate into *more than one* English word.



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

Many-to-one Translation

- *More than one* Foreign word may *not* translate into a single English word (can't represent this).

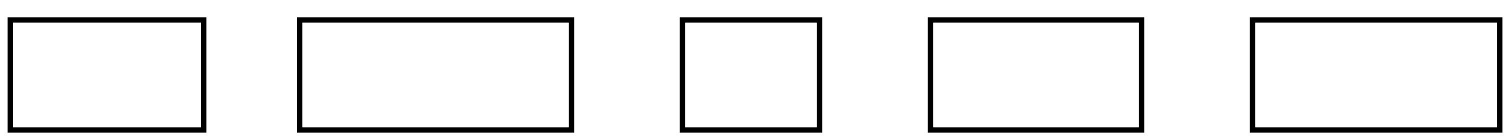


a = ????

Statistical model

- Generative story: Given Foreign string **f** and length m of English string, alignments **a** and English string **e** are generated randomly.

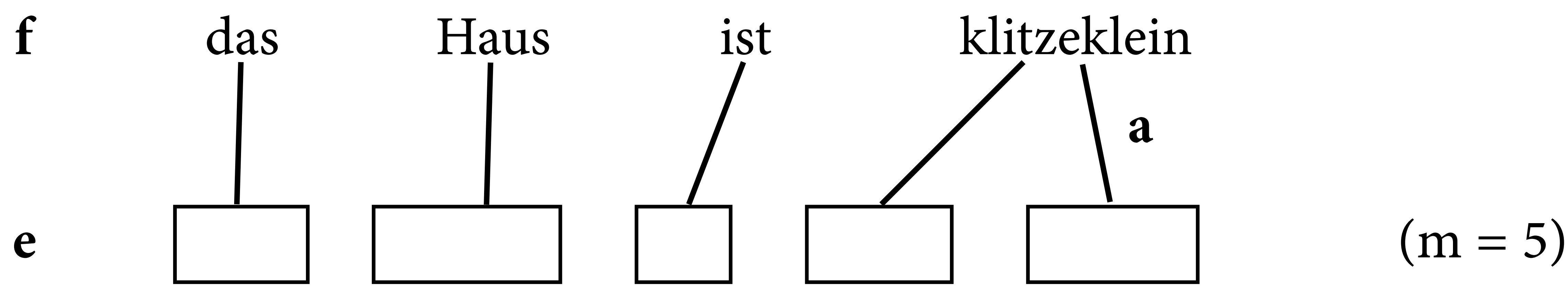
f das Haus ist klitzeklein

e  (m = 5)

- Model P(**a**, **e** | **f**, m) = P(**e** | **a**, **f**, m) * P(**a** | **f**, m).
  - obtain P(**e** | **f**, m) by marginalizing **a** out → translation
  - obtain P(**a** | **f**, m) by marginalizing **e** out → compute alignments

Statistical model

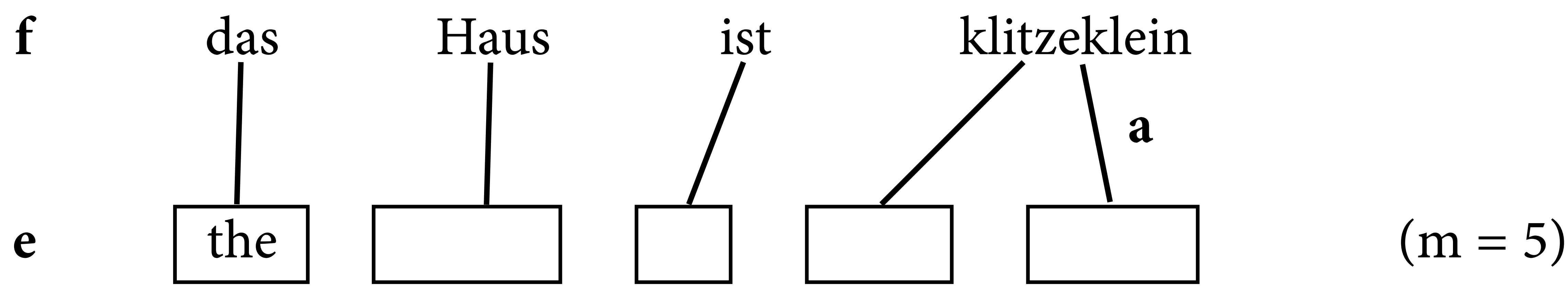
- Generative story: Given Foreign string **f** and length m of English string, alignments **a** and English string **e** are generated randomly.



