1. What's the Māori word for…
(a) "long"?
(b) "hot"?

2. How would you order a large cappuccino?

3. What's the word for chocolate?

AthNLP 2025

Machine Translation and Multilinguality

Antonis Anastasopoulos

Acknowledgement: Many slides are taken from
Greg Durrett CS388@UT Austin,
Graham Neubig's Advanced NLP course@CMU
and Philipp Koehn's MT course@JHU

Machine Translation

Machine Translation

- Intro

Machine Translation

- Intro
- A historical note

Machine Translation

- Intro
- A historical note
  - Alignment and EM algorithm

Machine Translation

- Intro
- A historical note
    - Alignment and EM algorithm
- MT Evaluation

Machine Translation

- Intro
- A historical note
    - Alignment and EM algorithm
- MT Evaluation
- Neural MT and LLM-based MT

Machine Translation

- Intro
- A historical note
  - Alignment and EM algorithm
- MT Evaluation
- Neural MT and LLM-based MT
- Semi-supervised and Unsupervised MT

Machine translation

Machine translation

- Machine Translation (MT) is the task of translating a sentence x from one language (the source language) to a sentence y in another language (the target language).

Machine translation

- Machine Translation (MT) is the task of translating a sentence x from one language (the source language) to a sentence y in another language (the target language).

Machine translation

- Machine Translation (MT) is the task of translating a sentence x from one language (the source language) to a sentence y in another language (the target language).

x: *L'homme est né libre, et partout il est dans les fers*

Machine translation

- Machine Translation (MT) is the task of translating a sentence x from one language (the source language) to a sentence y in another language (the target language).

x: *L'homme est né libre, et partout il est dans les fers*

Machine translation

- Machine Translation (MT) is the task of translating a sentence x from one language (the source language) to a sentence y in another language (the target language).

x: *L'homme est né libre, et partout il est dans les fers*

y: Man is born free, but everywhere he is in chains

Machine translation

- Machine Translation (MT) is the task of translating a sentence x from one language (the source language) to a sentence y in another language (the target language).

x: *L'homme est né libre, et partout il est dans les fers*

y: Man is born free, but everywhere he is in chains

Machine Translation

Machine Translation

- The classic test of language understanding!

Machine Translation

- The classic test of language understanding!
  - Both language analysis & generation

Machine Translation

- The classic test of language understanding!
  - Both language analysis & generation
- Big MT needs ... for humanity ... and commerce

Machine Translation

- The classic test of language understanding!
    - Both language analysis & generation

- Big MT needs ... for humanity ... and commerce
    - Translation is a US$40 billion a year industry

Machine Translation

- The classic test of language understanding!
  - Both language analysis & generation

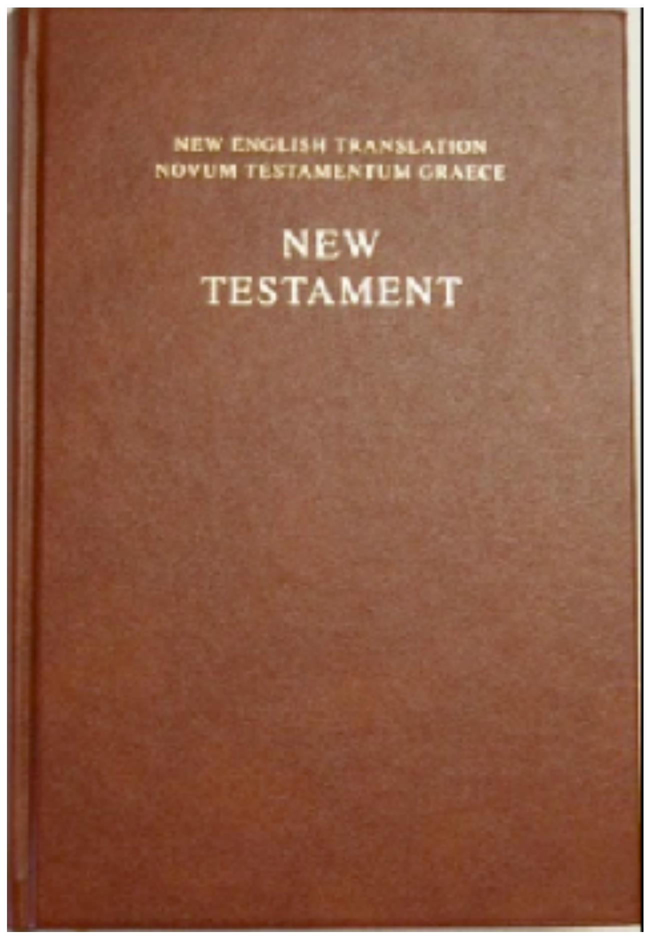
- Big MT needs ... for humanity ... and commerce
  - Translation is a US$40 billion a year industry
  - Huge in Europe, growing in Asia

Machine Translation

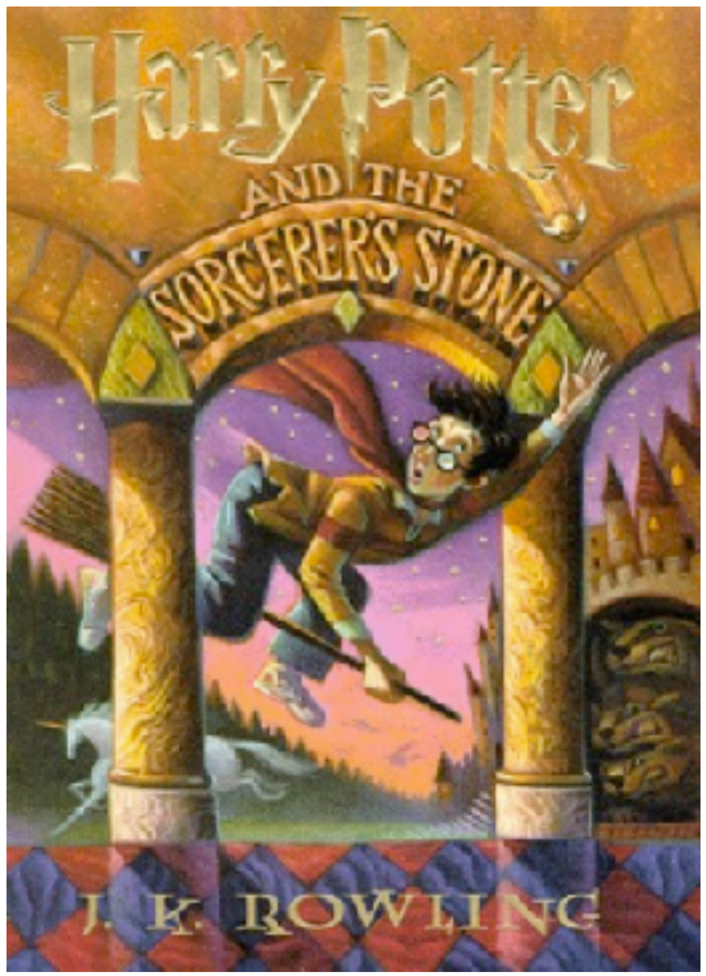
- The classic test of language understanding!
    - Both language analysis & generation
- Big MT needs ... for humanity ... and commerce
    - Translation is a US$40 billion a year industry
    - Huge in Europe, growing in Asia
    - Large social/government/military as well as commercial needs

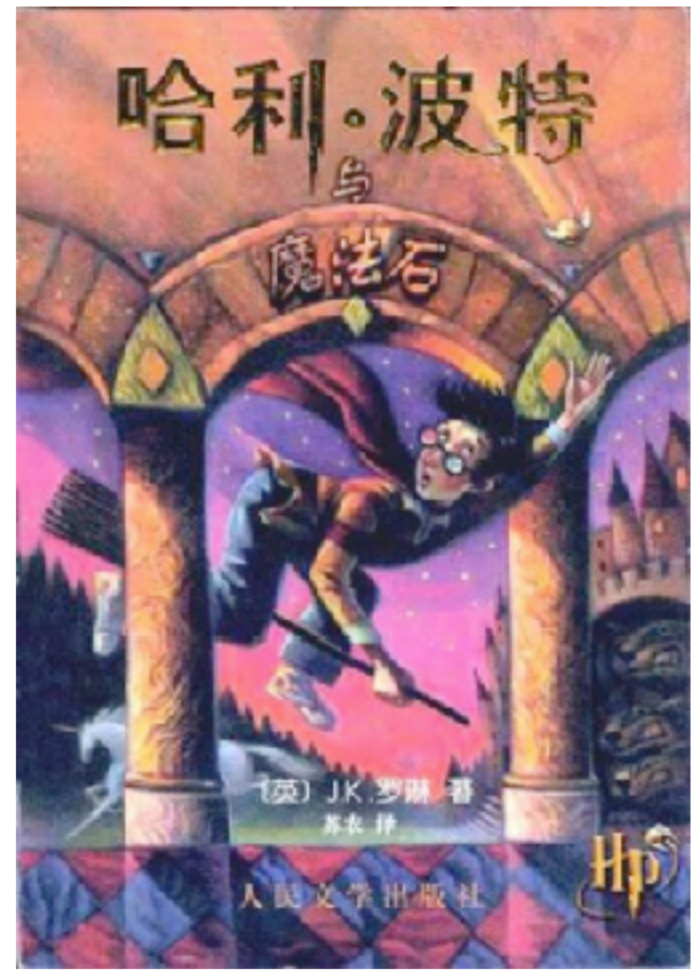
www.un.org
http://www.un.org/english/
Welcome to the
United Nations

http://www.un.org/chinese/
欢迎来到
联合国


NEW ENGLISH TRANSLATION
NOVUM TESTAMENTUM GRAECE
NEW
TESTAMENT


聖經


Harry Potter
AND THE
SORCERER'S STONE
J. K. ROWLING


哈利·波特
与
魔法石
(英) J.K.罗琳 著
苏农 译
人民文学出版社
HP

CLASSIC SOUPS

			Sm.	Lg.
清燉雞湯	57.	House Chicken Soup (Chicken, Celery, Potato, Onion, Carrot)	1.50	2.75
雞飯湯	58.	Chicken Rice Soup	1.85	3.25
雞麵湯	59.	Chicken Noodle Soup	1.85	3.25
廣東雲吞	60.	Cantonese Wonton Soup	1.50	2.75
蕃茄蛋湯	61.	Tomato Clear Egg Drop Soup	1.65	2.95
雲吞湯	62.	Regular Wonton Soup	1.10	2.10
酸辣湯	63.	Hot & Sour Soup	1.10	2.10
蛋花湯	64.	Egg Drop Soup	1.10	2.10
雲蛋湯	65.	Egg Drop Wonton Mix	1.10	2.10
豆腐菜湯	66.	Tofu Vegetable Soup	NA	3.50
雞玉米湯	67.	Chicken Corn Cream Soup	NA	3.50
蟹肉玉米湯	68.	Crab Meat Corn Cream Soup	NA	3.50
海鮮湯	69.	Seafood Soup	NA	3.50

The need for Machine Translation

The need for Machine Translation

- Huge commercial use

The need for Machine Translation

- Huge commercial use
    - Google translates over 100 billion words a day

The need for Machine Translation

- Huge commercial use
  - Google translates over 100 billion words a day
  - Facebook in 2016 rolled out new homegrown MT

The need for Machine Translation

- Huge commercial use
    - Google translates over 100 billion words a day
    - Facebook in 2016 rolled out new homegrown MT
    - eBay uses MT to enable cross-border trade

The need for Machine Translation

- Huge commercial use
    - Google translates over 100 billion words a day
    - Facebook in 2016 rolled out new homegrown MT
    - eBay uses MT to enable cross-border trade
- NMT is the flagship task for NLP Deep Learning

The need for Machine Translation

- Huge commercial use
  - Google translates over 100 billion words a day
  - Facebook in 2016 rolled out new homegrown MT
  - eBay uses MT to enable cross-border trade
- NMT is the flagship task for NLP Deep Learning
  - RNNs? Encoder-decoder? Attention mechanism?

The need for Machine Translation

- Huge commercial use
  - Google translates over 100 billion words a day
  - Facebook in 2016 rolled out new homegrown MT
  - eBay uses MT to enable cross-border trade
- NMT is the flagship task for NLP Deep Learning
  - RNNs? Encoder-decoder? Attention mechanism?
- NMT research has pioneered many of the recent innovations of NLP Deep Learning

A historical note

1950s: Early Machine Translation

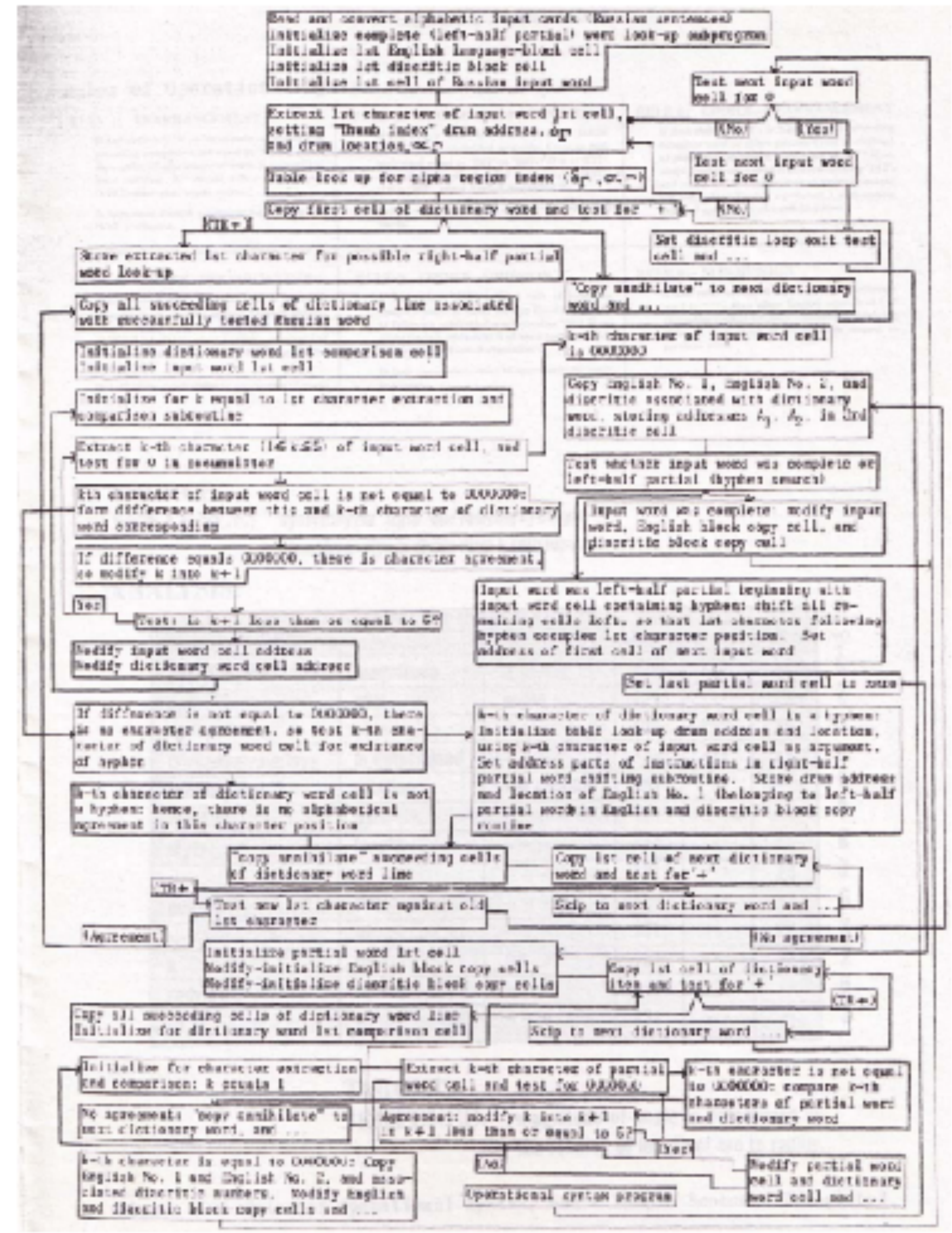


Fig. 7: Flowchart of part of the dictionary lookup procedures (from Sheridan 1955)

Flow chart of the dictionary look-up procedures (source)

1950s: Early Machine Translation

- Machine Translation research began in the early 1950s.

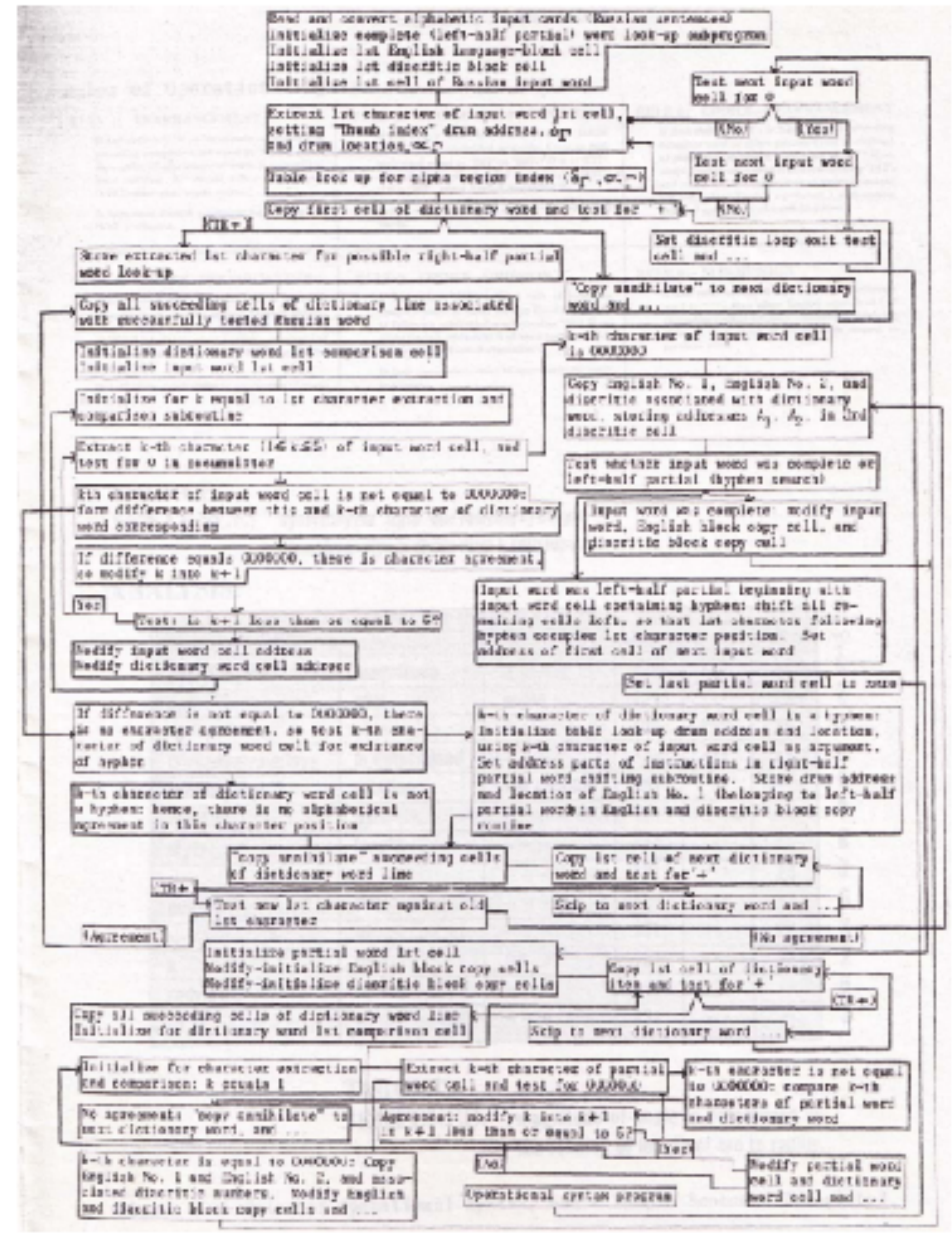


Fig. 7: Flowchart of part of the dictionary lookup procedures (from Sheridan 1955)

Flow chart of the dictionary look-up procedures (source)

1950s: Early Machine Translation

- Machine Translation research began in the early 1950s.
- Mostly Russian → English (motivated by the Cold War)