- Model P(**a**, **e** | **f**, m) = P(**e** | **a**, **f**, m) * P(**a** | **f**, m).
  - obtain P(**e** | **f**, m) by marginalizing **a** out → translation
  - obtain P(**a** | **f**, m) by marginalizing **e** out → compute alignments

Statistical model

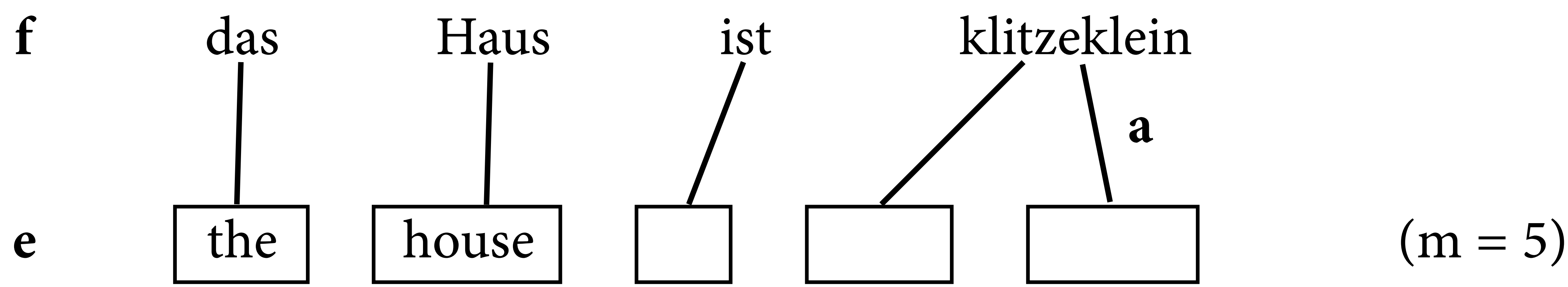
- Generative story: Given Foreign string **f** and length m of English string, alignments **a** and English string **e** are generated randomly.



- Model P(**a**, **e** | **f**, m) = P(**e** | **a**, **f**, m) * P(**a** | **f**, m).
  - obtain P(**e** | **f**, m) by marginalizing **a** out → translation
  - obtain P(**a** | **f**, m) by marginalizing **e** out → compute alignments

Statistical model

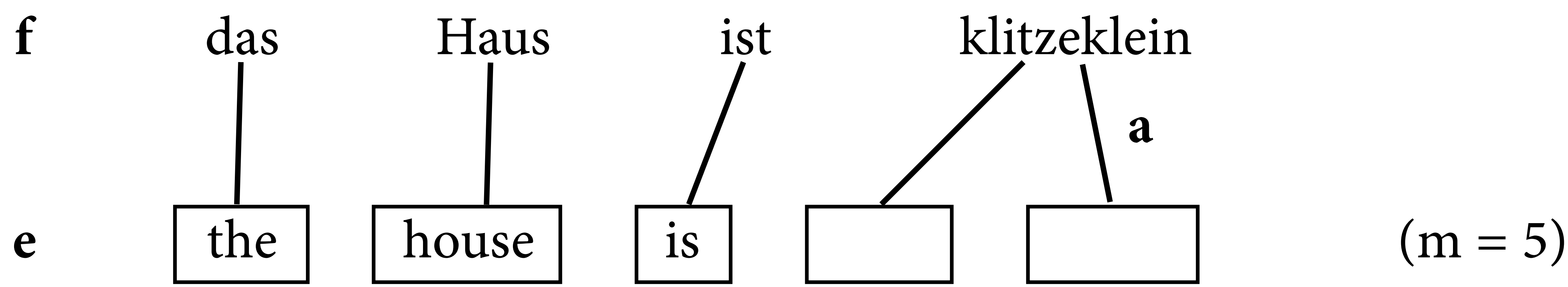
- Generative story: Given Foreign string **f** and length m of English string, alignments **a** and English string **e** are generated randomly.



- Model P(**a**, **e** | **f**, m) = P(**e** | **a**, **f**, m) * P(**a** | **f**, m).
  - obtain P(**e** | **f**, m) by marginalizing **a** out → translation
  - obtain P(**a** | **f**, m) by marginalizing **e** out → compute alignments