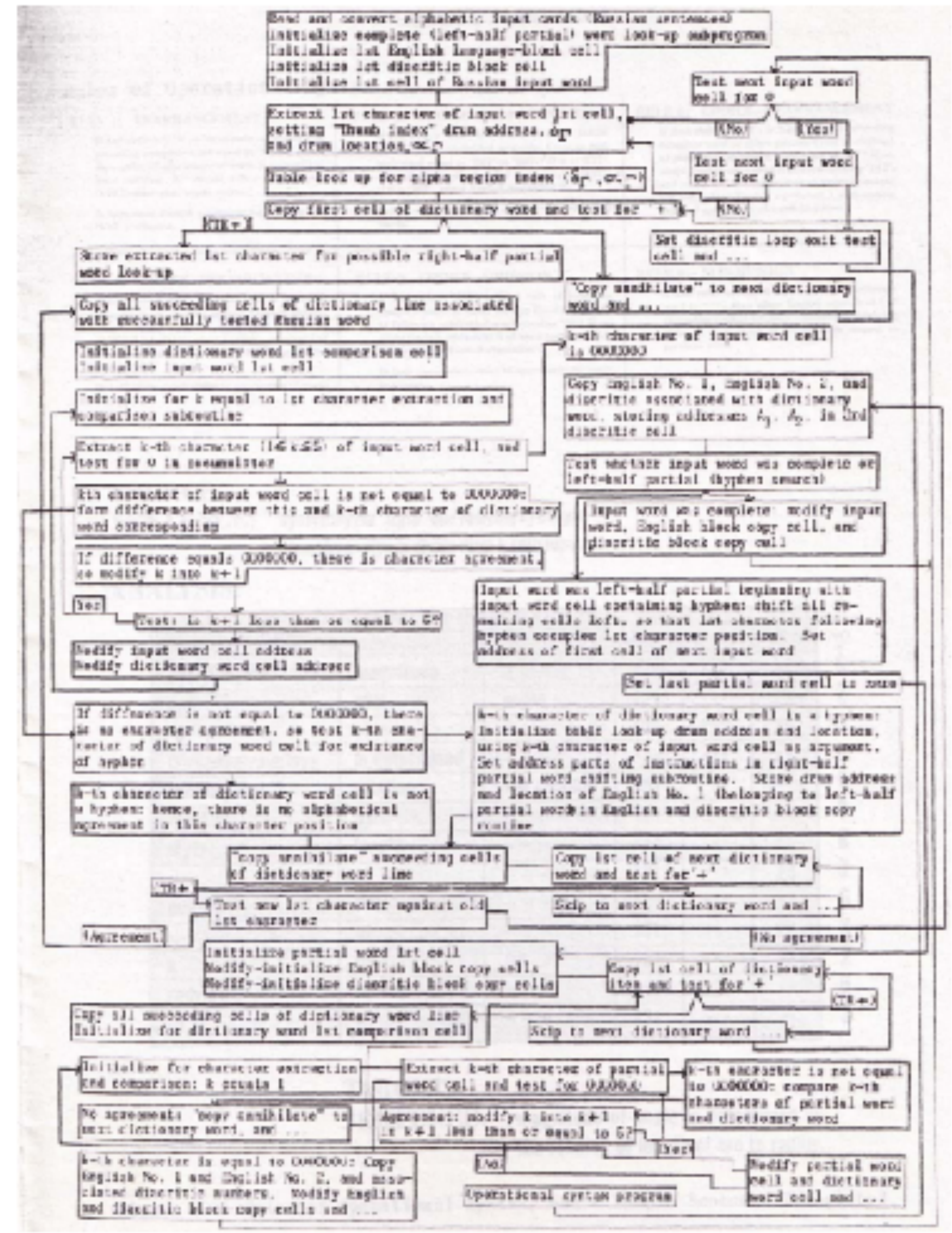


Fig. 7: Flowchart of part of the dictionary lookup procedures (from Sheridan 1955)

Flow chart of the dictionary look-up procedures (source)

1950s: Early Machine Translation

- Machine Translation research began in the early 1950s.
- Mostly Russian → English (motivated by the Cold War)
    - Georgetown–IBM experiment (1954)

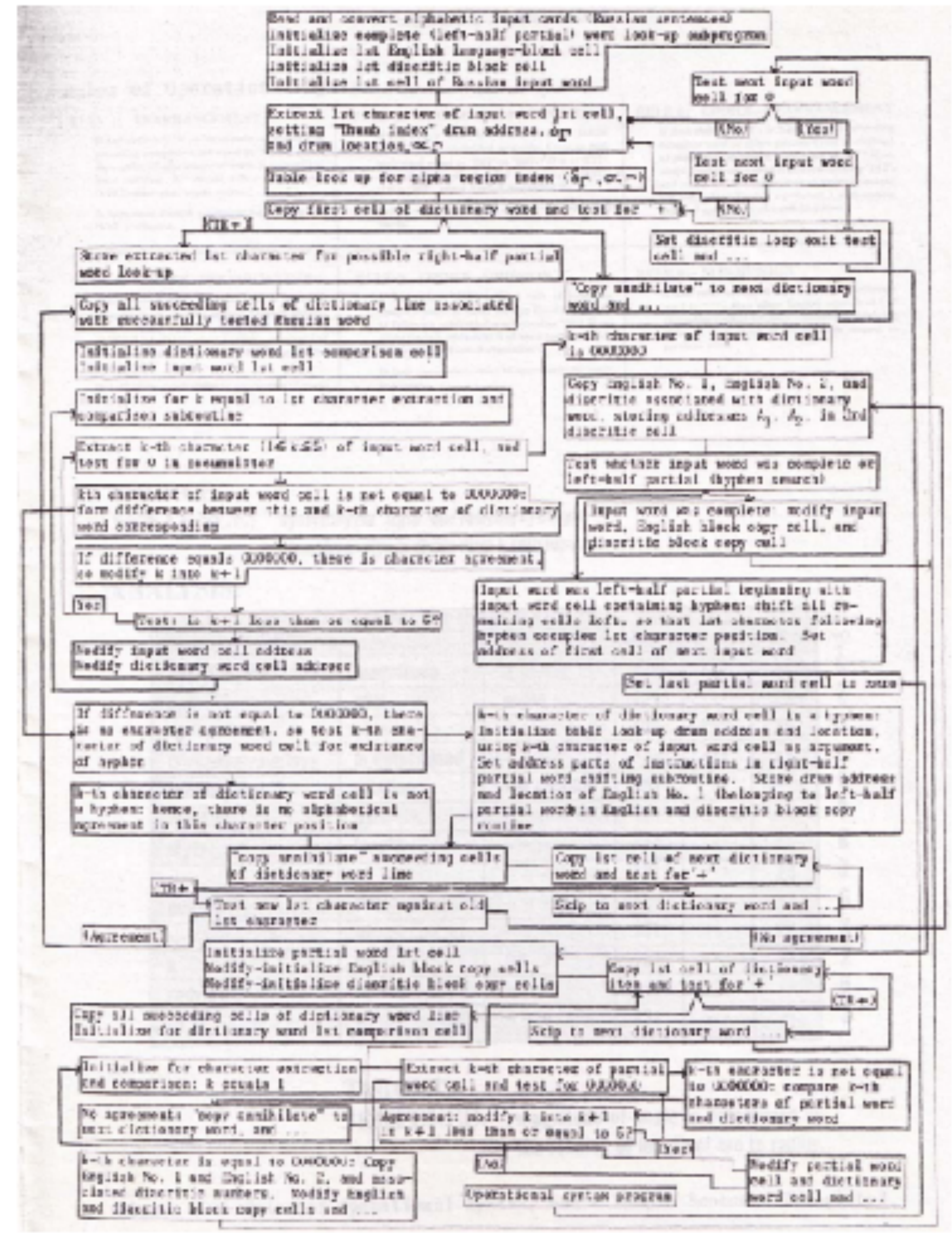


Fig. 7: Flowchart of part of the dictionary lookup procedures (from Sheridan 1955)

Flow chart of the dictionary look-up procedures (source)

1950s: Early Machine Translation

- Machine Translation research began in the early 1950s.
- Mostly Russian → English (motivated by the Cold War)
    - Georgetown–IBM experiment (1954)
- Systems were mostly rule-based, using a bilingual dictionary to map Russian words to their English counterparts

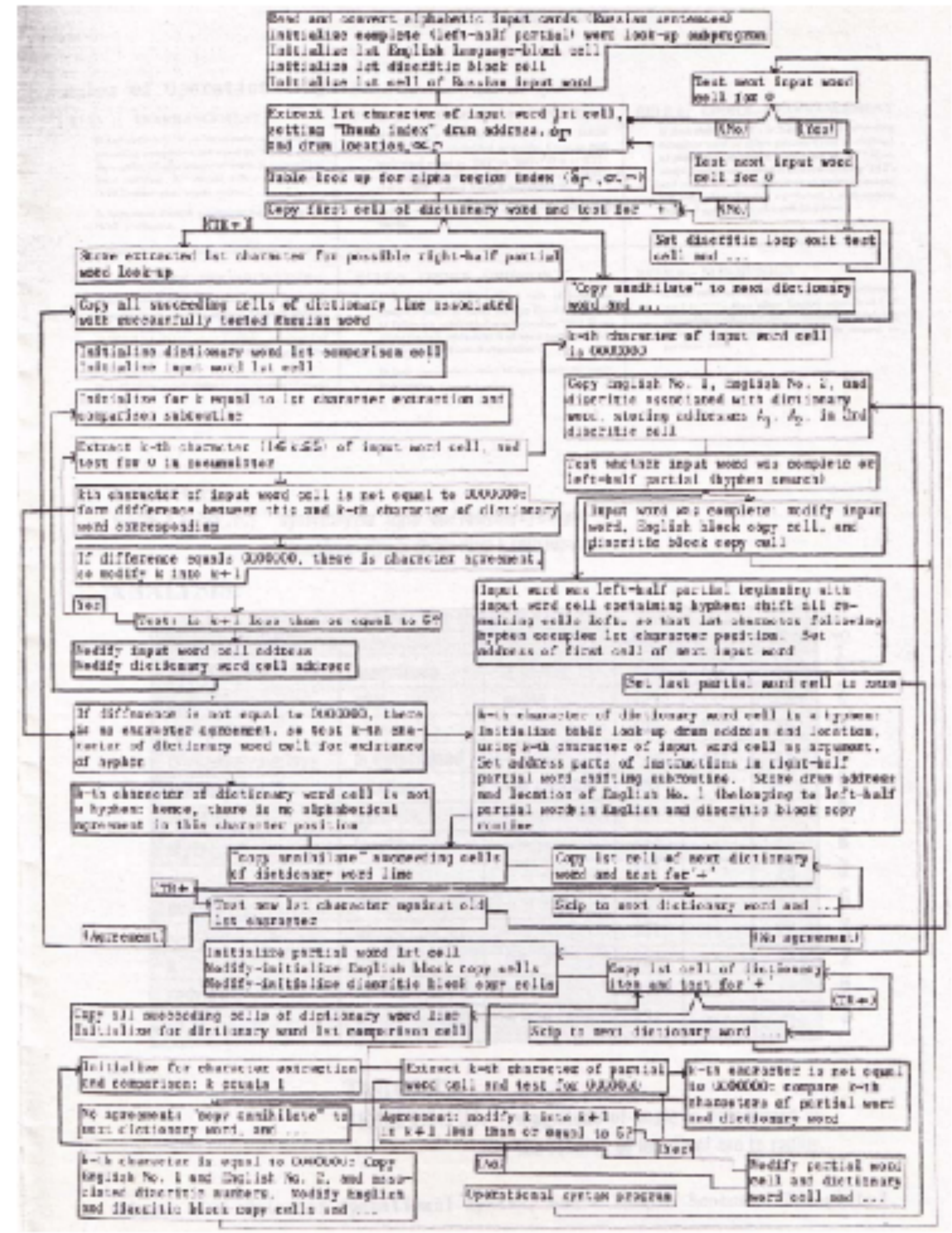


Fig. 7: Flowchart of part of the dictionary lookup procedures (from Sheridan 1955)

Flow chart of the dictionary look-up procedures (source)

1990s-2010s: Statistical Machine Translation

1990s-2010s: Statistical Machine Translation

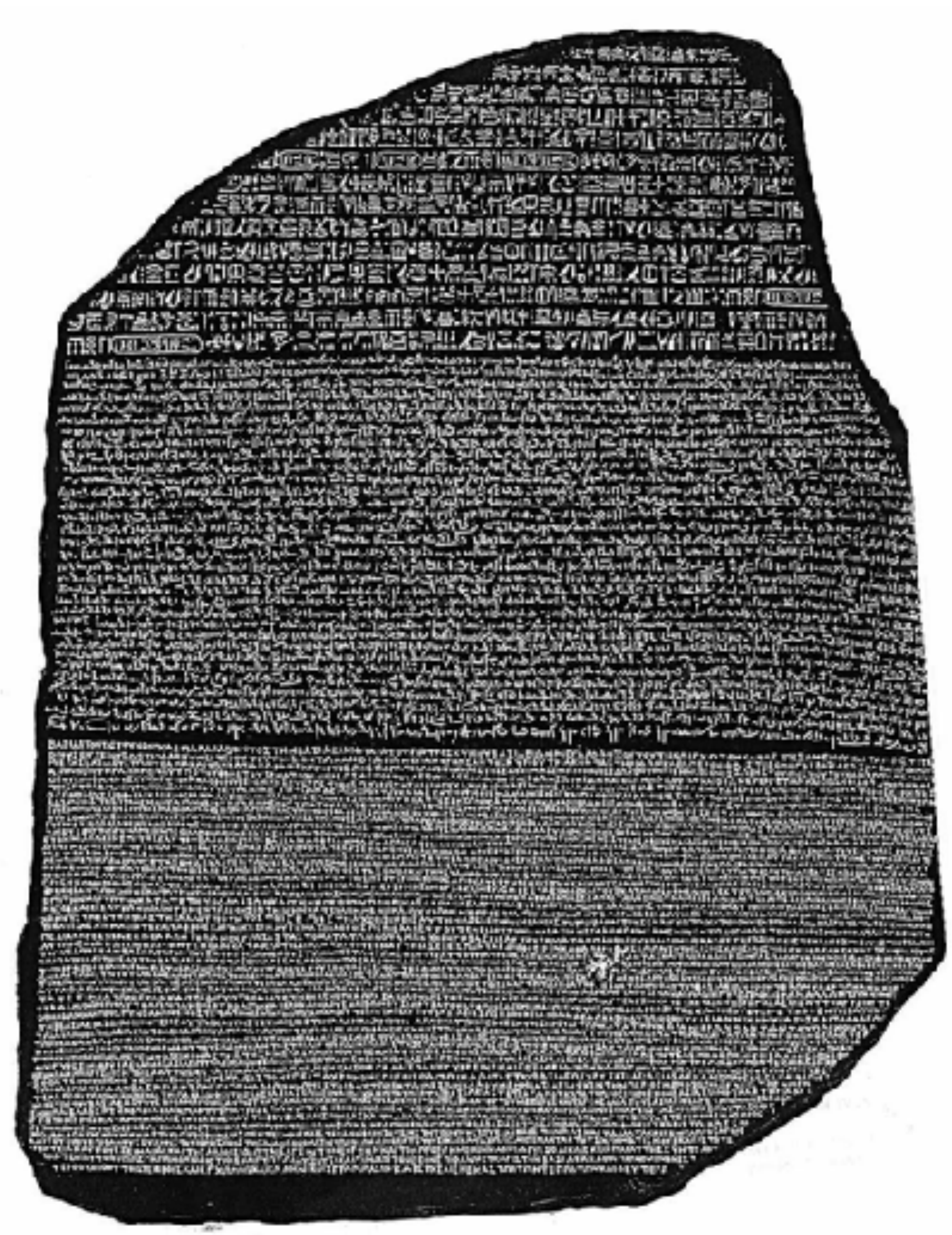
- Core idea: Learn a probabilistic model from data

1990s-2010s: Statistical Machine Translation

- Core idea: Learn a probabilistic model from data
- Suppose we're translating French → English.

1990s-2010s: Statistical Machine Translation

- Core idea: Learn a probabilistic model from data
- Suppose we're translating French → English.
- We want to find best English sentence y, given French sentence x


Egyptian
Greek

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, March, 1947

Noisy Channel MT

We want a model of $p(\boldsymbol{e}|\boldsymbol{f})$

Noisy Channel MT

We want a model of $p(\boldsymbol{e}|\boxed{\boldsymbol{f}})$

Confusing foreign sentence

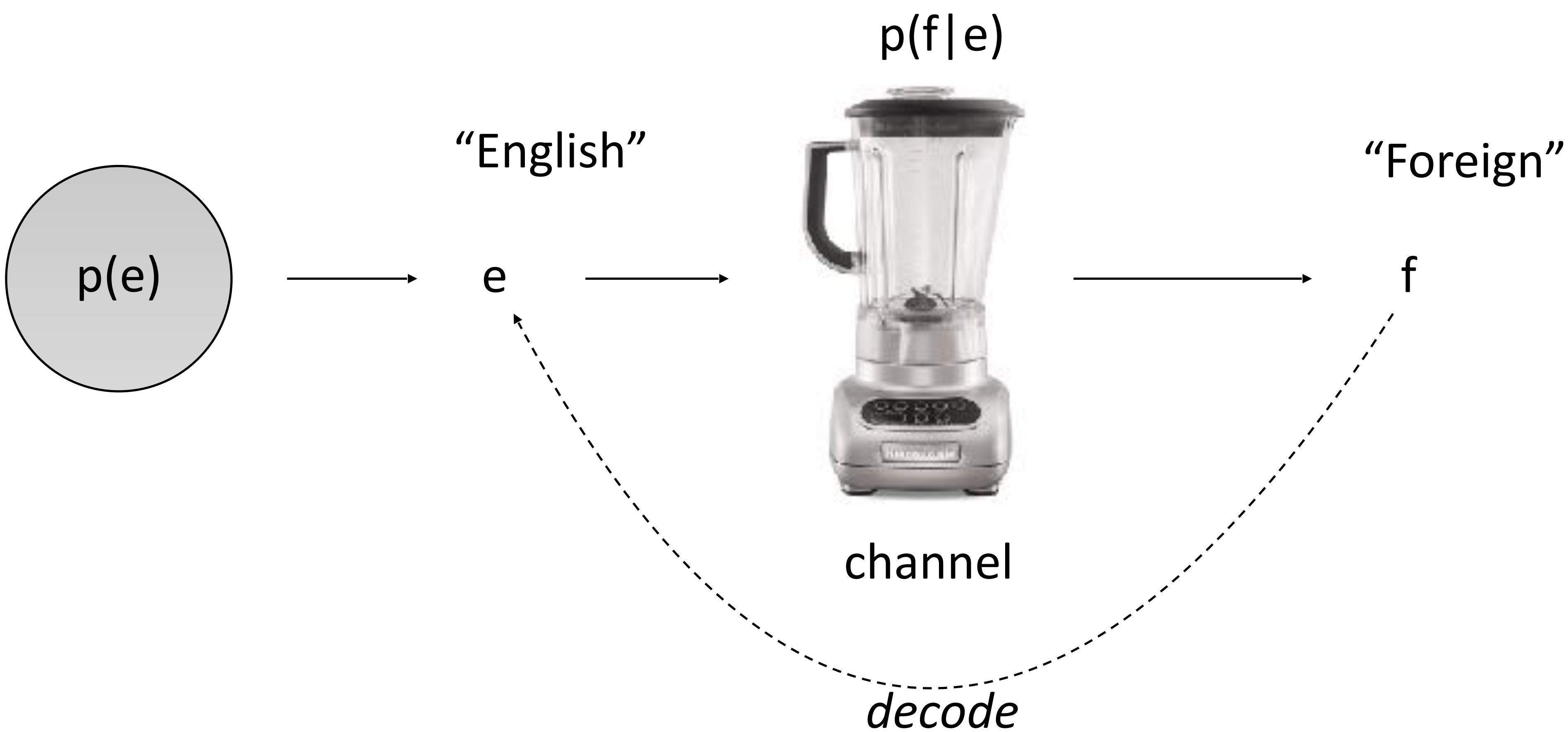
Noisy Channel MT

We want a model of $p(\boxed{\boldsymbol{e}}|\boxed{\boldsymbol{f}})$

Confusing foreign sentence

Possible English translation

Noisy Channel MT



Noisy Channel MT

$$\begin{aligned}\hat{e} &= \arg\max_e p(e|f) \\ &= \arg\max_e \frac{p(e) \times p(f|e)}{p(f)} \\ &= \arg\max_e \underbrace{\boxed{p(e)}}_{\text{“Language Model”}} \times \underbrace{\boxed{p(f|e)}}_{\text{“Translation Model”}}\end{aligned}$$

Noisy Channel Division of Labor

- Language model – $p(\boldsymbol{e})$
  - is the translation fluent, grammatical, and idiomatic?
  - use any model of $p(\boldsymbol{e})$ – typically an *n*-gram model
- Translation model – $p(\boldsymbol{f}|\boldsymbol{e})$
  - translation probability
  - ensures adequacy of translation

Translation Model

- $p(\boldsymbol{f}|\boldsymbol{e})$ gives the channel probability – the probability of translating an English sentence into a foreign sentence
- $\boldsymbol{f}$ = je voudrais un peu de frommage
- $\mathbf{e}_1$ = I would like some cheese
  $\mathbf{e}_2$ = I would like a little of cheese
  $\mathbf{e}_3$ = There is no train to Barcelona

	$p(\boldsymbol{f}\|\boldsymbol{e})$
$\mathbf{e}_1$	0.4
$\mathbf{e}_2$	0.5
$\mathbf{e}_3$	>0.00001

Translation Model

- How do we parameterize $p(\boldsymbol{f}|\boldsymbol{e})$?

$$p(f|e) = \frac{count(f,e)}{count(e)} \quad ?$$

- There are a lot of sentences: this won't generalize to new inputs

Lexical Translation

Lexical Translation

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

Lexical Translation

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

  Haus: house, home, shell, household

Lexical Translation

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

  Haus: house, home, shell, household

- Multiple translations

Lexical Translation

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

  Haus: house, home, shell, household

- Multiple translations
  - Different word senses, different registers, different inflections

Lexical Translation

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

  Haus: house, home, shell, household

- Multiple translations
  - Different word senses, different registers, different inflections
  - *house, home* are common

Lexical Translation

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

  Haus: house, home, shell, household

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

How common is each?

Translation	Count
house	5000
home	2000
shell	100
household	80

MLE

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

Lexical Translation

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

Lexical Translation

- Goal: a model *p(**e**|**f**,m)*
- where **e** and **f** are complete English and Foreign sentences

$$\mathbf{e} = \langle e_1, e_2, \ldots, e_m \rangle$$

Lexical Translation

- Goal: a model *p(**e**|**f**,m)*
- where **e** and **f** are complete English and Foreign sentences

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

Lexical Translation

Lexical Translation

- Goal: a model $p(\boldsymbol{e}|\boldsymbol{f},m)$

Lexical Translation

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

Lexical Translation

- Goal: a model $p(\boldsymbol{e}|\boldsymbol{f},m)$
- where **e** and **f** are complete English and Foreign sentences
- Lexical translation makes the following ***assumptions***:

Lexical Translation

- Goal: a model $p(\mathbf{e}|\mathbf{f},m)$
- where **e** and **f** are complete English and Foreign sentences
- Lexical translation makes the following ***assumptions***:
  - Each word $\mathbf{e}_i$ in **e** is generated from exactly one word in **f**

Lexical Translation

- Goal: a model $p(\boldsymbol{e}|\boldsymbol{f},m)$
- where **e** and **f** are complete English and Foreign sentences
- Lexical translation makes the following ***assumptions***:
  - Each word $\mathbf{e}_i$ in **e** is generated from exactly one word in **f**
  - Thus, we have a latent *alignment* $\mathbf{a}_i$ that indicates which word $\mathbf{e}_i$ "came from." Specifically it came from $\mathbf{f}_{\mathbf{a}_i}$.

Lexical Translation

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

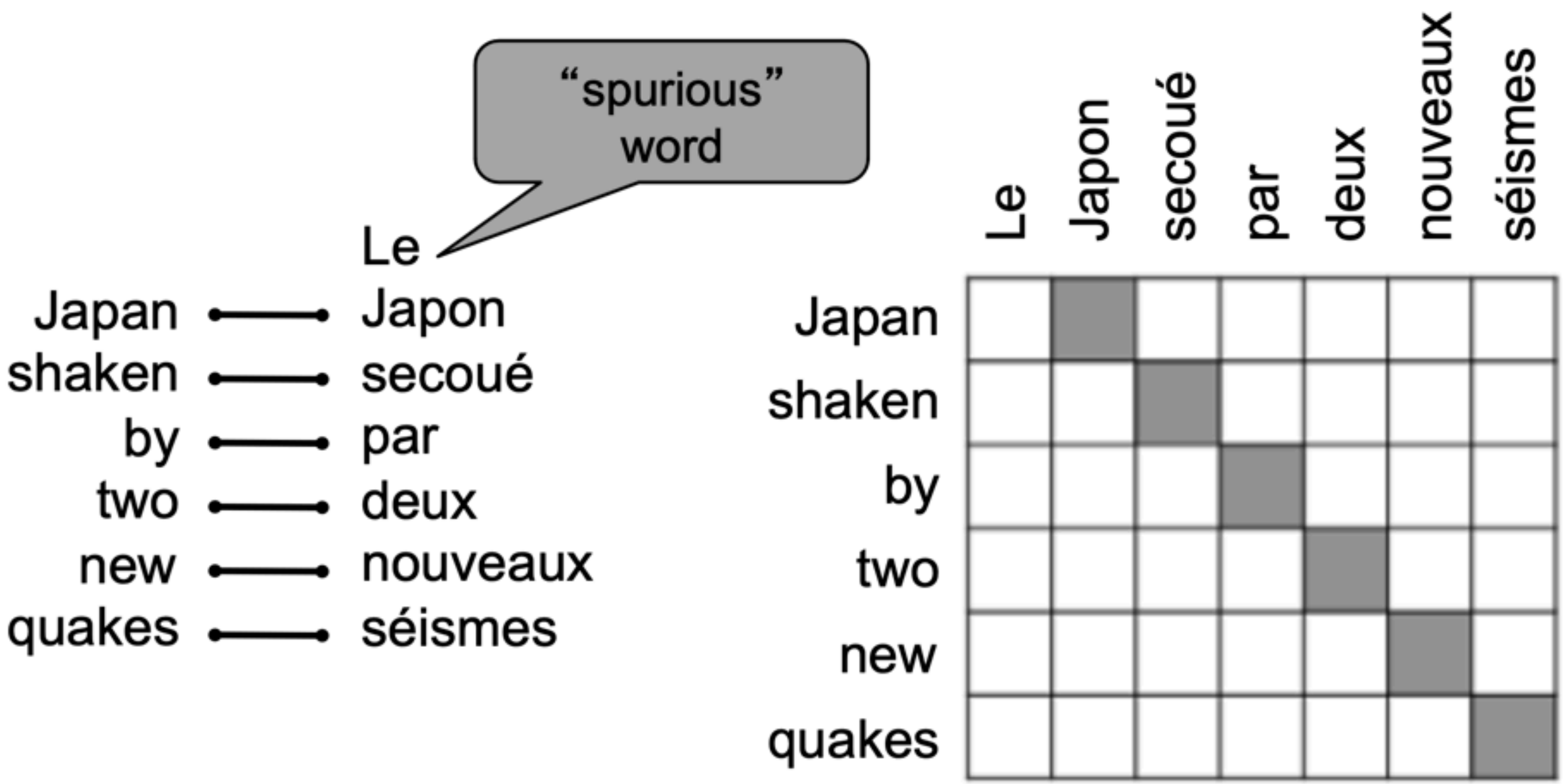
Lexical Translation

- Putting our assumptions together, we have:

$$p(\mathbf{e} \mid \mathbf{f}, m) = \underbrace{\sum_{\mathbf{a} \in [0,n]^m} p(\mathbf{a} \mid \mathbf{f}, m)}_{\text{p(Alignment)}} \times \underbrace{\prod_{i=1}^{m} p(e_i \mid f_{a_i})}_{\text{p(Translation | Alignment)}}$$

What is alignment?

- Alignment is the correspondence between particular words in the translated sentence pair.



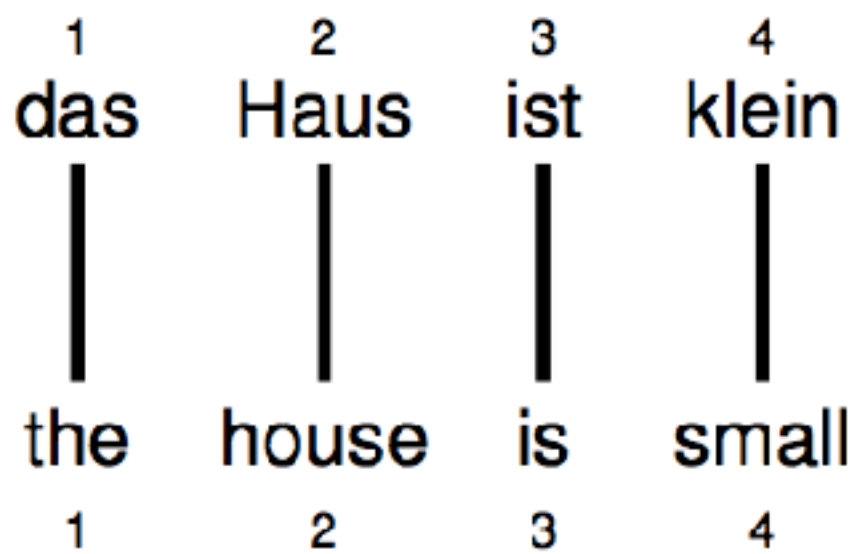
Alignment

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

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

Alignment

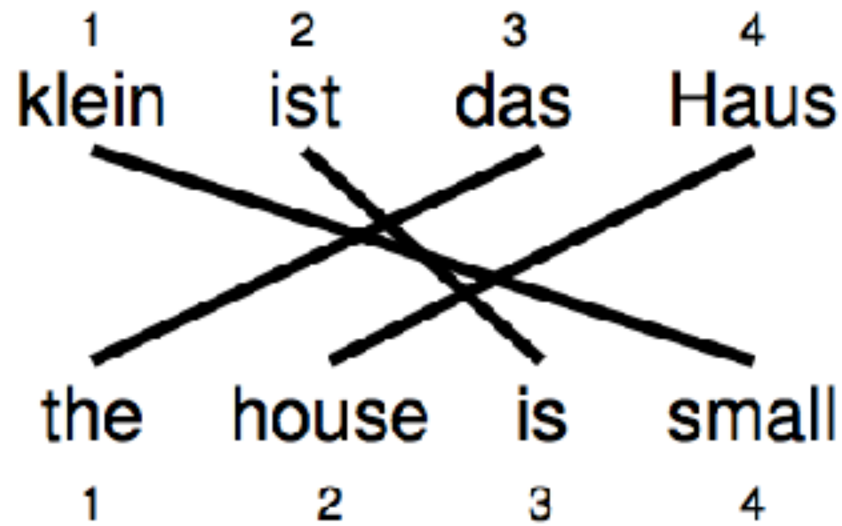
- Alignments can be visualized by drawing links between two sentences, and they are represented as vectors of positions:



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

Reordering

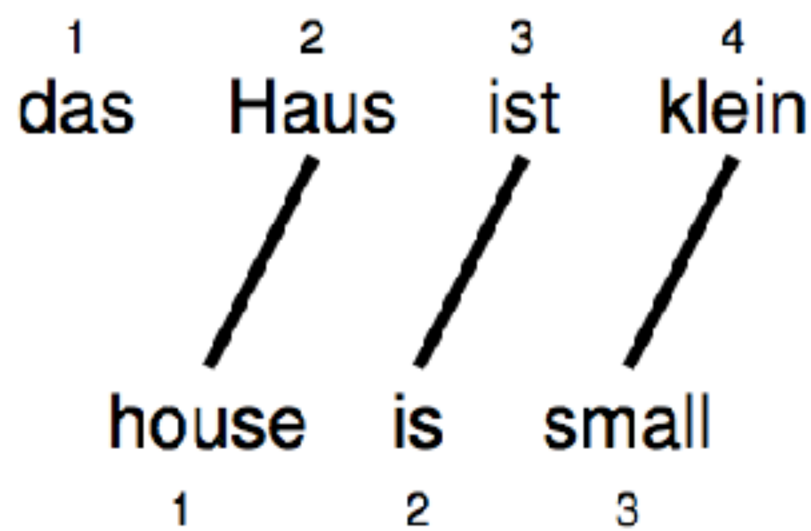
- Words may be reordered during translation



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

Word Dropping

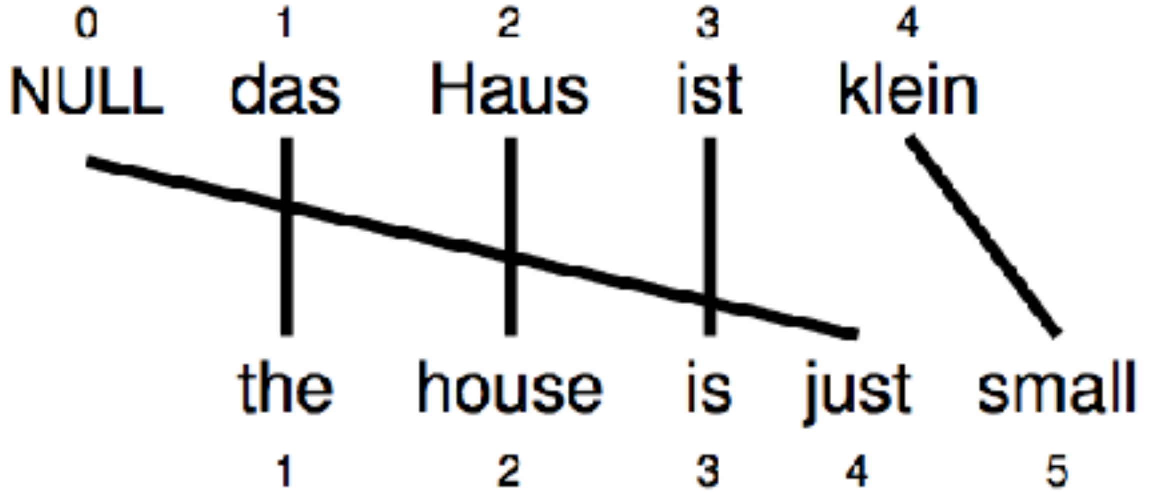
- A source word may not be translated at all



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

Word Insertion

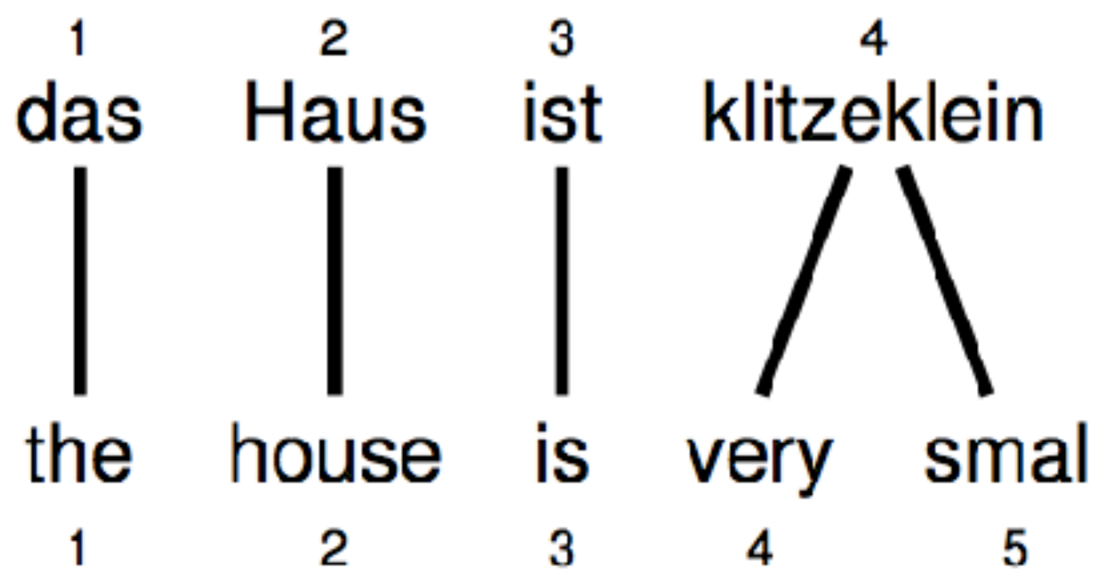
- Words may be inserted during translation
- E.g. English just does not have an equivalent
- But these words must be explained – we typically assume every source sentence contains a NULL token



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

One-to-many Translation

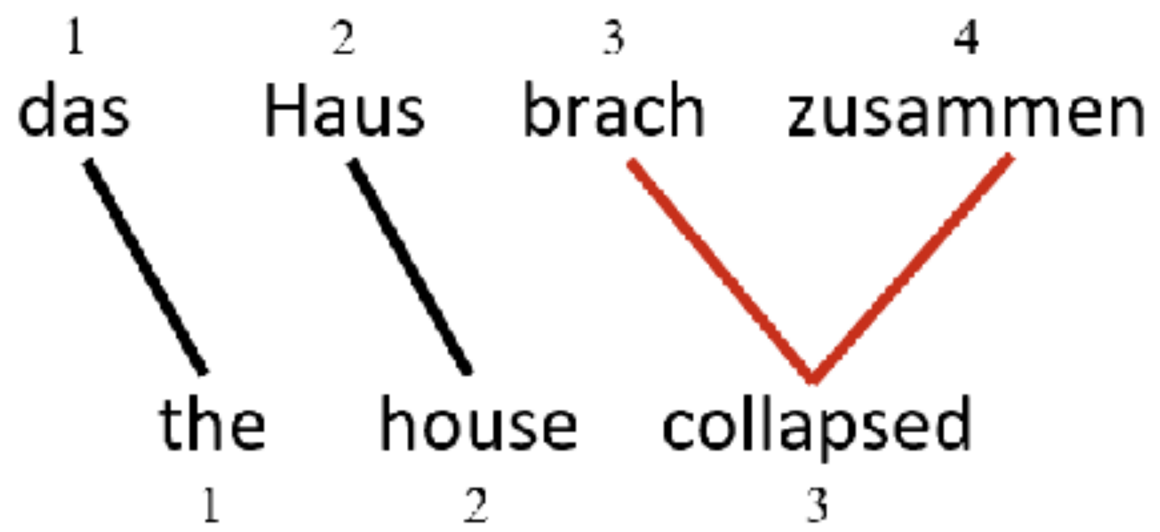
- A source word may translate into **more than one** target word



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

Many-to-one Translation

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



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

IBM Model 1

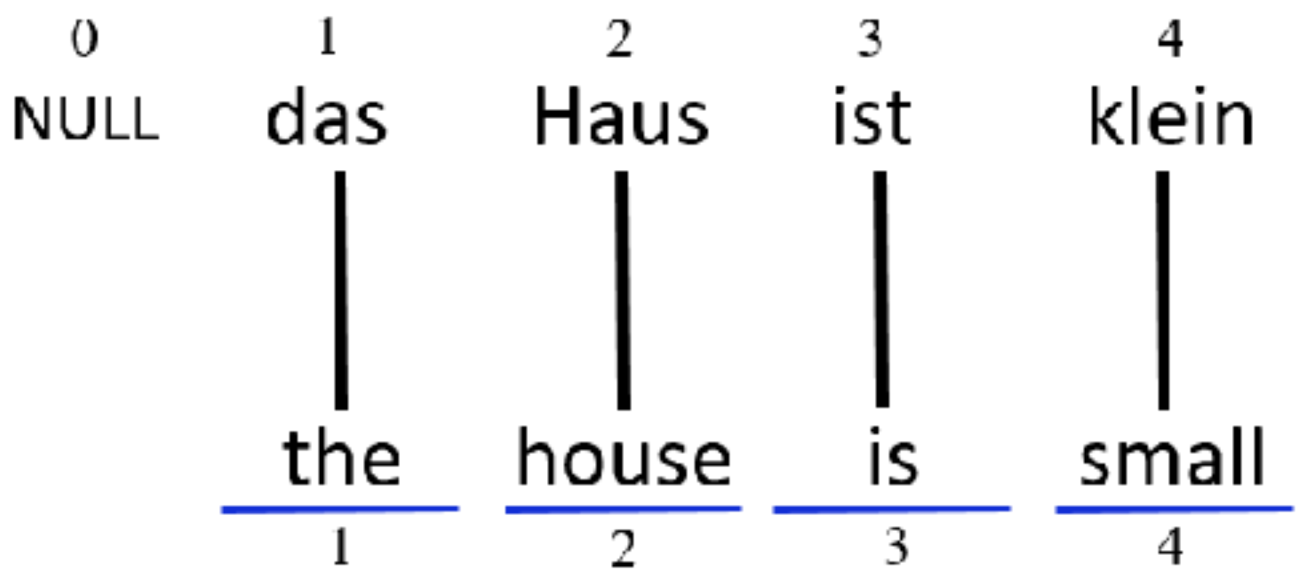
- Simplest possible lexical translation model
- Additional assumptions:
  - The *m* alignment decisions are independent
  - The alignment distribution for each $\mathbf{a}_i$ is uniform over all source words and NULL

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

Translating with Model 1

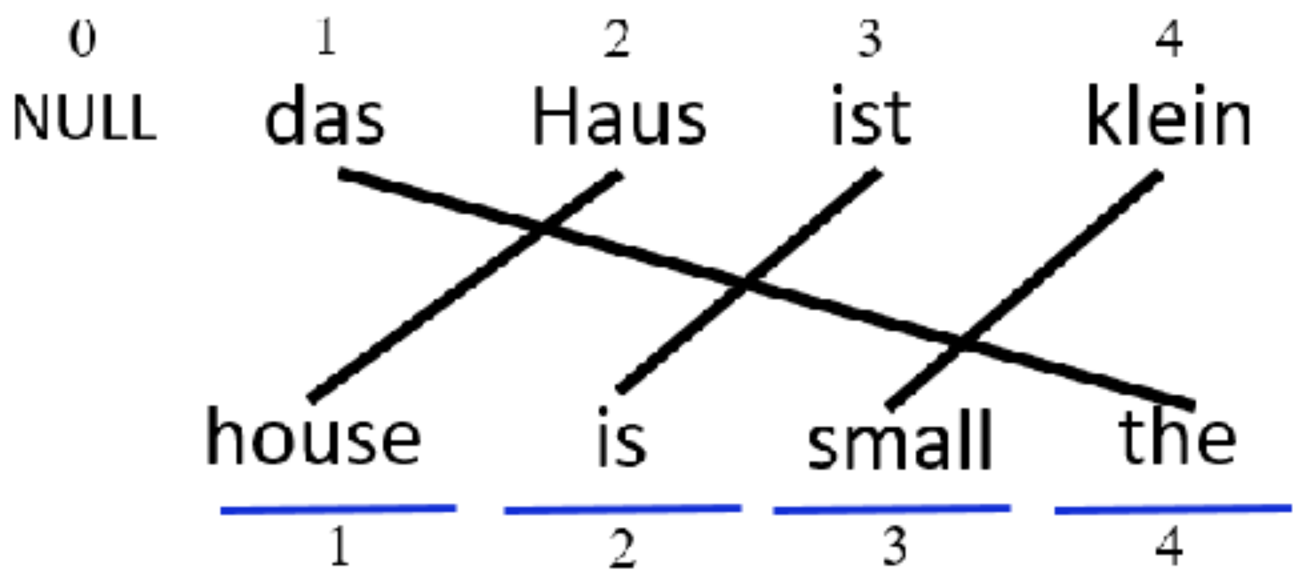
0	1	2	3	4
NULL	das	Haus	ist	klein
	___	___	___	___
	1	2	3	4

Translating with Model 1



Language model says: ☺

Translating with Model 1



Language model says: ☹

Learning Lexical Translation Models

Learning Lexical Translation Models

- How do we learn the parameters $p(e|f)$?