Statistical model

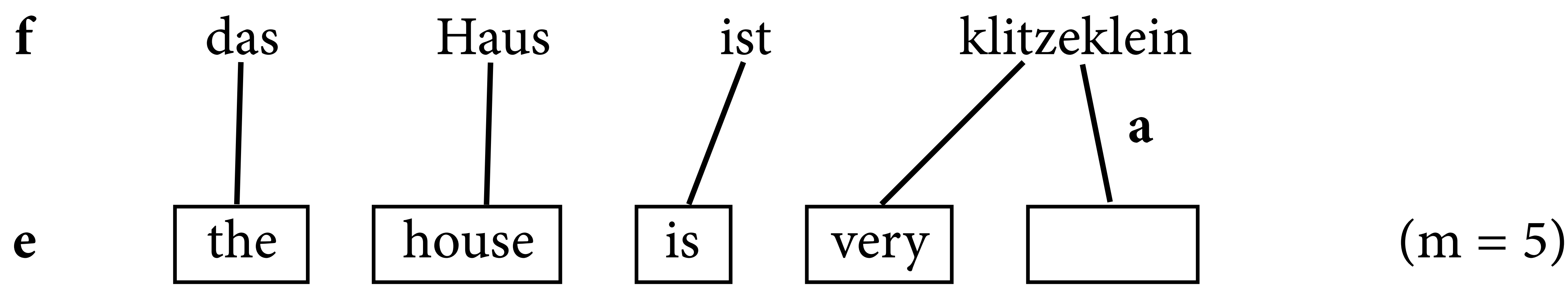
- Generative story: Given Foreign string **f** and length m of English string, alignments **a** and English string **e** are generated randomly.



- Model P(**a**, **e** | **f**, m) = P(**e** | **a**, **f**, m) * P(**a** | **f**, m).
  - obtain P(**e** | **f**, m) by marginalizing **a** out → translation
  - obtain P(**a** | **f**, m) by marginalizing **e** out → compute alignments

Statistical model

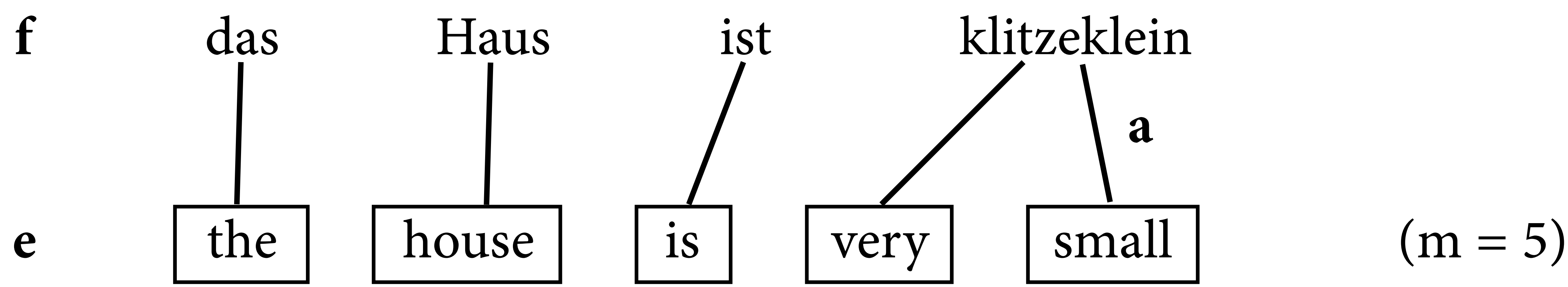
- Generative story: Given Foreign string **f** and length m of English string, alignments **a** and English string **e** are generated randomly.



- Model P(**a**, **e** | **f**, m) = P(**e** | **a**, **f**, m) * P(**a** | **f**, m).
  - obtain P(**e** | **f**, m) by marginalizing **a** out → translation
  - obtain P(**a** | **f**, m) by marginalizing **e** out → compute alignments

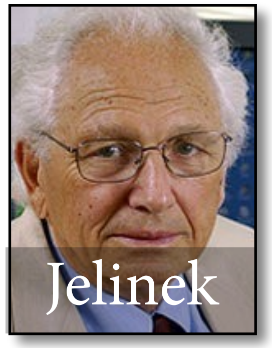
Statistical model

- Generative story: Given Foreign string **f** and length m of English string, alignments **a** and English string **e** are generated randomly.



- Model P(**a**, **e** | **f**, m) = P(**e** | **a**, **f**, m) * P(**a** | **f**, m).
  - obtain P(**e** | **f**, m) by marginalizing **a** out → translation
  - obtain P(**a** | **f**, m) by marginalizing **e** out → compute alignments

IBM Model 1



- Simplifying assumptions:
  - The alignment decisions for the *m* English words are independent.
  - The alignment distribution for each a_i is uniform over all source words and NULL.
  - The English words are generated independently, conditioned only on their aligned Foreign words.