Learning Lexical Translation Models

- How do we learn the parameters $p(e|f)$?
- "Chicken and egg" problem

Learning Lexical Translation Models

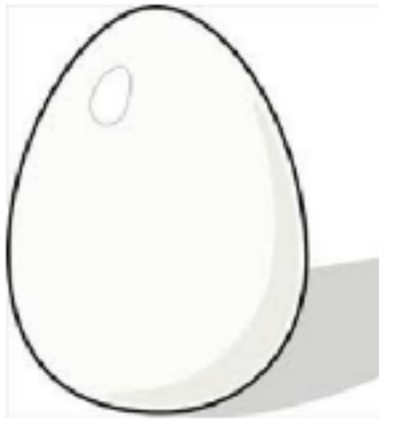
- How do we learn the parameters $p(e|f)$?
- "Chicken and egg" problem
  - If we had the alignments, we could estimate the translation probabilities (MLE estimation)

Learning Lexical Translation Models

- How do we learn the parameters $p(e|f)$?
- "Chicken and egg" problem
  - If we had the alignments, we could estimate the translation probabilities (MLE estimation)
  - If we had the translation probabilities we could find the most likely alignments (greedy)

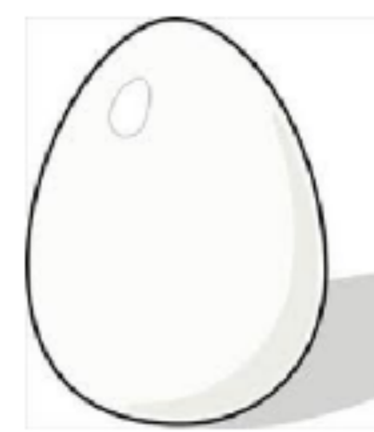
Learning Lexical Translation Models

- How do we learn the parameters $p(e|f)$?
- "Chicken and egg" problem
  - If we had the alignments, we could estimate the translation probabilities (MLE estimation)
  - If we had the translation probabilities we could find the most likely alignments (greedy)



Learning Lexical Translation Models

- How do we learn the parameters *p(e|f)*?
- “Chicken and egg” problem
  - If we had the alignments, we could estimate the translation probabilities (MLE estimation)
  - If we had the translation probabilities we could find the most likelv alignments (greedy)



EM Algorithm

EM Algorithm

- Pick some random (or uniform) starting parameters

EM Algorithm

- Pick some random (or uniform) starting parameters
- Repeat until bored (~5 iterations for lexical translation models):

EM Algorithm

- Pick some random (or uniform) starting parameters
- Repeat until bored (~5 iterations for lexical translation models):
  - Using the current parameters, compute "expected" alignments $p(\mathbf{a}_i|\mathbf{e}, \mathbf{f})$ for every target word token in the training data

EM Algorithm

- Pick some random (or uniform) starting parameters
- Repeat until bored (~5 iterations for lexical translation models):
  - Using the current parameters, compute "expected" alignments $p(\mathbf{a}_i|\mathbf{e}, \mathbf{f})$ for every target word token in the training data
  - Keep track of the expected number of times *f* translates into *e* throughout the whole corpus

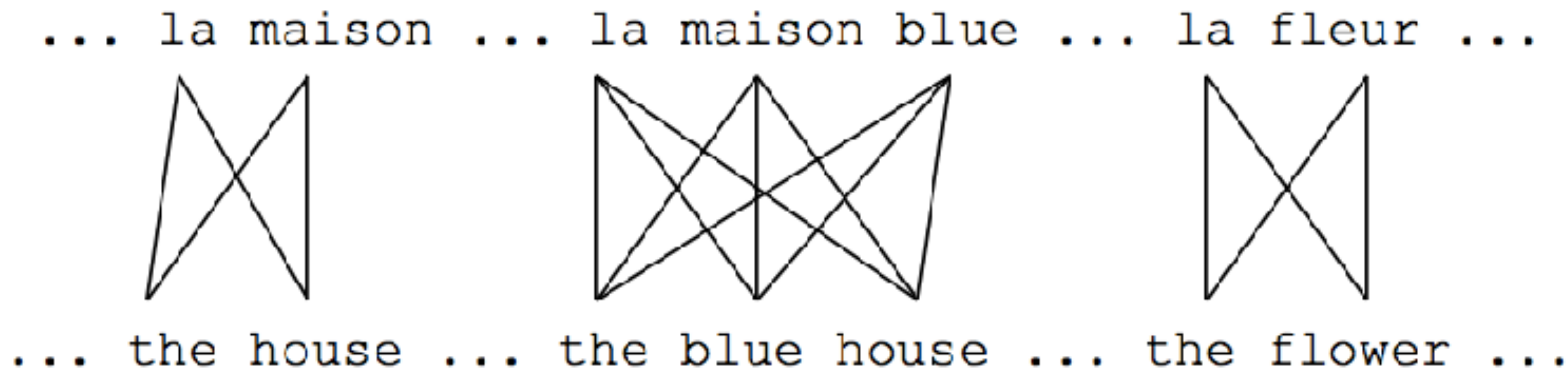
EM Algorithm

- Pick some random (or uniform) starting parameters
- Repeat until bored (~5 iterations for lexical translation models):
  - Using the current parameters, compute "expected" alignments $p(\mathbf{a}_i|\mathbf{e}, \mathbf{f})$ for every target word token in the training data
  - Keep track of the expected number of times *f* translates into *e* throughout the whole corpus
  - Keep track of the number of times *f* is used in the source of any translation

EM Algorithm

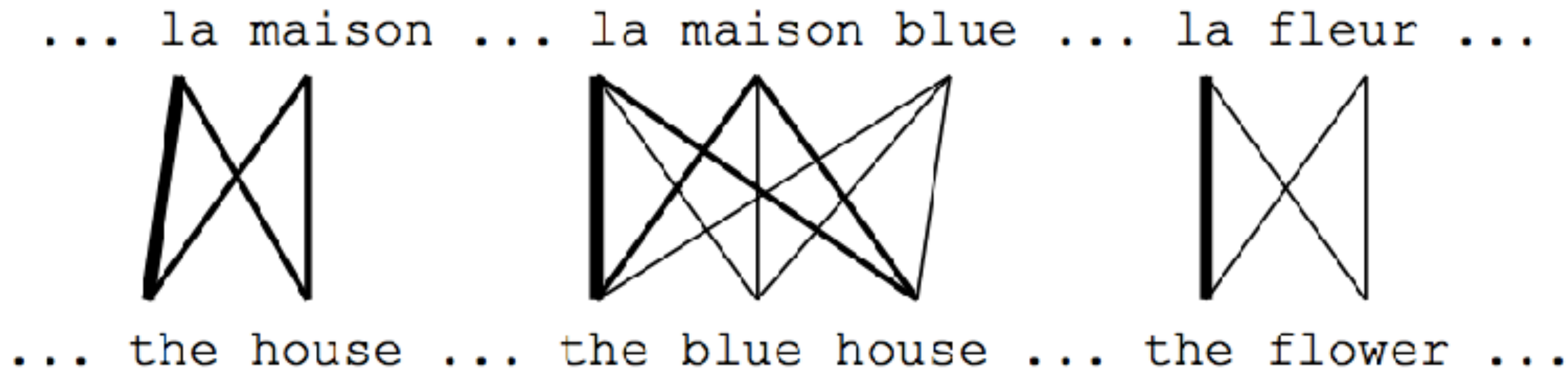
- Pick some random (or uniform) starting parameters
- Repeat until bored (~5 iterations for lexical translation models):
  - Using the current parameters, compute "expected" alignments $p(\mathbf{a}_i|\mathbf{e}, \mathbf{f})$ for every target word token in the training data
  - Keep track of the expected number of times *f* translates into *e* throughout the whole corpus
  - Keep track of the number of times *f* is used in the source of any translation
  - Use these estimates in the standard MLE equation to get a better set of parameters

EM for Model 1



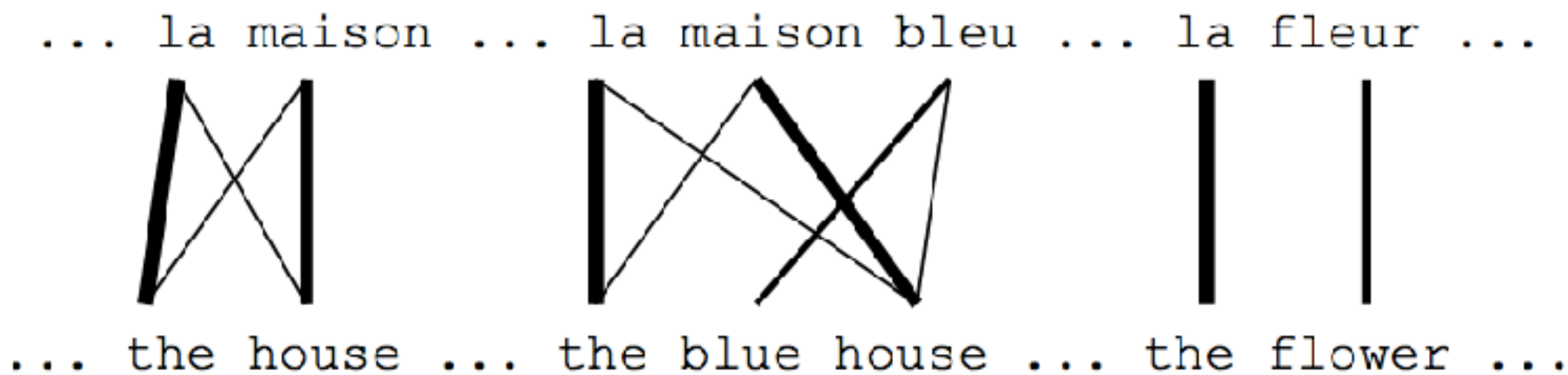
- Initial step: all alignments equally likely
- Model learns that, e.g., la is often aligned with the

EM for Model 1



- After one iteration
- Alignments, e.g., between la and the are more likely

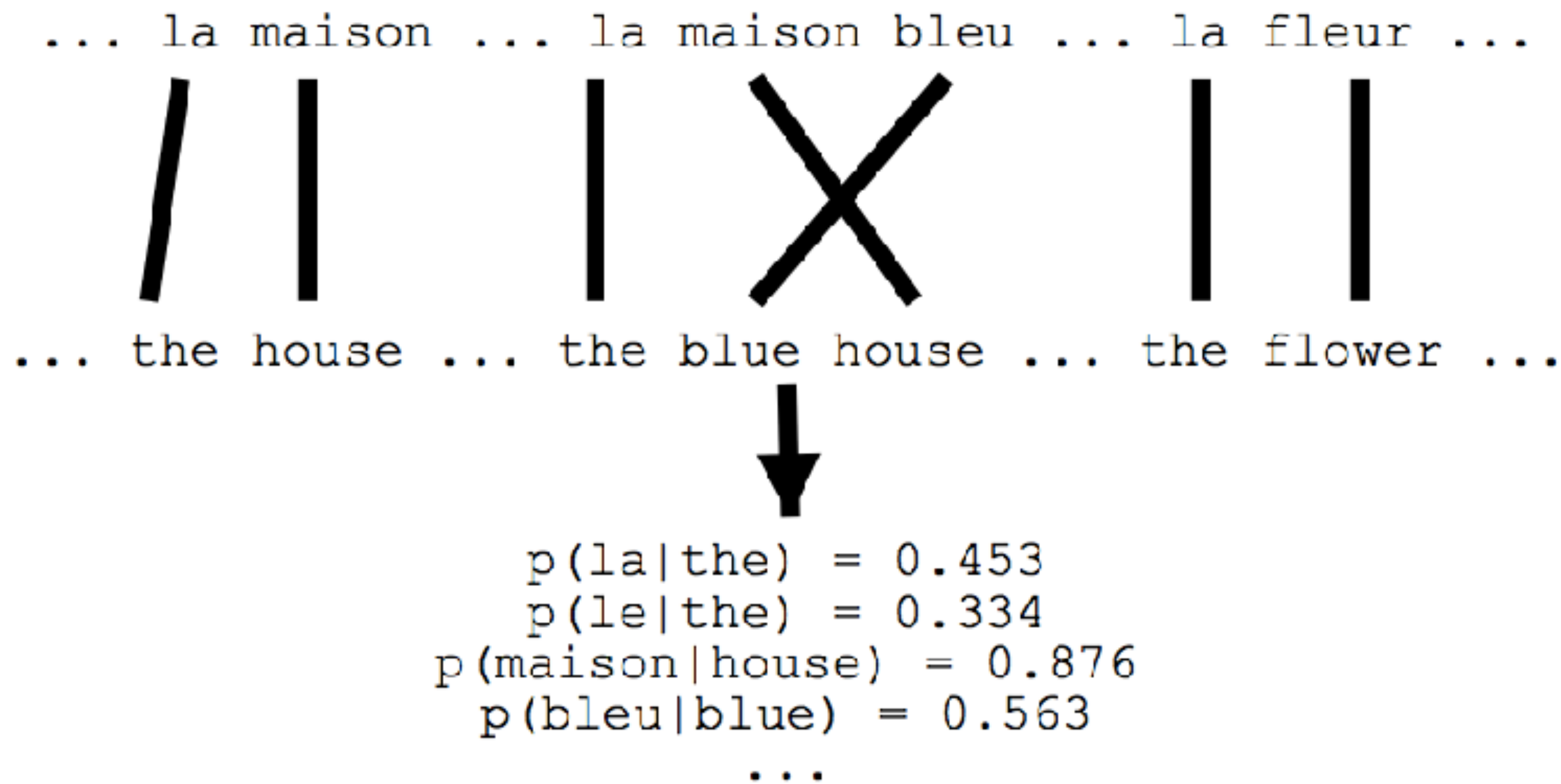
EM for Model 1



- After another iteration

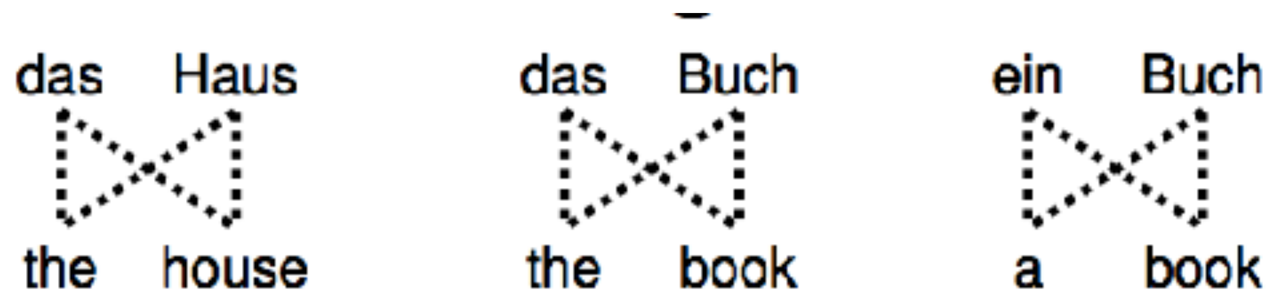
- It becomes apparent that alignments, e.g., between fleur and flower are more likely (pigeon hole principle)

EM for Model 1



- Parameter estimation from the aligned corpus

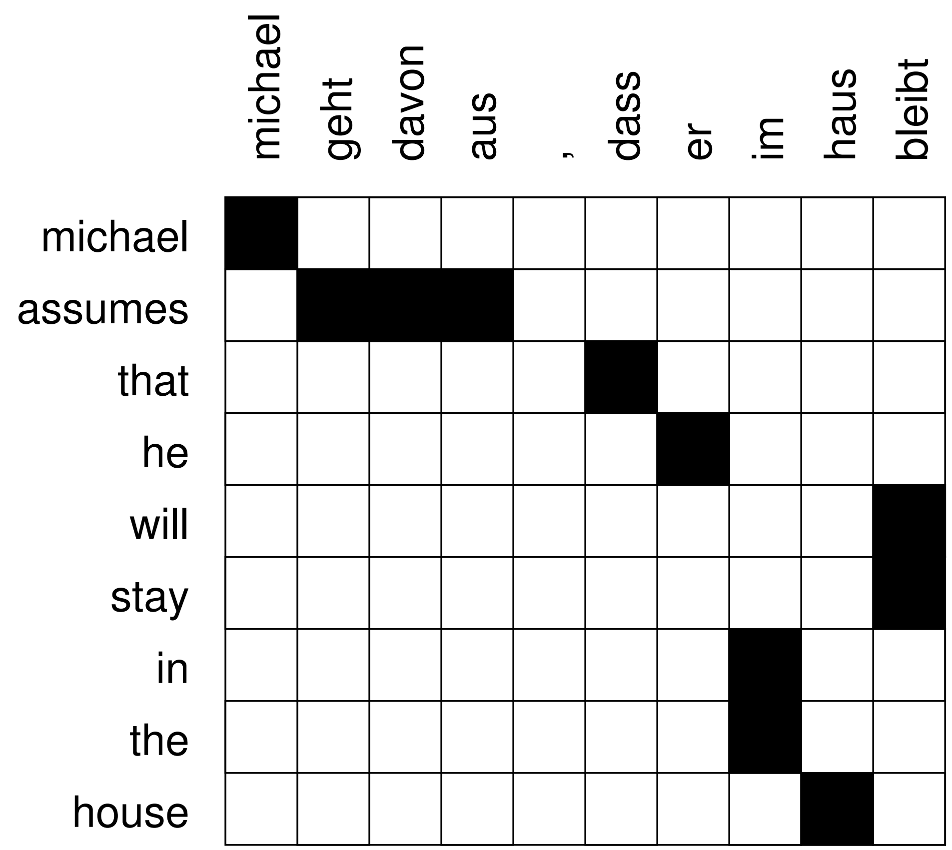
Convergence



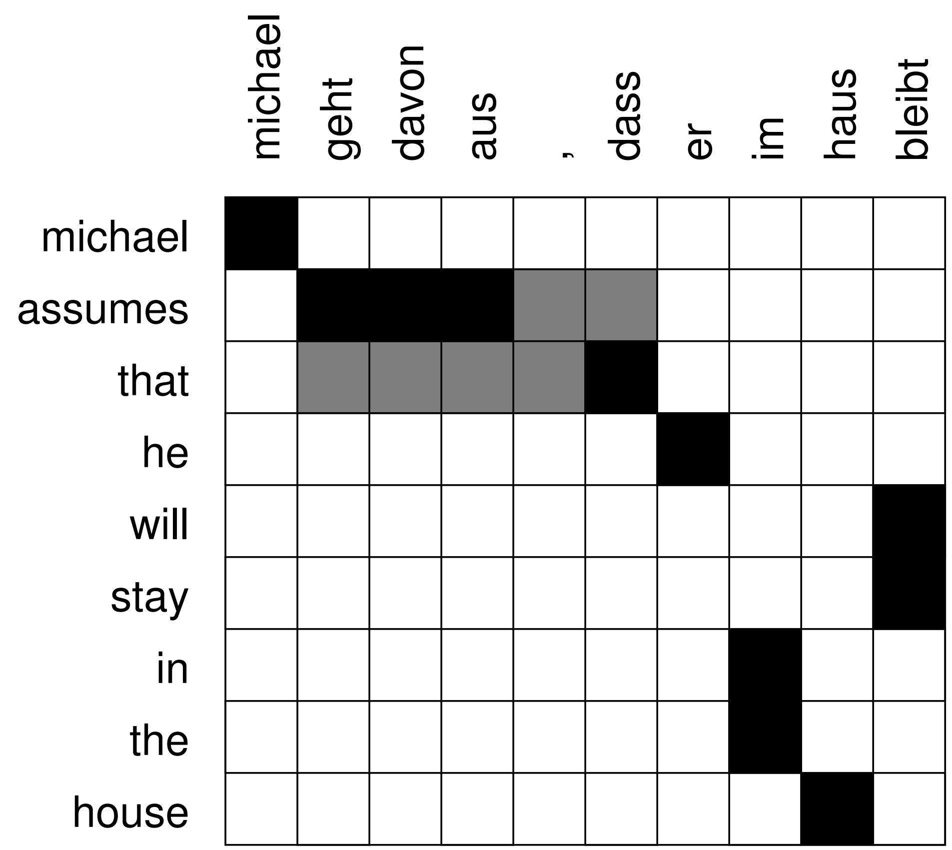
e	f	initial	1st it.	2nd it.	3rd it.	...	final
the	das	0.25	0.5	0.6364	0.7479	...	1
book	das	0.25	0.25	0.1818	0.1208	...	0
house	das	0.25	0.25	0.1818	0.1313	...	0
the	buch	0.25	0.25	0.1818	0.1208	...	0
book	buch	0.25	0.5	0.6364	0.7479	...	1
a	buch	0.25	0.25	0.1818	0.1313	...	0
book	ein	0.25	0.5	0.4286	0.3466	...	0
a	ein	0.25	0.5	0.5714	0.6534	...	1
the	haus	0.25	0.5	0.4286	0.3466	...	0
house	haus	0.25	0.5	0.5714	0.6534	...	1

From words to phrases

Word Alignment



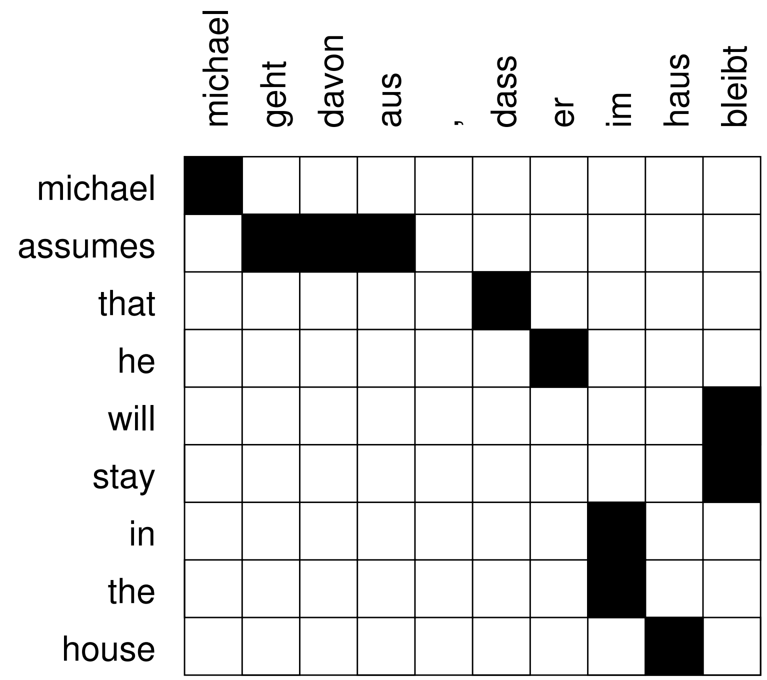
Extracting Phrase Pairs



extract phrase pair consistent with word alignment:

assumes that / geht davon aus , dass

Phrase Pair Extraction

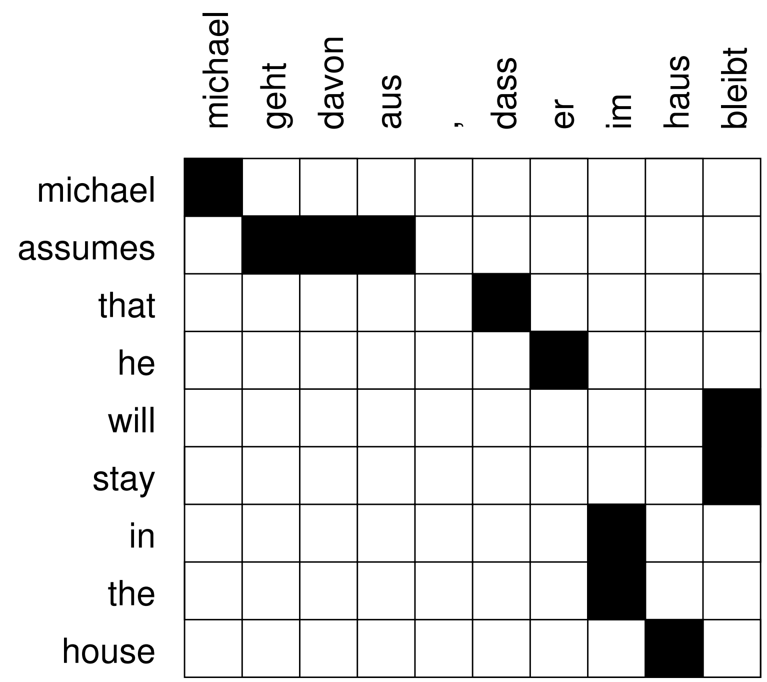


Smallest phrase pairs:

michael — michael
assumes — geht davon aus / geht davon aus ,
that — dass / , dass
he — er
will stay — bleibt
in the — im
house — haus

unaligned words (here: German comma) lead to multiple translations

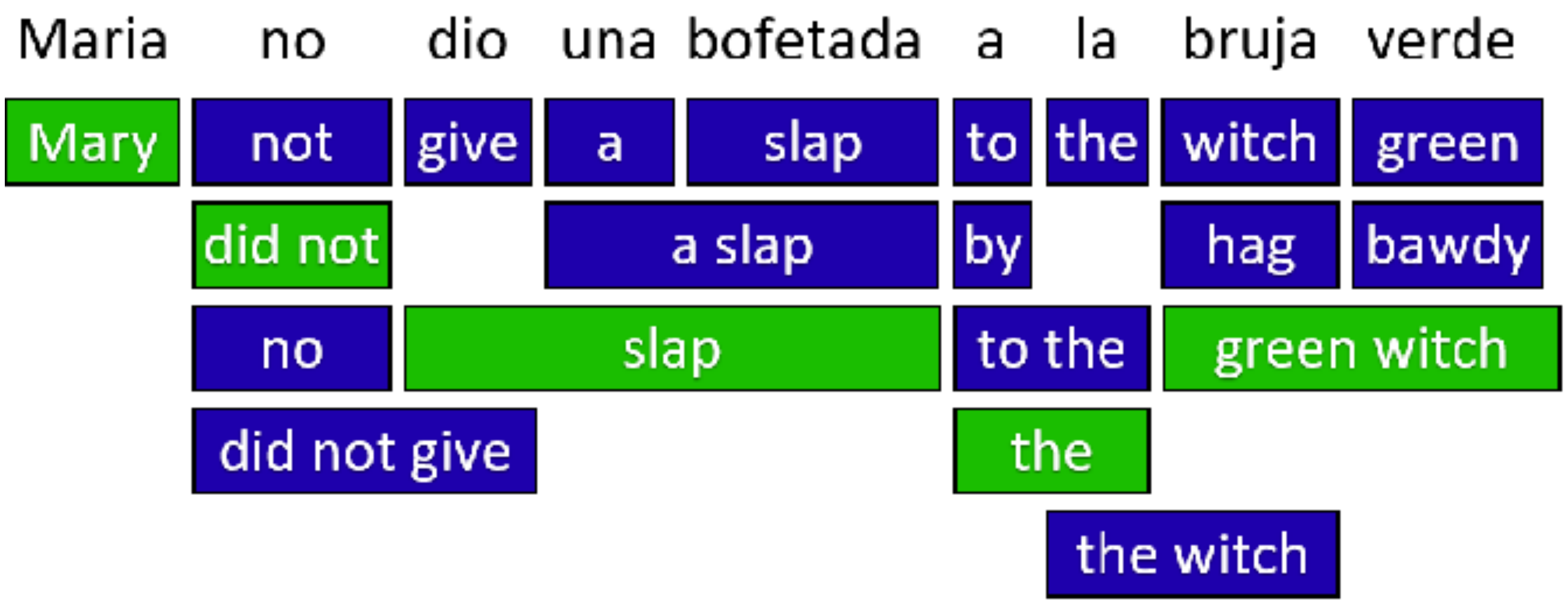
Larger Phrase Pairs



michael assumes — michael geht davon aus / michael geht davon aus ,
assumes that — geht davon aus , dass ; assumes that he — geht davon aus , dass er
that he — dass er / , dass er ; in the house — im haus
michael assumes that — michael geht davon aus , dass
michael assumes that he — michael geht davon aus , dass er
michael assumes that he will stay in the house — michael geht davon aus , dass er im haus bleibt
assumes that he will stay in the house — geht davon aus , dass er im haus bleibt
that he will stay in the house — dass er im haus bleibt ; dass er im haus bleibt ,
he will stay in the house — er im haus bleibt ; will stay in the house — im haus bleibt

Extensions

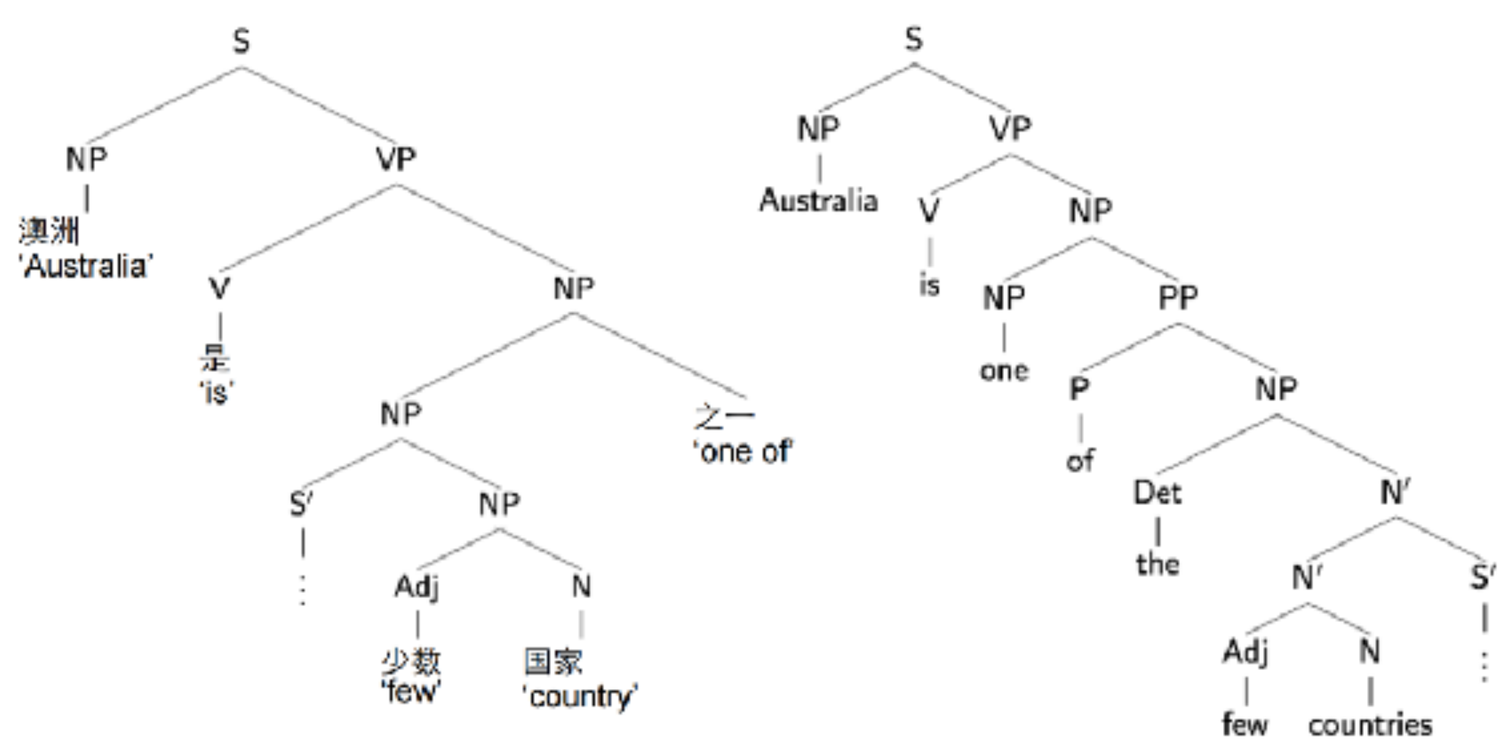
- Phrase-based MT:
  - Allow multiple words to translate as chunks (including many-to-one)
  - Introduce another latent variable, the source *segmentation*



Adapted from Koehn (2006)

Another Paradigm: Syntax-Based MT

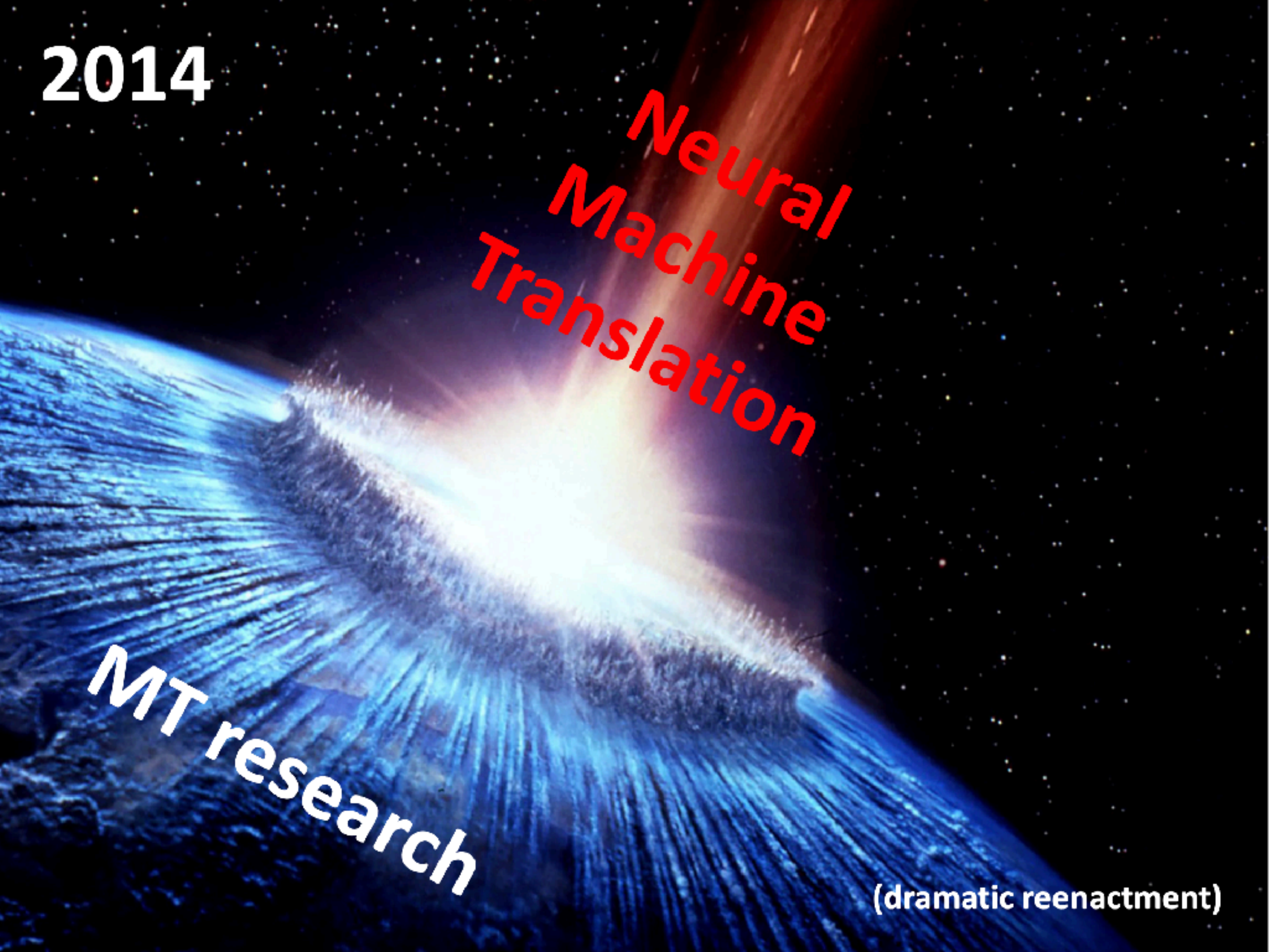
- Syntactic structure
- Rules of the form:
- X之一 → one of the X



Chang (2005),
Galley et al. (2006)

2014

(dramatic reenactment)


2014
Neural Machine Translation
MT research
(dramatic reenactment)

What is Neural Machine Translation?

- Neural Machine Translation (NMT) is a way to do Machine Translation with a single neural network
- The neural network architecture is called sequence-to-sequence (aka seq2seq) and it involves two RNNs.

Conditional Language Models

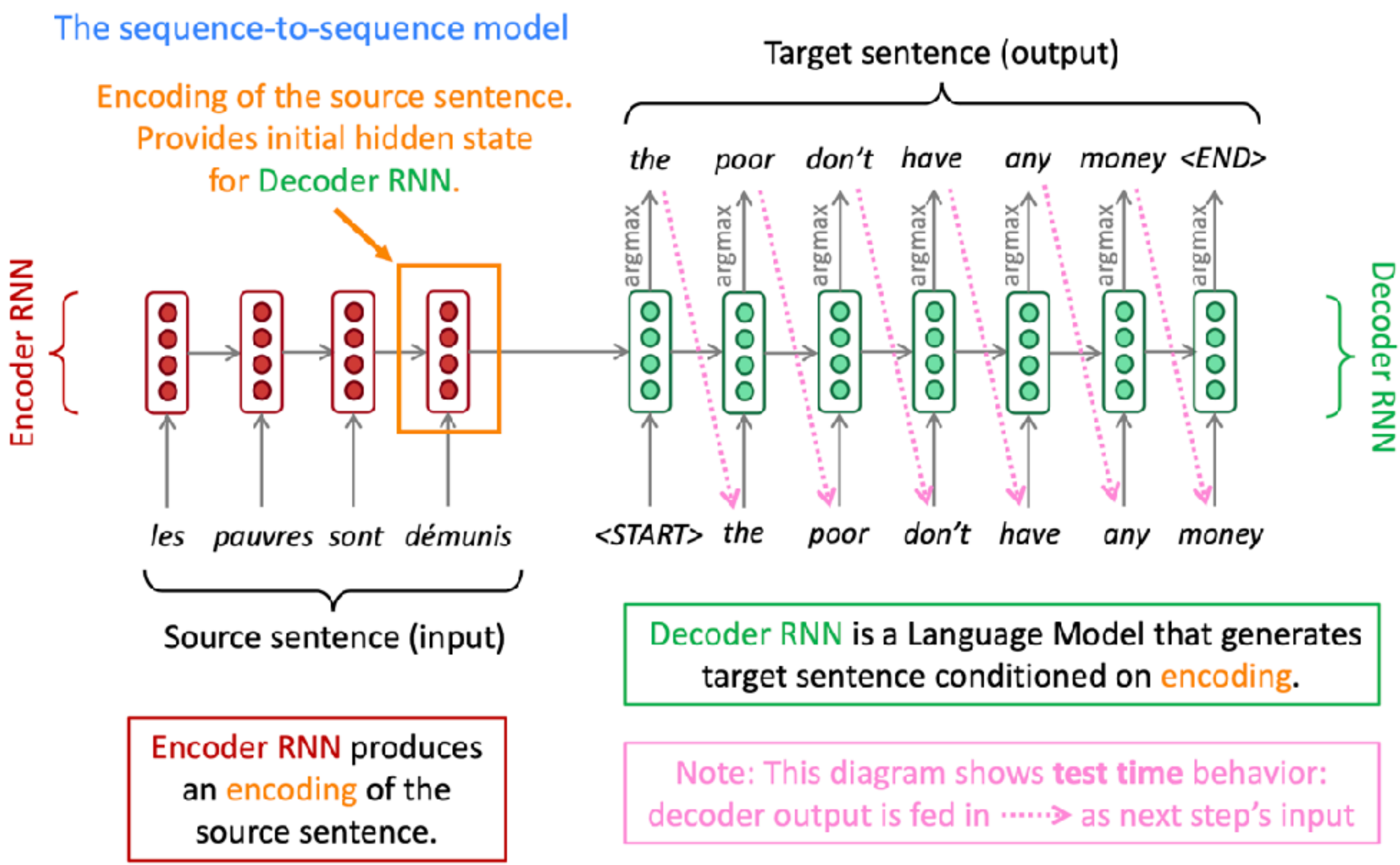
$$P(Y|X) = \prod_{j=1}^{J} P(y_j \mid X, y_1, \ldots, y_{j-1})$$

Conditional Language Models

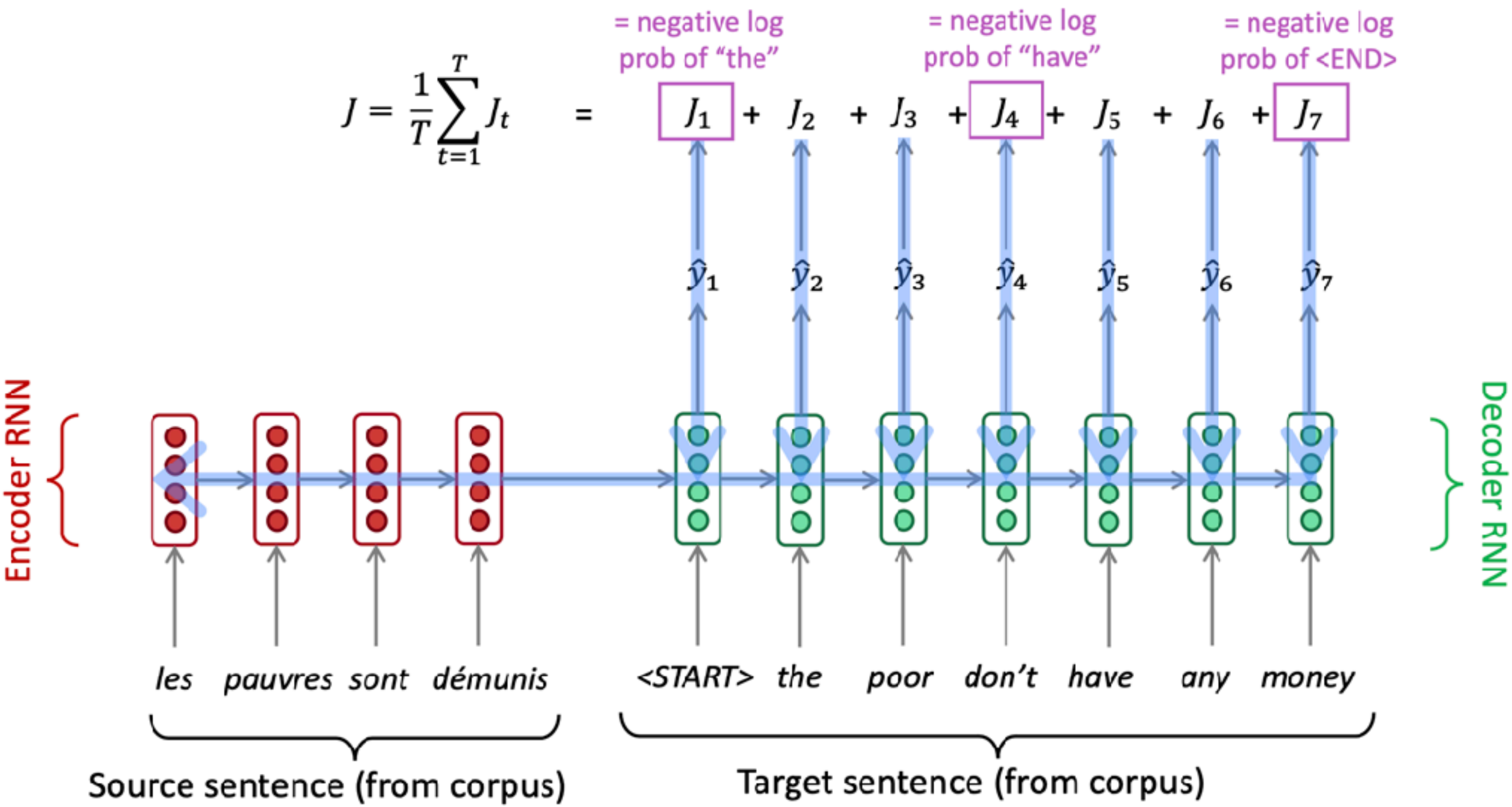
$$P(Y|X) = \prod_{j=1}^{J} P(y_j \mid \underline{X}, y_1, \ldots, y_{j-1})$$

Added Context!

Neural Machine Translation (NMT)



Neural Machine Translation (NMT)



Seq2seq is optimized as a **single system.**
Backpropagation operates *“end to end”*.