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

IBM Model 1

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

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

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

$$P(e_i, a_i \mid \mathbf{f}, m) = P(a_i \mid \mathbf{f}, m) \cdot P(e_i \mid a_i, \mathbf{f}, m) = \frac{1}{n+1} \cdot P(e_i \mid f_{a_i})$$

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

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

Example

das

e	$t(e\|f)$
the	0.7
that	0.15
which	0.075
who	0.05
this	0.025

Haus

e	$t(e\|f)$
house	0.8
building	0.16
home	0.02
household	0.015
shell	0.005

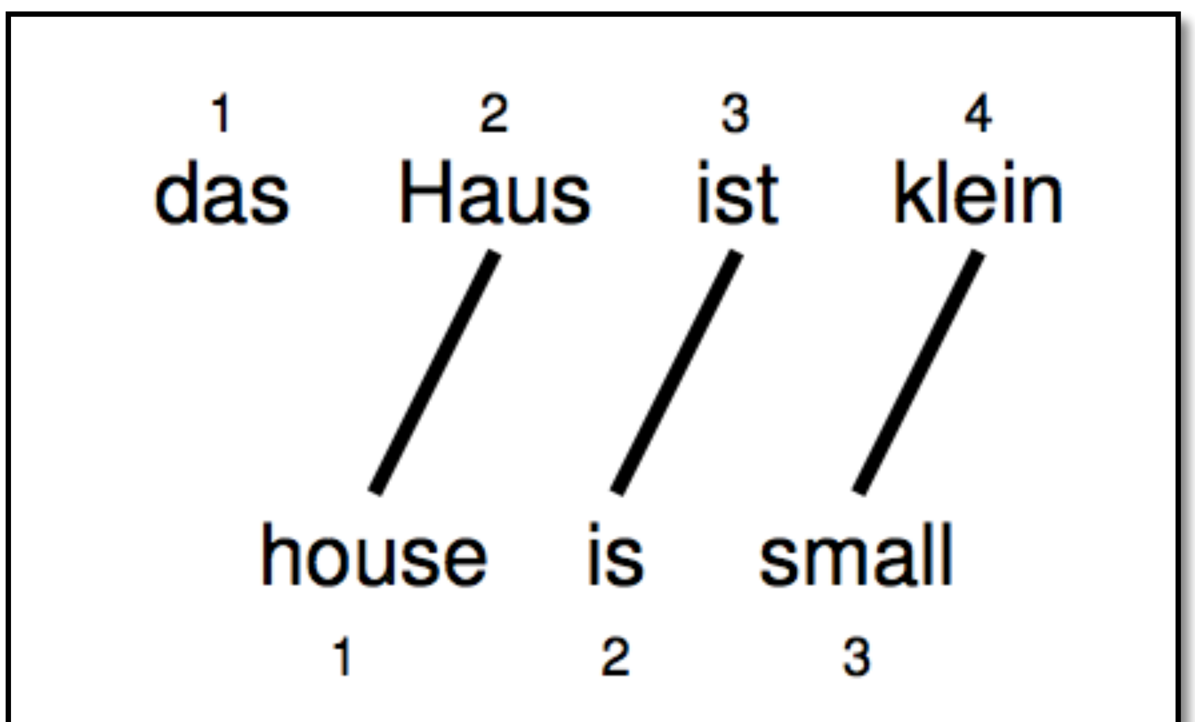
ist

e	$t(e\|f)$
is	0.8
's	0.16
exists	0.02
has	0.015
are	0.005

klein

e	$t(e\|f)$
small	0.4
little	0.4
short	0.1
minor	0.06
petty	0.04

t(e | f) = P(e | f)



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

$$
\begin{aligned}
P(\mathbf{e}, \mathbf{a} \mid \mathbf{f}, m) = \\
&\quad 1/5 * P(\text{Haus}|\text{house}) * \\
&\quad 1/5 * P(\text{ist}|\text{is}) * \\
&\quad 1/5 * P(\text{small}|\text{klein}) \\
&= 1/125 * 0.8 * 0.8 * 0.4 \\
&= 0.002
\end{aligned}
$$

Computing best alignments

- Assume that we know parameters P(e | f) and we are given **e** and **f**. What is alignment **a** that maximizes P(**a** | **e**, **f**)?

- Because of independence of $a_1, \ldots, a_m$, can choose best aligned word in **f** for each word in **e** separately.

$$
\begin{aligned}
a_i^* &= \arg\max_{a_i=0}^{n} \frac{1}{1+n} p(e_i \mid f_{a_i}) \\
&= \arg\max_{a_i=0}^{n} p(e_i \mid f_{a_i})
\end{aligned}
$$