Advantages of NMT

Advantages of NMT

- Compared to SMT, NMT has many advantages:

Advantages of NMT

- Compared to SMT, NMT has many advantages:
- Better performance

Advantages of NMT

- Compared to SMT, NMT has many advantages:
- Better performance
  - More fluent

Advantages of NMT

- Compared to SMT, NMT has many advantages:
- Better performance
  - More fluent
  - Better use of context

Advantages of NMT

- Compared to SMT, NMT has many advantages:
- Better performance
  - More fluent
  - Better use of context
  - Better use of phrase similarities

Advantages of NMT

- Compared to SMT, NMT has many advantages:
- Better performance
  - More fluent
  - Better use of context
  - Better use of phrase similarities
- A single neural network to be optimized end-to-end

Advantages of NMT

- Compared to SMT, NMT has many advantages:
- Better performance
  - More fluent
  - Better use of context
  - Better use of phrase similarities
- A single neural network to be optimized end-to-end
  - No subcomponents to be individually optimized

Advantages of NMT

- Compared to SMT, NMT has many advantages:
- Better performance
  - More fluent
  - Better use of context
  - Better use of phrase similarities
- A single neural network to be optimized end-to-end
  - No subcomponents to be individually optimized
- Requires much less human engineering effort

Advantages of NMT

- Compared to SMT, NMT has many advantages:
- Better performance
  - More fluent
  - Better use of context
  - Better use of phrase similarities
- A single neural network to be optimized end-to-end
  - No subcomponents to be individually optimized
- Requires much less human engineering effort
  - No feature engineering

Advantages of NMT

- Compared to SMT, NMT has many advantages:
- Better performance
  - More fluent
  - Better use of context
  - Better use of phrase similarities
- A single neural network to be optimized end-to-end
  - No subcomponents to be individually optimized
- Requires much less human engineering effort
  - No feature engineering
  - Same method for all language pairs

Disadvantages of NMT?

Disadvantages of NMT?

Compared to SMT:

Disadvantages of NMT?

Compared to SMT:

- NMT is less interpretable

Disadvantages of NMT?

Compared to SMT:

- NMT is less interpretable
  - Hard to debug

Disadvantages of NMT?

Compared to SMT:

- NMT is less interpretable
  - Hard to debug
- NMT is difficult to control

Disadvantages of NMT?

Compared to SMT:

- NMT is less interpretable
  - Hard to debug
- NMT is difficult to control
  - For example, can’t easily specify rules or guidelines for translation

Disadvantages of NMT?

Compared to SMT:

- NMT is less interpretable
  - Hard to debug
- NMT is difficult to control
  - For example, can’t easily specify rules or guidelines for translation
  - Safety concerns!

Generation

Can we find the best (most likely) translation?

Generation through Sampling

No but we can approximate it!

Generating New Sentences

Generating New Sentences

- Generate sentences:

while didn't choose end-of-sentence symbol:
  calculate probability of

$$P(x_t \mid x_1, \ldots, x_{t-1})$$

Greedy Decoding

- Generate next word conditioned on the context (i.e., the previously generated words)

- "Greedy": always pick the most probable next word
$$x_t = argmax_{\hat{x}} P(\hat{x} \mid x_1, \ldots, x_{t-1})$$

Greedy Decoding

- Generate next word conditioned on the context (i.e., the previously generated words)

- "Greedy": always pick the most probable next word

$$x_t = argmax_{\hat{x}} P(\hat{x} \mid x_1, \ldots, x_{t-1})$$

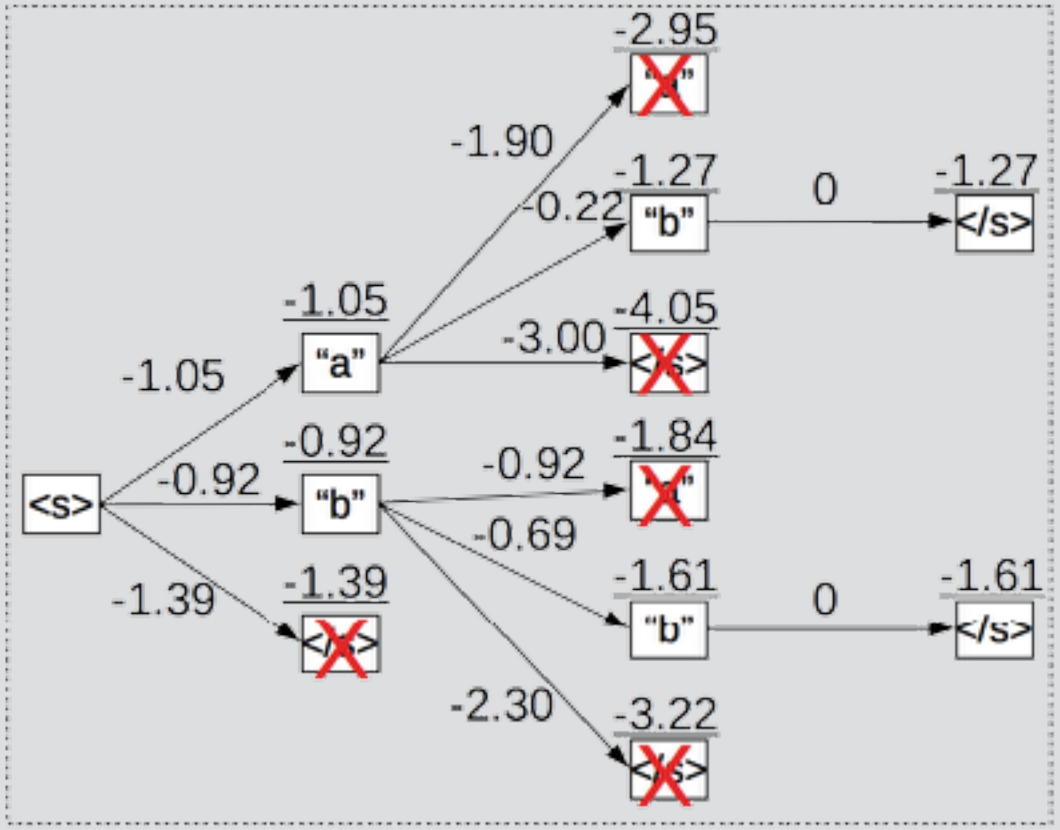
- Problem:
  - The most probable next word does not always lead to the most probable sentence;
  - We should be able to generate a diverse set of sentences!

Beam Search

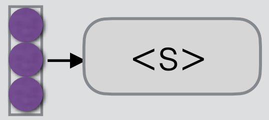
- Beam search: instead of picking one high-probability word, maintain several paths

Beam Search

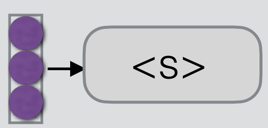
- Beam search: instead of picking one high-probability word, maintain several paths



Beam Search



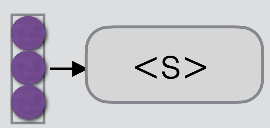
Beam Search



a	0.001
the	0.0002
I	0.12
you	0.04
cat	0.0004
movie	0.01
this	0.02
…	

Beam Search

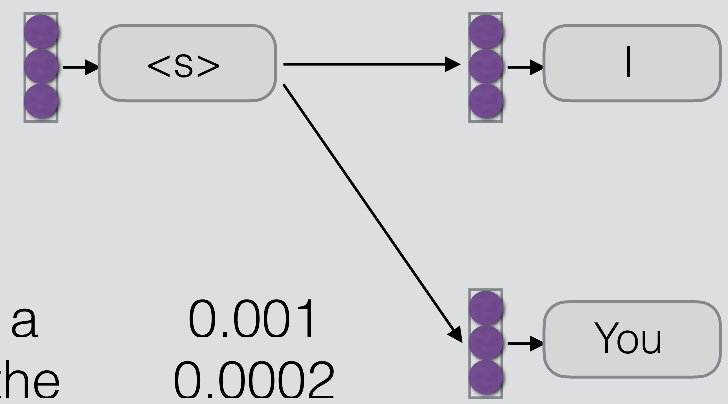
k=2



a	0.001
the	0.0002
I	0.12
you	0.04
cat	0.0004
movie	0.01
this	0.02
…	

Beam Search

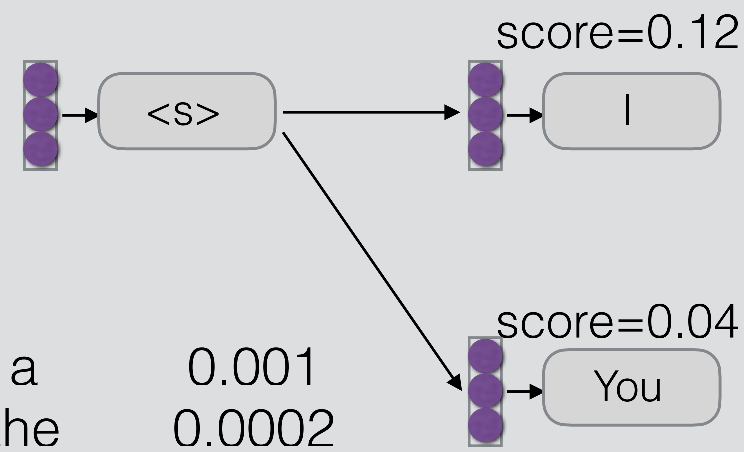
k=2



a	0.001
the	0.0002
I	0.12
you	0.04
cat	0.0004
movie	0.01
this	0.02
…	

Beam Search

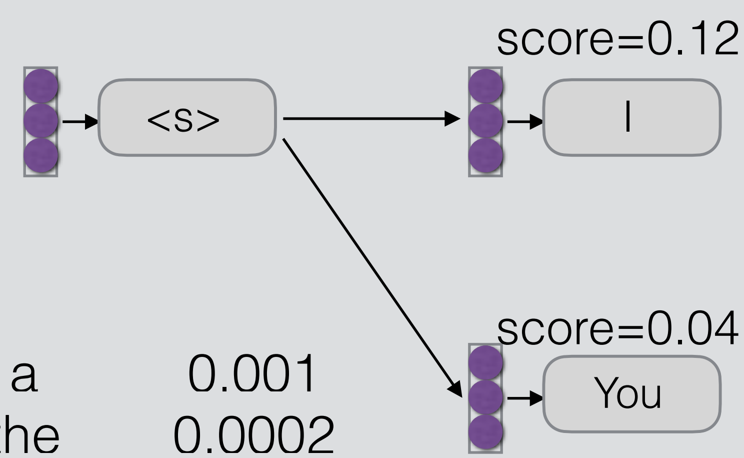
k=2



a	0.001
the	0.0002
I	0.12
you	0.04
cat	0.0004
movie	0.01
this	0.02
…	

Beam Search

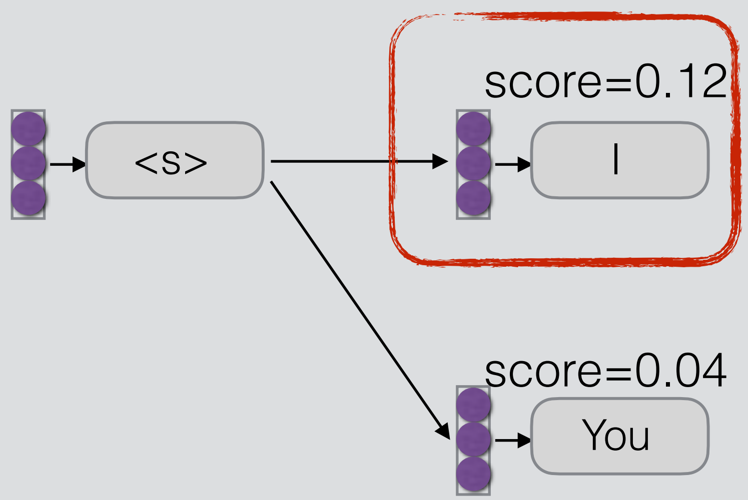
k=2 Expand



a	0.001
the	0.0002
I	0.12
you	0.04
cat	0.0004
movie	0.01
this	0.02
…	

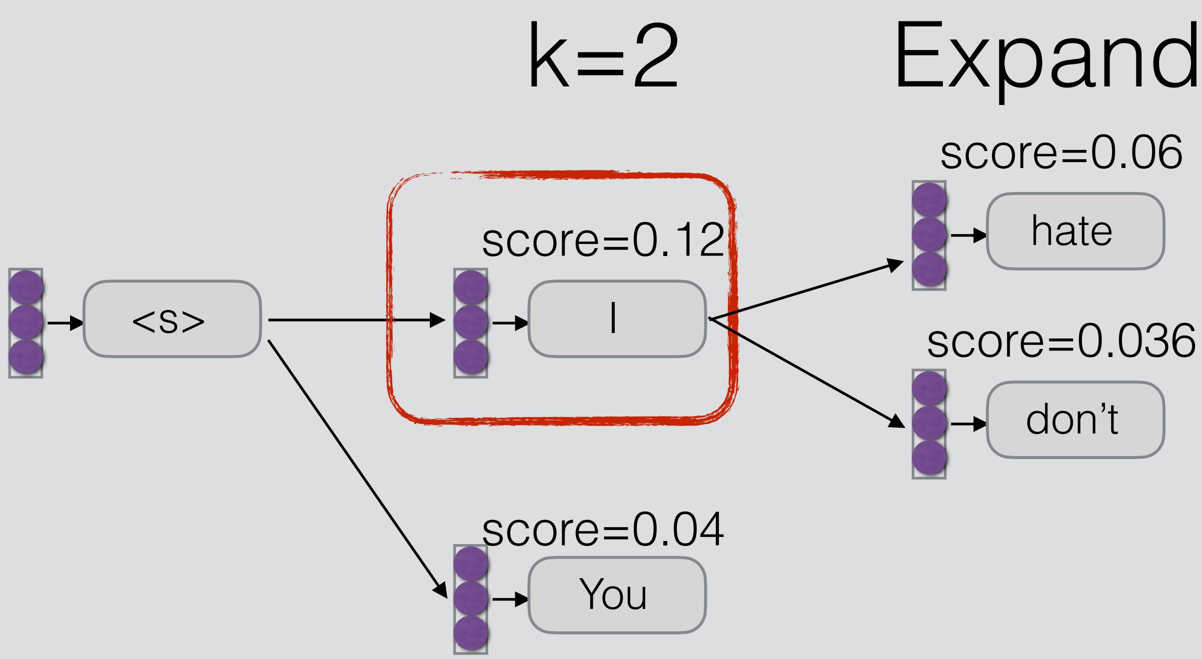
Beam Search

k=2 Expand



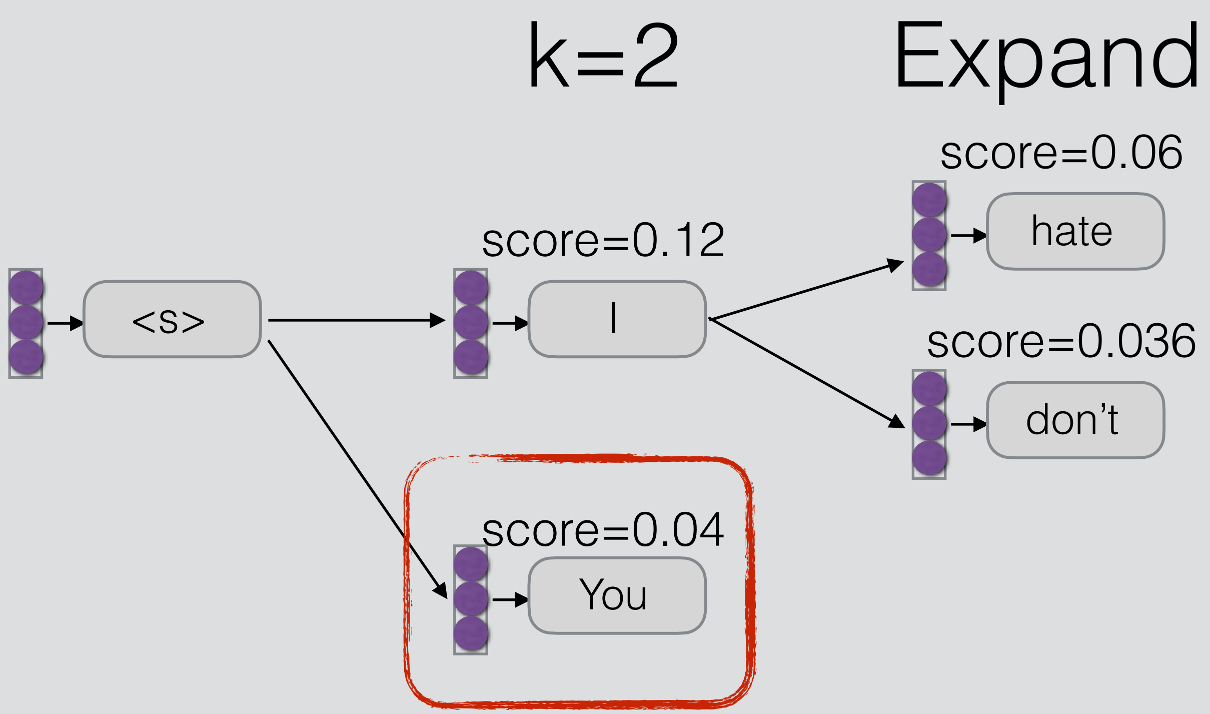
a	0.001
the	0.0002
hate	0.5
this	0.001
cat	0.003
movie	0.07
don’t	0.3
…	…

Beam Search



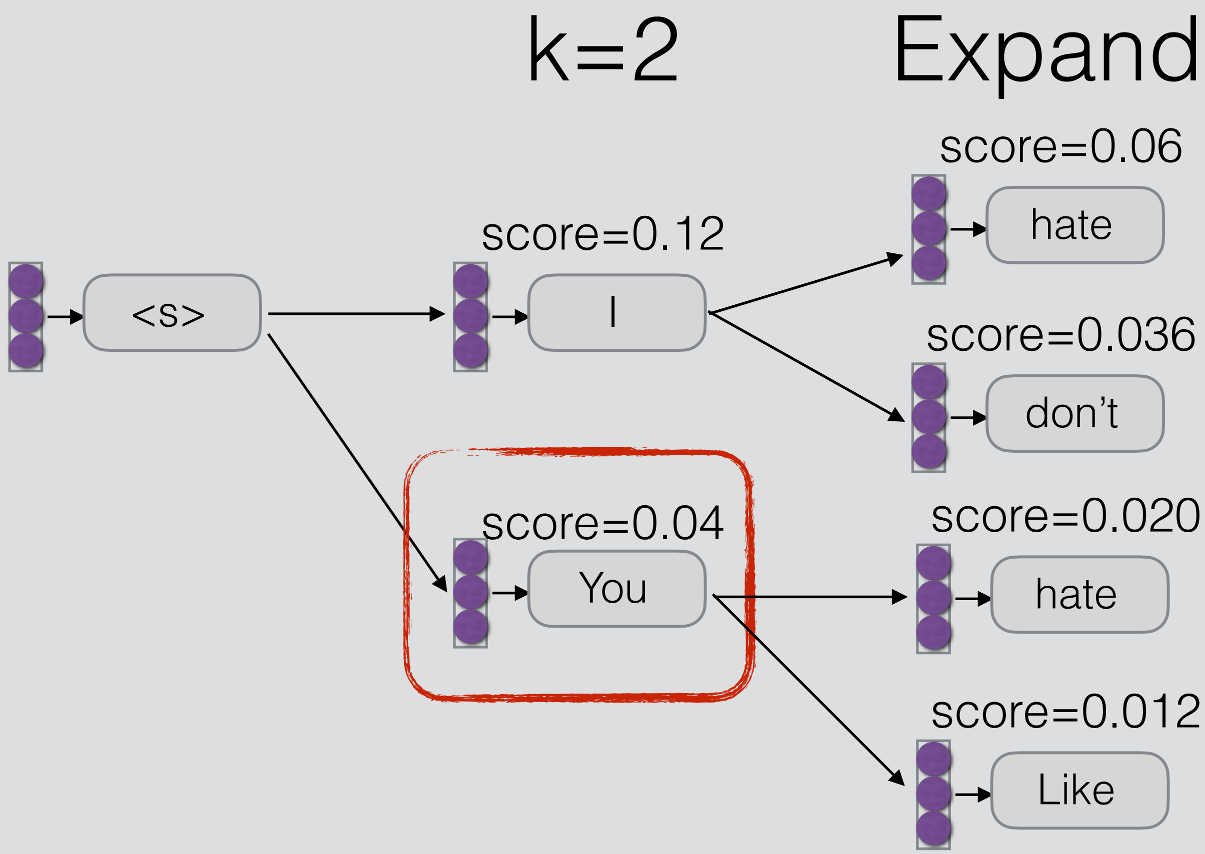
a	0.001
the	0.0002
hate	0.5
this	0.001
cat	0.003
movie	0.07
don’t	0.3
…	…

Beam Search



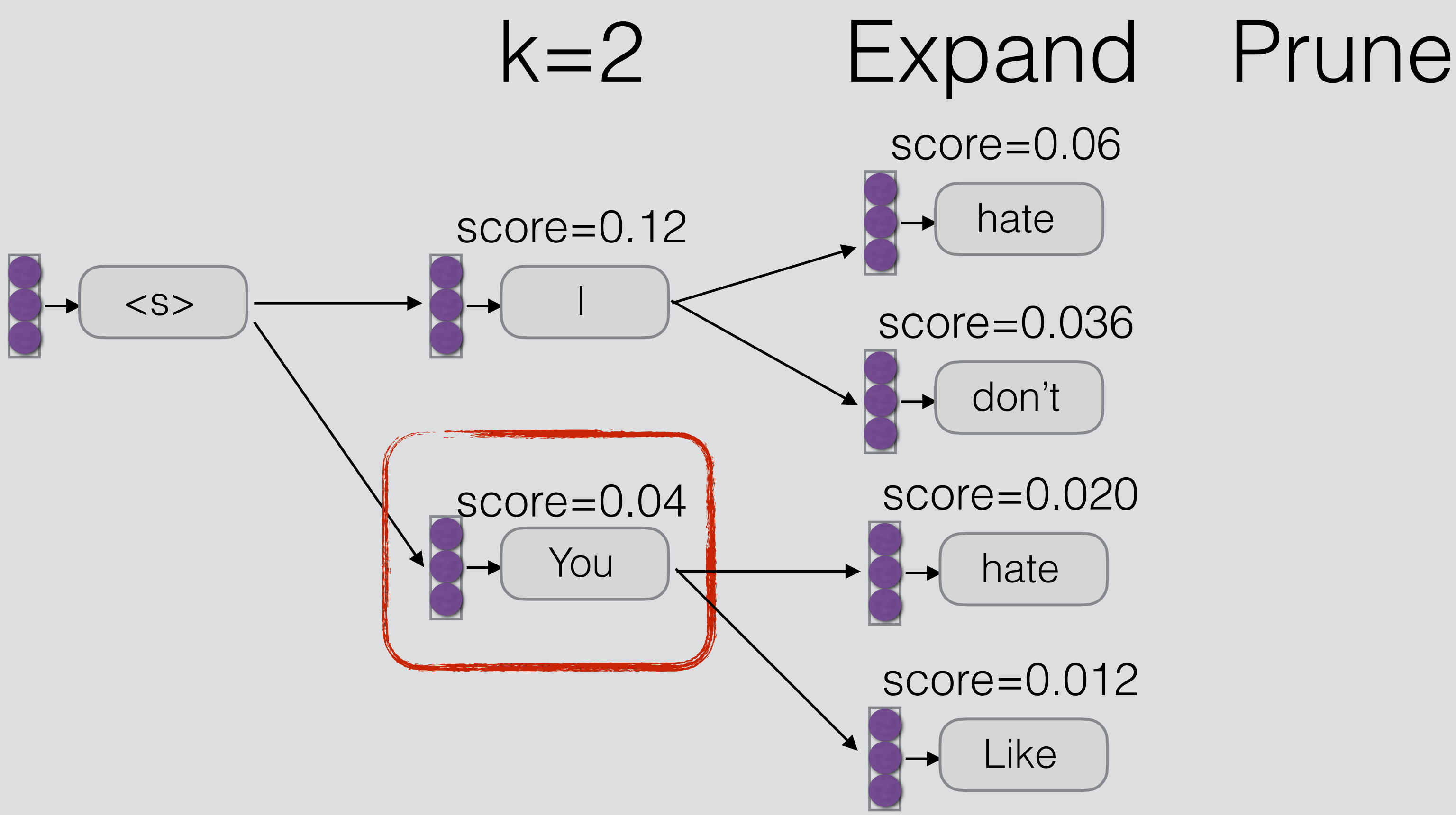
an	0.0012
be	0.0002
hate	0.5
these	0.001
dog	0.003
movie	0.07
like	0.3
…	…

Beam Search



an	0.0012
be	0.0002
hate	0.5
these	0.001
dog	0.003
movie	0.07
like	0.3
…	…

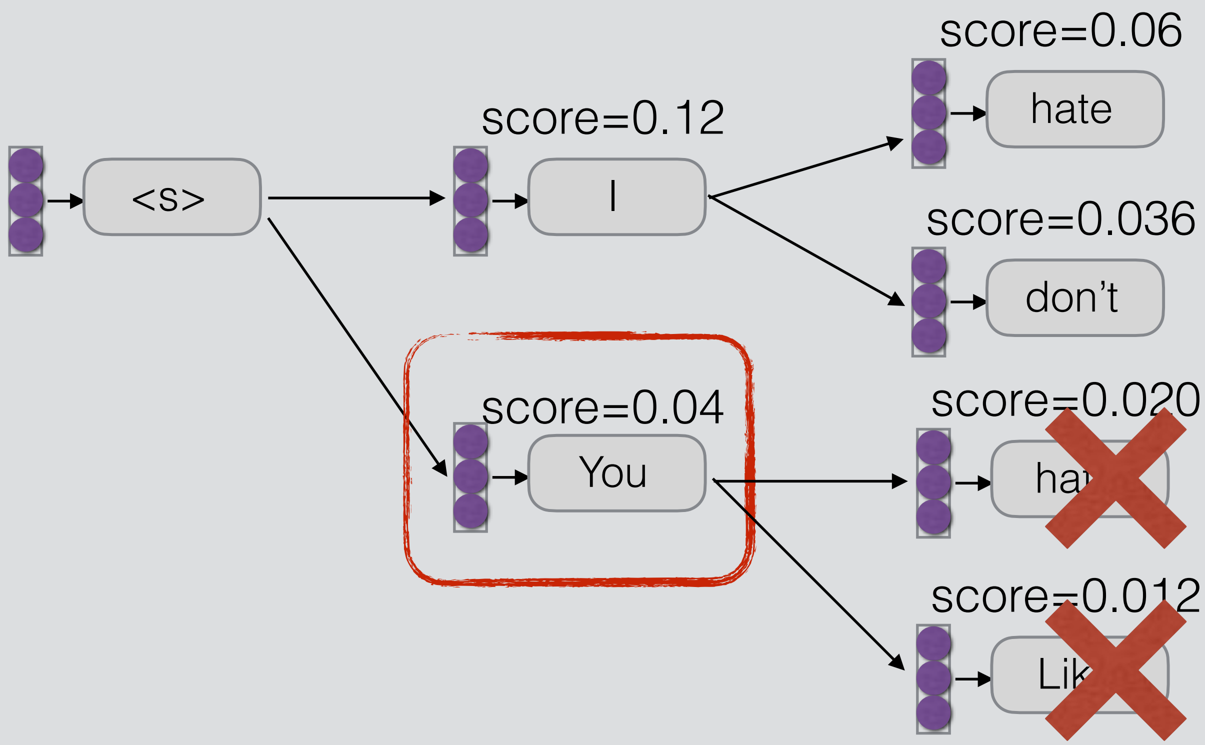
Beam Search



an	0.0012
be	0.0002
hate	0.5 ←
these	0.001
dog	0.003
movie	0.07
like	0.3 ←
…	…

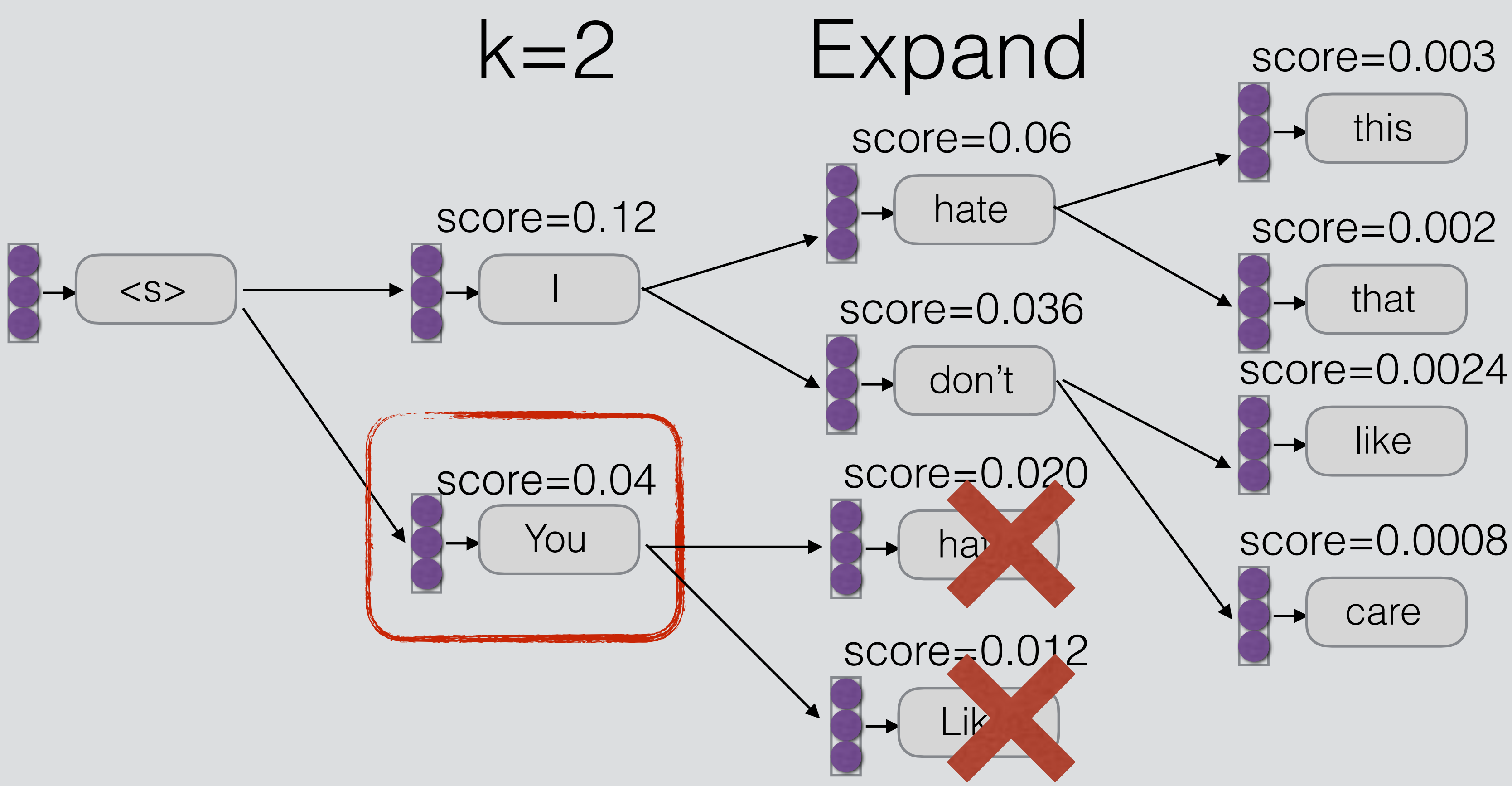
Beam Search

k=2 Expand Prune



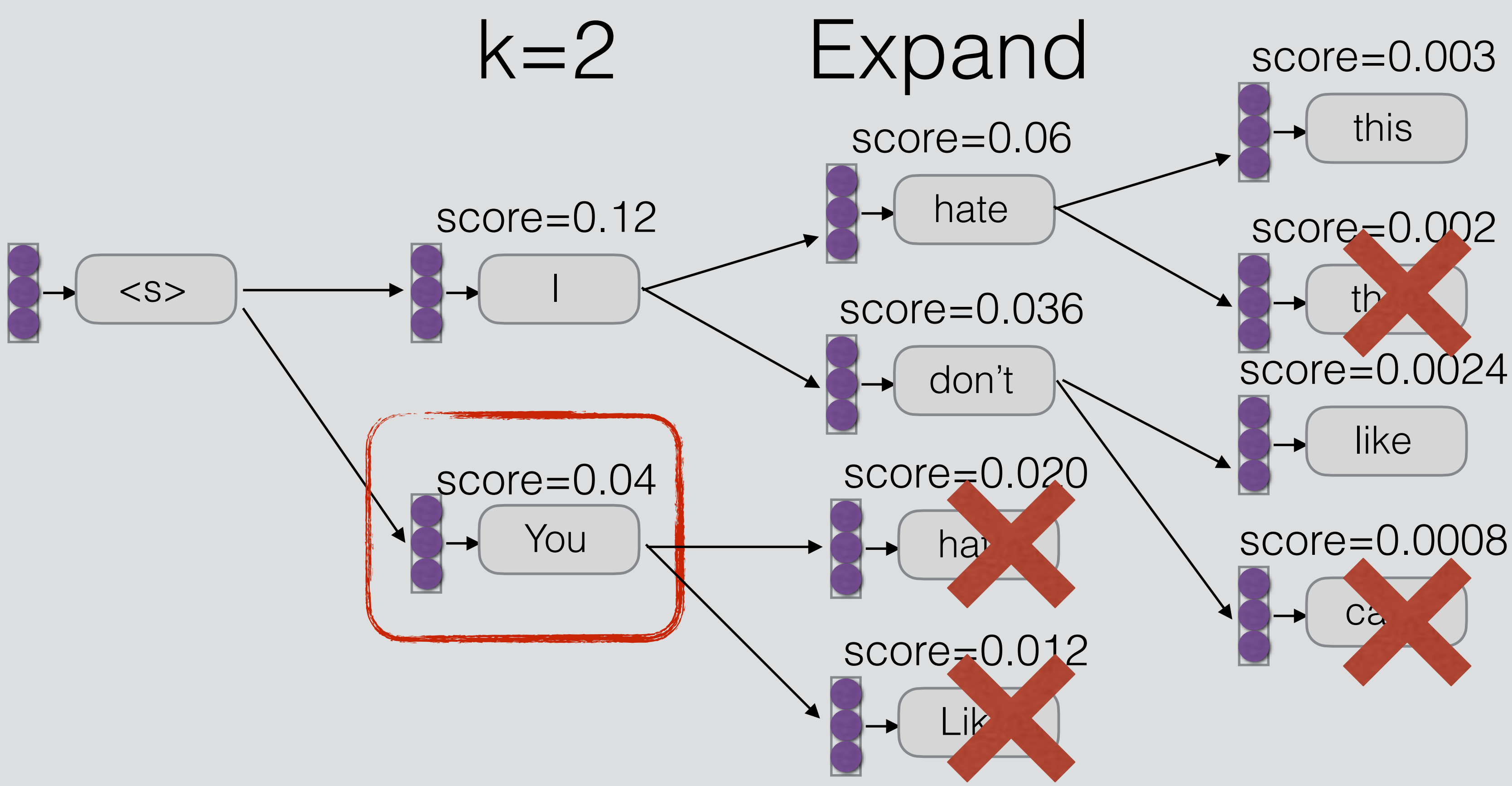
an	0.0012
be	0.0002
hate	0.5
these	0.001
dog	0.003
movie	0.07
like	0.3
…	…

Beam Search



k=2
Expand
<s>
score=0.12
I
score=0.04
You
score=0.06
hate
score=0.036
don't
score=0.020
score=0.012
score=0.003
this
score=0.002
that
score=0.0024
like
score=0.0008
care

Beam Search

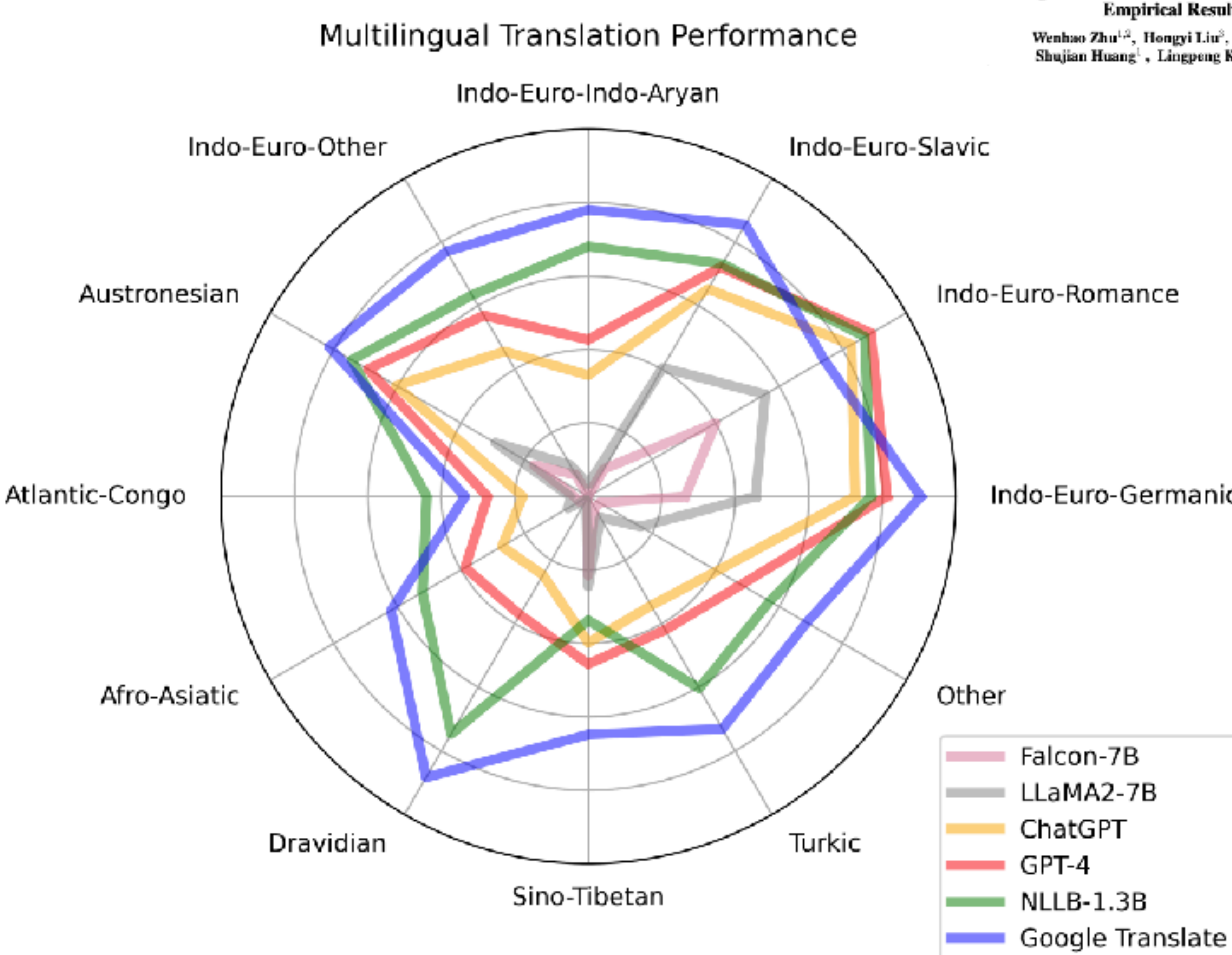

k=2
Expand
<s>
score=0.12
I
score=0.04
You
score=0.06
hate
score=0.036
don’t
score=0.020
score=0.012
score=0.003
this
score=0.002
score=0.0024
like
score=0.0008


2022
NLP
~~MT research~~
(dramatic reenactment)

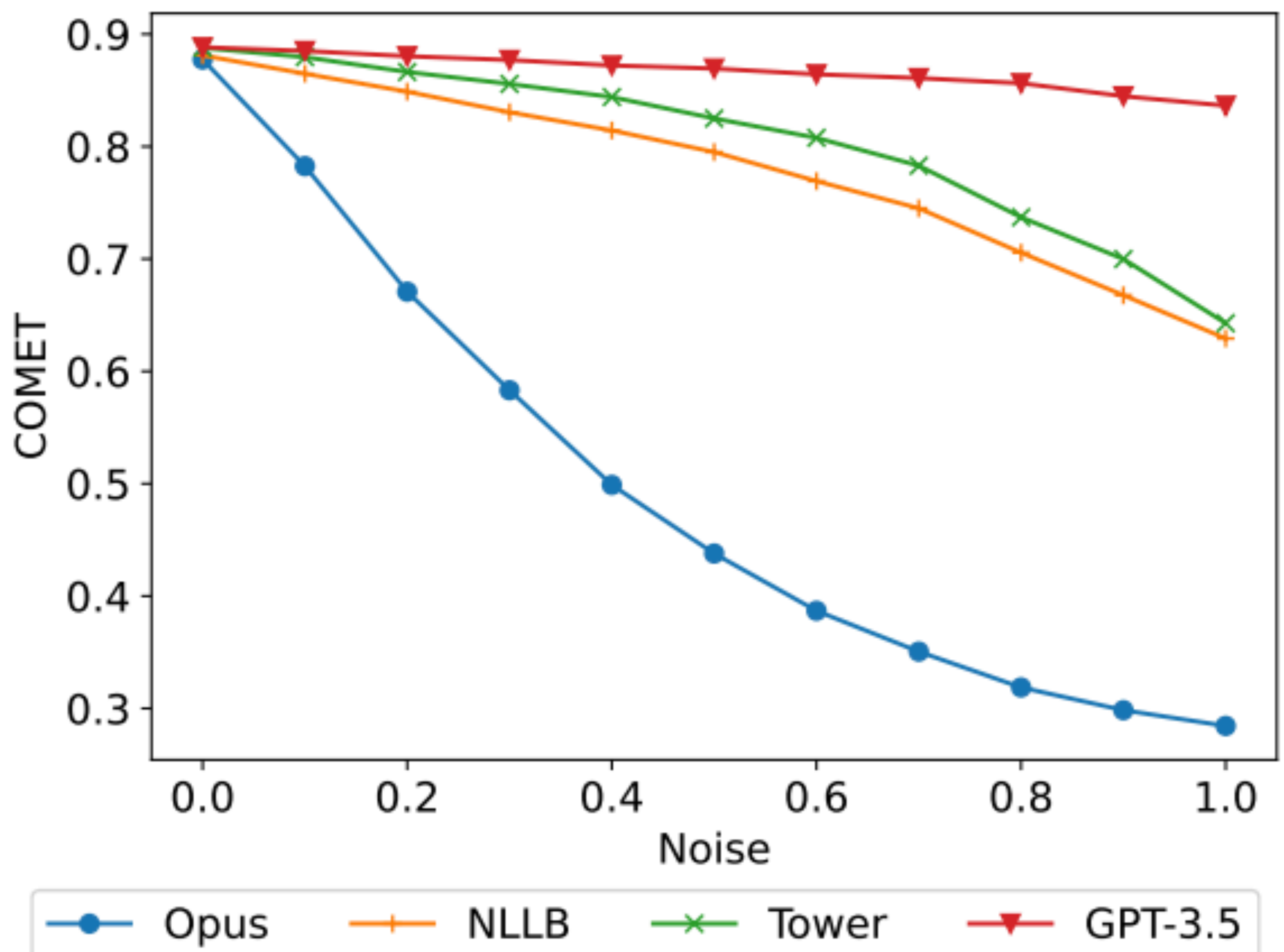
Benchmarking LLMs for Translation

Multilingual Machine Translation with Large Language Models: Empirical Results and Analysis