Training

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

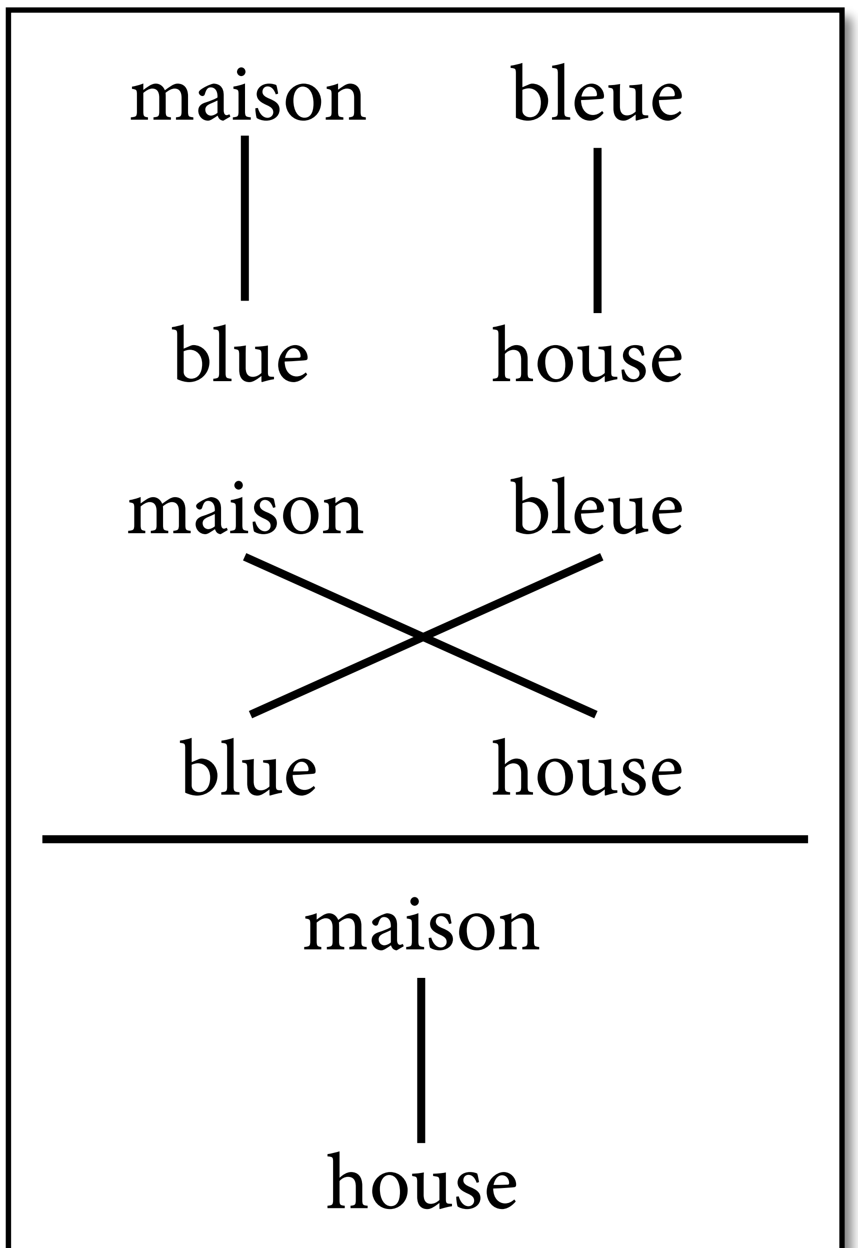
- Parameters of our model: translation probs P(e | f) for any two words e and f.

- If we could observe alignments, then we could just do MLE: C(e aligned with f) / C(f)

- Because we usually only have raw parallel text, we need to use EM.
  - estimate counts from estimate of P
  - re-estimate P from estimated counts

EM: An Example

P(e|f)	house	blue
maison	0.5	0.5
bleue	0.5	0.5



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

*1. Compute P(**e**, **a** | **f**) for each alignment of each sentence pair.*

$P(\mathbf{e}_1, \mathbf{a}_{11} \mid \mathbf{f}_1) = 1/9 * 1/2 * 1/2 = 1/36$

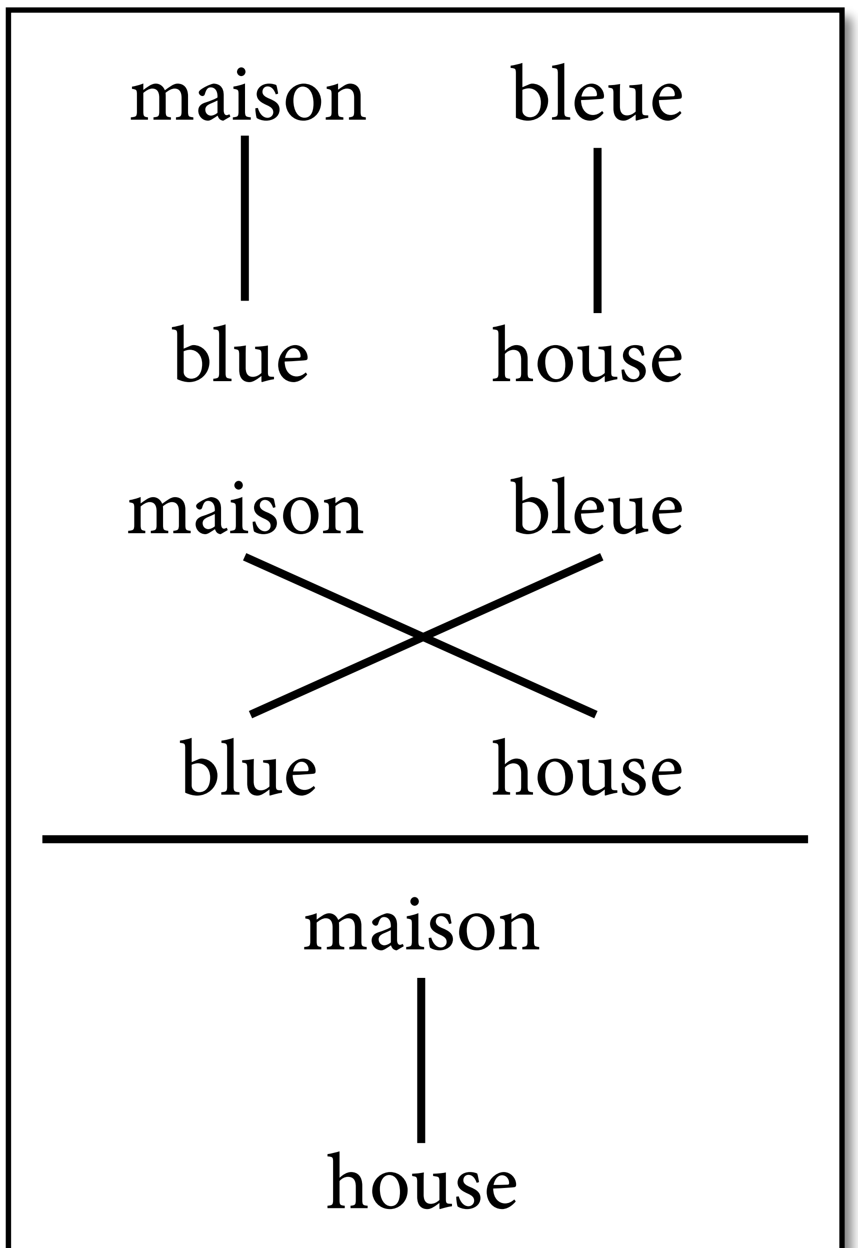
$P(\mathbf{e}_1, \mathbf{a}_{12} \mid \mathbf{f}_1) = 1/9 * 1/2 * 1/2 = 1/36$

$P(\mathbf{e}_2, \mathbf{a}_2 \mid \mathbf{f}_2) = 1/2 * 1/2 = 1/4$

(note: these are not really all alignments)

EM: An Example

P(e|f)	house	blue
maison	0.5	0.5
bleue	0.5	0.5



*2. Normalize P(**e**, **a** | **f**) to yield P(**a** | **e**, **f**).*

$$P(\mathbf{a} \mid \mathbf{e}, \mathbf{f}) = \frac{P(\mathbf{a}, \mathbf{e} \mid \mathbf{f})}{P(\mathbf{e} \mid \mathbf{f})} = \frac{P(\mathbf{a}, \mathbf{e} \mid \mathbf{f})}{\sum_{\mathbf{a}'} P(\mathbf{a}', \mathbf{e} \mid \mathbf{f})}$$

$P(\mathbf{a}_{11} \mid \mathbf{e}_1, \mathbf{f}_1) = 1/2$

$P(\mathbf{a}_{12} \mid \mathbf{e}_1, \mathbf{f}_1) = 1/2$

$P(\mathbf{a}_2 \mid \mathbf{e}_2, \mathbf{f}_2) = 1$

3. collect expected counts

tc	house	blue
maison	3/2	1/2
bleue	1/2	1/2

EM: An Example

*4. Normalize expected counts C(**e**, **f**)*
*by total expected counts C(**f**)*
*to obtain revised translation probs P(**e** | **f**).*

expected counts

tc	house	blue
maison	3/2	1/2
bleue	1/2	1/2

→

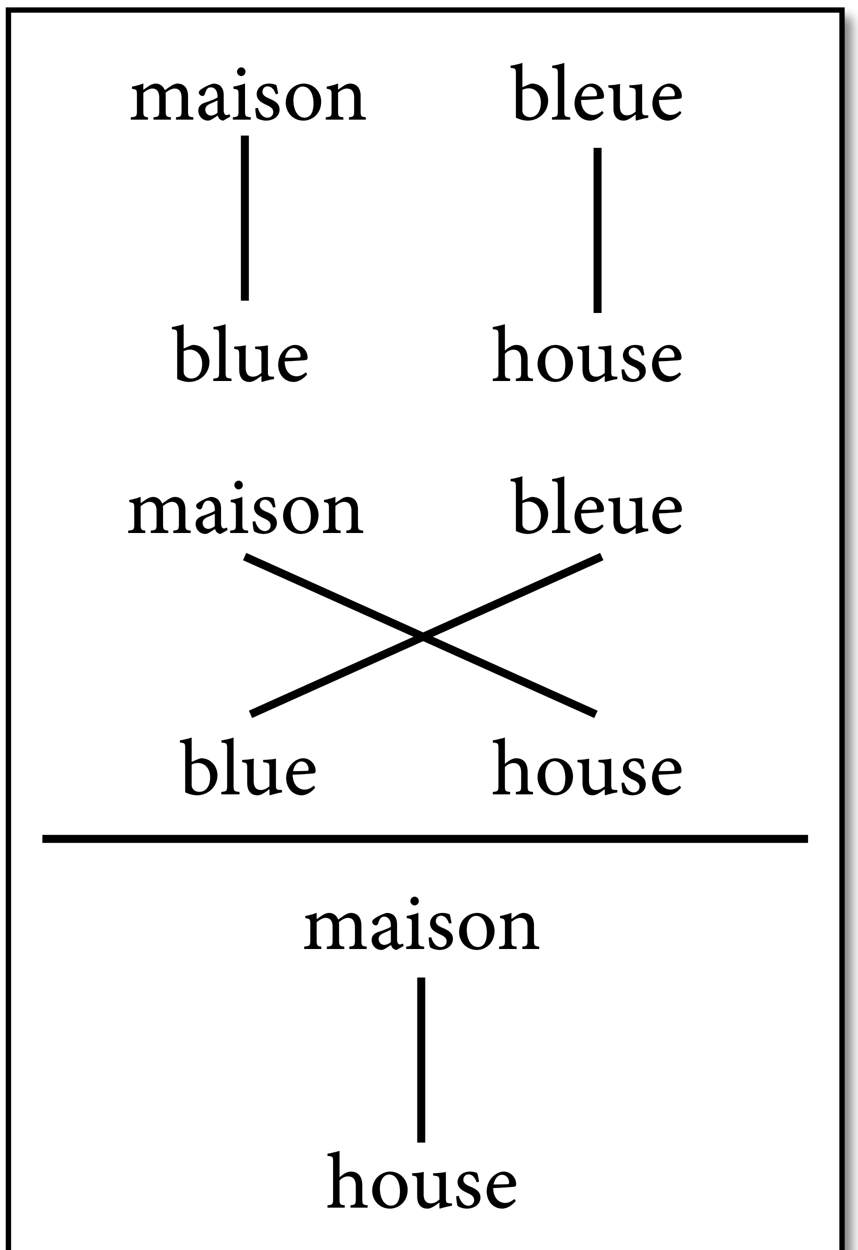
revised translation probs

P(e|f)	house	blue
maison	3/4	1/4
bleue	1/2	1/2

EM: Round Two

P(e|f)	house	blue
maison	3/4	1/4
bleue	1/2	1/2

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



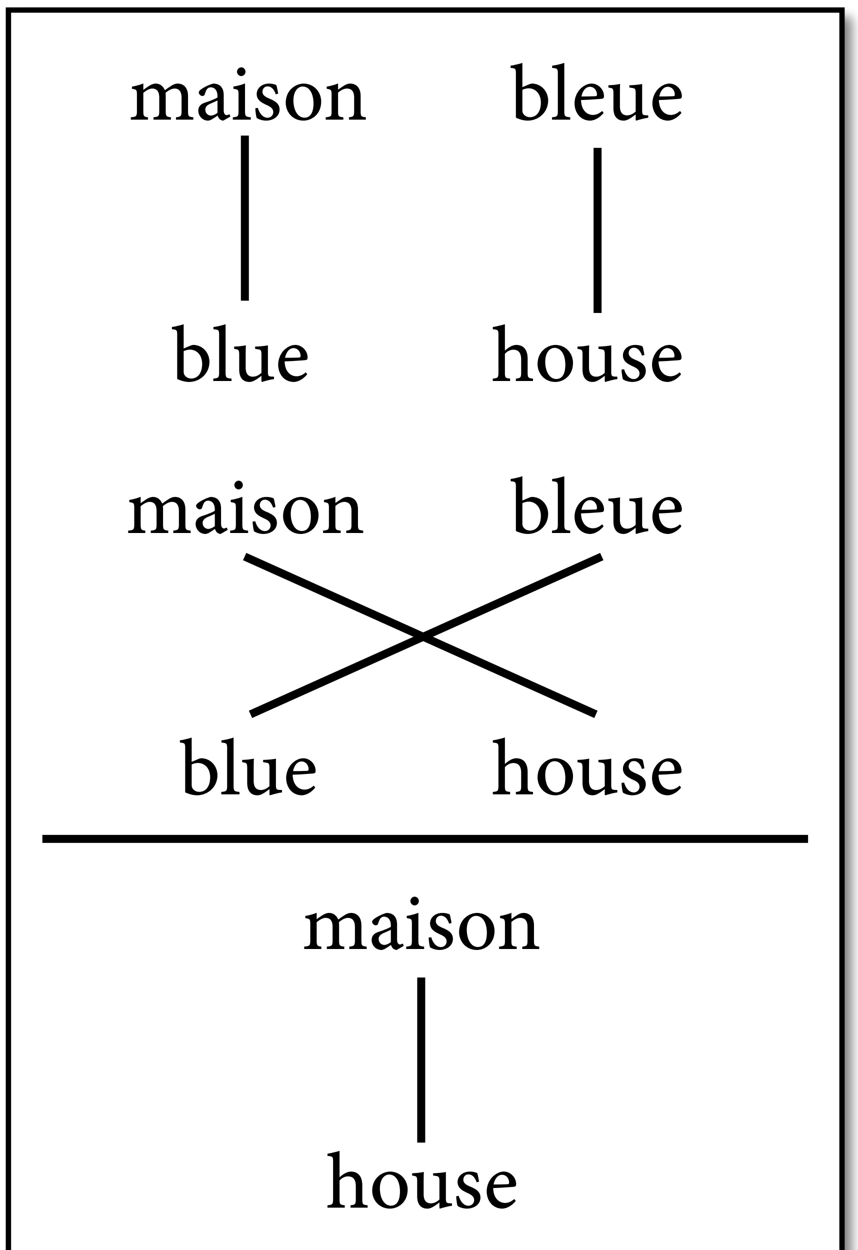
$P(e_1, a_{11} \mid f_1) = 1/9 * 1/4 * 1/2 = 1/72$

$P(e_1, a_{12} \mid f_1) = 1/9 * 3/4 * 1/2 = 3/72$

$P(e_2, a_2 \mid f_2) = 1/2 * 3/4 = 3/8$

EM: Round Two

P(e|f)	house	blue
maison	3/4	1/4
bleue	1/2	1/2



$P(a_{11} \mid e_1, f_1) = 1/4$

$P(a_{12} \mid e_1, f_1) = 3/4$

$P(a_2 \mid e_2, f_2) = 1$

tc	house	blue
maison	7/4	1/4
bleue	1/4	3/4

EM: Round Two

expected counts

tc	house	blue
maison	7/4	1/4
bleue	1/4	3/4

→

revised translation probs

P(e|f)	house	blue
maison	7/8	1/8
bleue	1/4	3/4

After many iterations:

P(e|f)	house	blue
maison	≈ 1	≈ 0
bleue	≈ 0	≈ 1

Efficient computation

- Computation of P(**a** | **e**, **f**) in E-step is tricky:

$$P(a_i = j \mid \mathbf{e}, \mathbf{f}) = \frac{P(a_i = j, \mathbf{e} \mid \mathbf{f})}{P(\mathbf{e} \mid \mathbf{f})} = \frac{\sum_{\mathbf{a}:a_i=j} \prod_{i'=1}^{m} P(e_{i'} \mid f_{a_{i'}})}{\sum_{\mathbf{a}} \prod_{i'=1}^{m} P(e_{i'} \mid f_{a_{i'}})}$$

- Summation over **a** is exponential in sentence length.

- By clever use of law of distributivity, can rewrite this term so it can be computed in quadratic time.
  See Lopez tutorial on website. (Note flipped **e** and **f**.)

Extensions

- IBM Model 2: P(**a**) not uniform, but implements *reordering model* that prefers alignments in which words stay close to their original position.
- Model 3: adds *fertility model* that predicts the number of English words to which a given f will be aligned. Can't do EM, approximate with sampling.
- Models 4-5: more complicated reordering models.
- Implemented in GIZA++ and successor tools.

Conclusion

- Machine translation: one of the most useful and most challenging disciplines of NLP.
- Today: word alignments.
  - IBM Model 1
  - computing best alignments
  - EM training
  - advanced models
- Next time: let's actually translate something.