Wenhao Zhu[1,2], Hongyi Liu[3], Qingxiu Dong[4], Jingjing Xu[2]
Shujian Huang[1], Lingpeng Kong[5], Jiajun Chen[1], Lei Li[6]



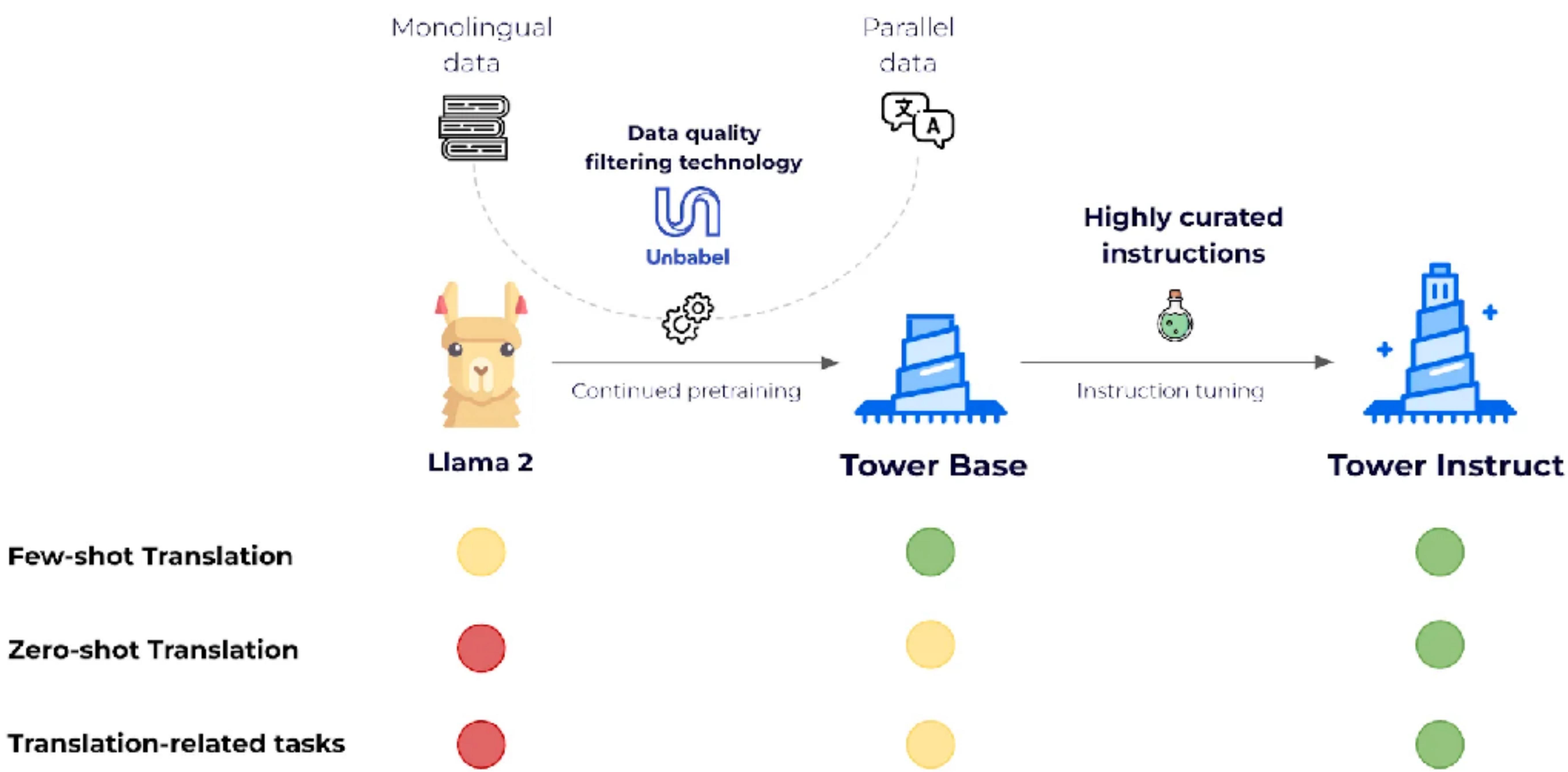
LLMs for Translation: Robustness



Did Translation Models Get More Robust Without Anyone Even Noticing?

Ben Peters* and **André F. T. Martins*†‡◇**

The Tower models



LLM-powered translation

Simulate a translation pipeline with noisy inputs.

User prompt

Fix *all errors in the following sentence.*
If the sentence does not contain any errors, just copy it.
Do not change the meaning of the sentence or insert stylistic changes.
Sentence: Ontme, a meu amiga foi ao supermrcado mas estava fechado. Queria comprsr legumes e fruta.
Corrected sentence:

TowerInstruct

`Ontem, a minha amiga foi ao supermercado mas estava fechado. Queria comprar legumes e fruta.`

User prompt

Now, ***translate*** *the corrected sentence from Portuguese into English.*
English:

TowerInstruct

`Yesterday, my friend went to the supermarket but it was closed. She wanted to buy vegetables and fruit.`

correct pronoun resolution

LLM-powered translation

Introduce terminology/glossary terms on the prompt for terminology-aware translation

Without terminology terms:

User prompt

Translate the source text from Portuguese to English.
Portuguese: Eu vou de férias este verão para Itália.
English:

TowerInstruct

`I'm going on holiday this summer to Italy.`

With terminology terms:

User prompt

Consider the following translation rule:
férias -> vacation
Translate the source text from Portuguese to English following the provided translation rules.
Portuguese: Eu vou de férias este verão para Itália.
English:

TowerInstruct

`I'm going on vacation this summer to Italy.`

LLM-powered translation

Translate into multiple languages on the same conversation

User prompt	*Translate the following text from Portuguese into English.* *Portuguese: Ontem, a minha amiga foi ao supermercado mas estava fechado. Queria comprar legumes e fruta.* *English:*
TowerInstruct	`Yesterday, my friend went to the supermarket but it was closed. She wanted to buy vegetables and fruit.`
User prompt	*Can you now translate into Spanish?*
TowerInstruct	`Ayer mi amiga fue al supermercado, pero estaba cerrado. Quería comprar verduras y fruta.`
User prompt	*Now translate to Italian.*
TowerInstruct	`Ieri la mia amica è andata al supermercato, ma era chiuso. Voleva comprare verdure e frutta.`

Evaluation

Machine Translation (reference based)

Mi piacerebbe un cappuccino freddo.

MT Model

I like one cold cappuccino.

Machine Translation (reference based)

Mi piacerebbe un cappuccino freddo.

MT Model

I like one cold cappuccino.

reference: *I would like a cold cappuccino.*

Machine Translation (reference based)

Mi piacerebbe un cappuccino freddo.

MT Model

I like one cold cappuccino.

reference: *I would like a cold cappuccino.*

Compare the output with the reference!

How do we evaluate MT?

How do we evaluate MT?

BLEU (Bilingual Evaluation Understudy)

How do we evaluate MT?

BLEU (Bilingual Evaluation Understudy)

- BLEU compares the machine-written translation to one or several human-written translation(s), and computes a similarity score based on:

How do we evaluate MT?

BLEU (Bilingual Evaluation Understudy)

- BLEU compares the machine-written translation to one or several human-written translation(s), and computes a similarity score based on:
  - n-gram precision (usually up to 3 or 4-grams)

How do we evaluate MT?

BLEU (Bilingual Evaluation Understudy)

- BLEU compares the machine-written translation to one or several human-written translation(s), and computes a similarity score based on:
  - n-gram precision (usually up to 3 or 4-grams)
  - Penalty for too-short system translations

How do we evaluate MT?

BLEU (Bilingual Evaluation Understudy)

- BLEU compares the machine-written translation to one or several human-written translation(s), and computes a similarity score based on:
  - n-gram precision (usually up to 3 or 4-grams)
  - Penalty for too-short system translations
- BLEU is useful but imperfect

How do we evaluate MT?

BLEU (Bilingual Evaluation Understudy)

- BLEU compares the machine-written translation to one or several human-written translation(s), and computes a similarity score based on:
  - n-gram precision (usually up to 3 or 4-grams)
  - Penalty for too-short system translations
- BLEU is useful but imperfect
  - There are many valid ways to translate a sentence

How do we evaluate MT?

BLEU (Bilingual Evaluation Understudy)

- BLEU compares the machine-written translation to one or several human-written translation(s), and computes a similarity score based on:
  - n-gram precision (usually up to 3 or 4-grams)
  - Penalty for too-short system translations
- BLEU is useful but imperfect
  - There are many valid ways to translate a sentence
  - So a good translation can get a poor BLEU score because it has low n-gram overlap with the human translation ☹️

Machine Translation: BLEU

reference: I would like a cold cappuccino

hypothesis: I like one cold cappuccino

Machine Translation: BLEU

reference: I would like a cold cappuccino

hypothesis: I like one cold cappuccino

Unigrams	4/5

Machine Translation: BLEU

reference: I would like a cold cappuccino

hypothesis: I like one cold cappuccino

Unigrams	4/5
Bigrams	1/4

Machine Translation: BLEU

reference: I would like a cold cappuccino

hypothesis: I like one cold cappuccino

Unigrams	4/5
Bigrams	1/4
3-grams	0/3

Machine Translation: BLEU

reference: I would like a cold cappuccino

hypothesis: I like one cold cappuccino

Unigrams	4/5
Bigrams	1/4
3-grams	0/3
4-grams	0/2

Machine Translation: BLEU

reference: I would like a cold cappuccino

hypothesis: I like one cold cappuccino

Unigrams	4/5
Bigrams	1/4
3-grams	0/3
4-grams	0/2

→ **average**

Machine Translation: BLEU

reference: I would like a cold cappuccino

hypothesis: I like one cold cappuccino

Unigrams	4/5
Bigrams	1/4
3-grams	0/3
4-grams	0/2

→ **average**

Can we cheat?

Machine Translation: BLEU

reference: I would like a cold cappuccino

hypothesis: I like like like like one cold cappuccino

Unigrams	7/8
Bigrams	1/7
3-grams	0/6
4-grams	0/5

Can we cheat?

Solution: Only count each word once.

Machine Translation: BLEU

reference: I would like a cold cappuccino

hypothesis: I would like

Unigrams	3/3
Bigrams	2/2
3-grams	1/1
4-grams	—

Can we cheat?

Solution: Brevity Penalty.

MT: Problems with BLEU

reference: I would like a cold cappuccino

hypothesis 1: *I would like one cold cappuccino*

These three hypotheses have the same BLEU score!

Solution: Use paraphrases, synonyms, etc (Meteor)

MT: Problems with BLEU

reference: I would like a cold cappuccino

hypothesis 1: *I would like one cold cappuccino*

hypothesis 2: *I would like a cold espresso*

These three hypotheses have the same BLEU score!

Solution: Use paraphrases, synonyms, etc (Meteor)

MT: Problems with BLEU

reference: I would like a cold cappuccino

hypothesis 1: *I would like one cold cappuccino*

hypothesis 2: *I would like a cold espresso*

hypothesis 3: *I would like a cold monk*

These three hypotheses have the same BLEU score!

Solution: Use paraphrases, synonyms, etc (Meteor)

MT: Problems with BLEU

source: ***behaving as if you are among those whom we could not civilize***

reference: *uygarlatıramadıklarımızdanmı¸ssınızcasına*

Languages with Rich Morphology: How dow we even evaluate this?

Solution: Use subwords, character-Fscore — chrF

MT: Human Evaluation

It is almost always better to ask humans!
e.g. in MT, we ask translators

Way 1:
We show system outputs to the annotators, and they provide a score (e.g. 1-5 Likert scale, or 0-100 score)

Way 2:
We show **2** system outputs to the annotators, and they annotate which one of the two they think is better.

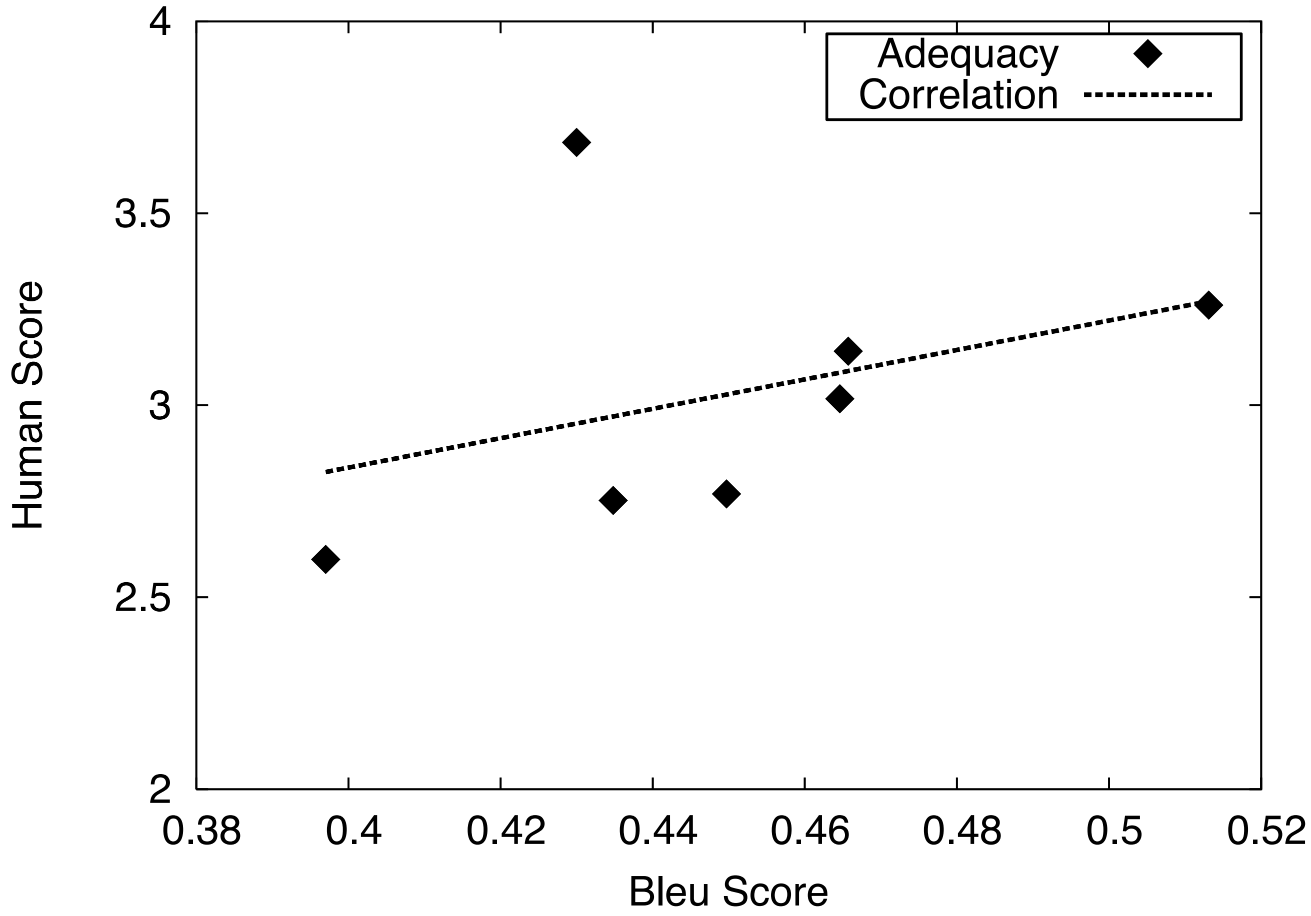
Evaluation of Evaluation Metrics

- Automatic metrics are low cost, tunable, consistent
- But are they correct?

→ Yes, if they correlate with human judgement

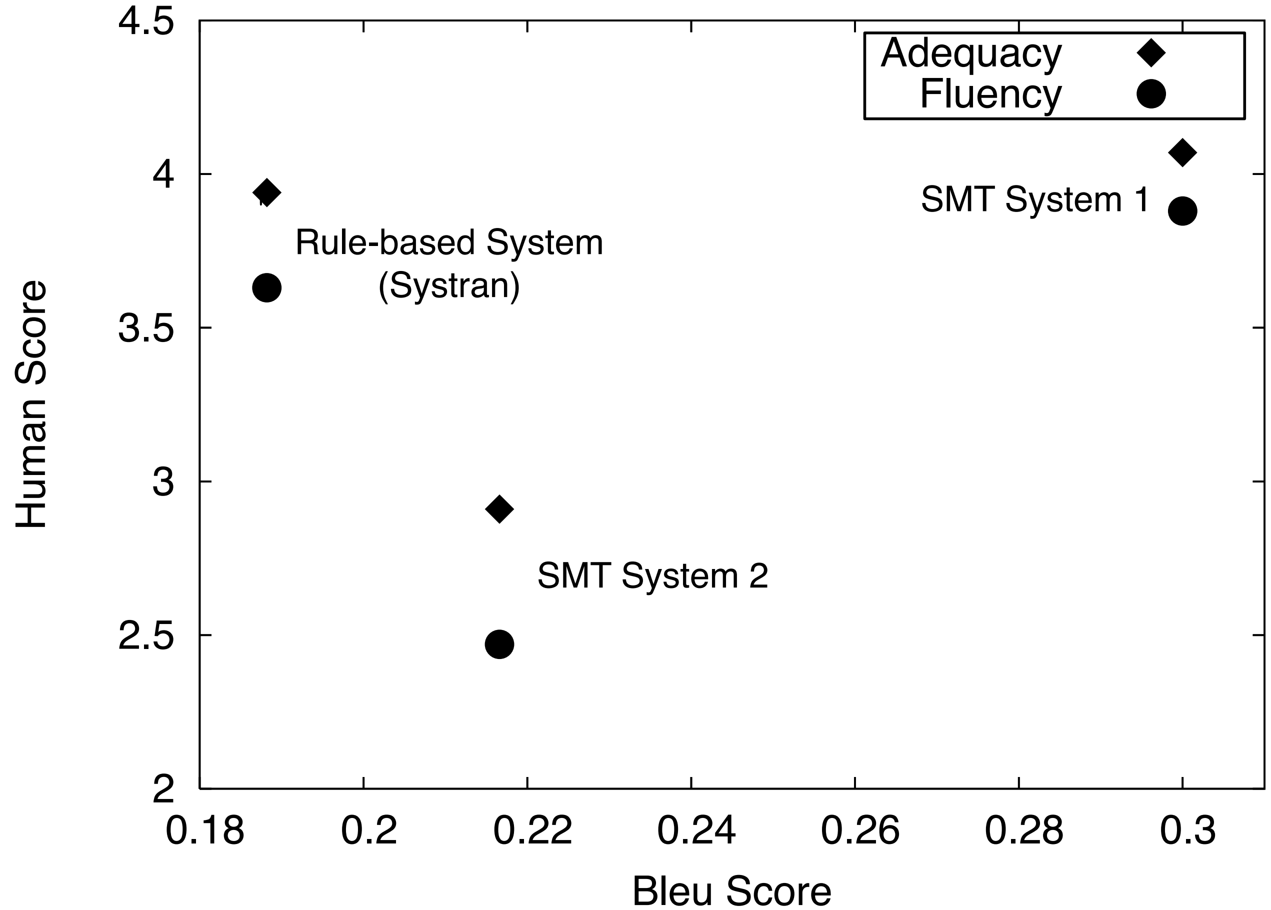
Evidence of Shortcomings of Automatic Metrics

Post-edited output vs. statistical systems (NIST 2005)



Evidence of Shortcomings of Automatic Metrics

Rule-based vs. statistical systems



WMT Metrics Shared Task

- Annual event to evaluate metrics
- Piggy-backs on the WMT General Translation Task
  - new test set every year
  - research systems and commercial systems
  - lately also large language models
  - human evaluation of automatic evaluations
- New metrics proposed
- Evaluation by correlation with human judgments

Metric		avg corr
XCOMET-Ensemble	**1**	**0.825**
XCOMET-QE-Ensemble*	2	0.808
MetricX-23	2	0.808
GEMBA-MQM*	2	0.802
MetricX-23-QE*	2	0.800
mbr-metricx-qe*	3	0.788
MaTESe	3	0.782
CometKiwi*	3	0.782
COMET	3	0.779
BLEURT-20	3	0.776
KG-BERTScore*	3	0.774
sescoreX	3	0.772
cometoid22-wmt22*	4	0.772
docWMT22CometDA	4	0.768
docWMT22CometKiwiDA*	4	0.767
Calibri-COMET22	4	0.767
Calibri-COMET22-QE*	4	0.755
YiSi-1	4	0.754
MS-COMET-QE-22*	5	0.744
prismRef	5	0.744
mre-score-labse-regular	5	0.743
BERTscore	5	0.742
XLsim	6	0.719
f200spBLEU	7	0.704
MEE4	7	0.704
tokengram_F	7	0.703
embed_llama	7	0.701
BLEU	7	0.696
chrF	7	0.694
eBLEU	7	0.692
Random-sysname*	8	0.529
prismSrc*	9	0.455

(WMT 2023)

Trained Metrics: COMET

- Two decades of evaluation campaigns for machine translation metrics
  $\rightarrow$ a lot of human judgment data

- Goal: automatic metric that correlates with human judgment

- Make it a machine learning problem
  - input: machine translation, reference translation
  - output: human annotation score

- COMET: Trained neural model for evaluation

Reference-Free Evaluation

- We have data in the form

  input, translation, human reference → human judgment

- We can also train a model on

  input, translation → human judgment

- CometKiwi: trained evaluation model without references
- Also called **quality estimation** or **confidence estimation**

Semisupervised and Unsupervised Methods

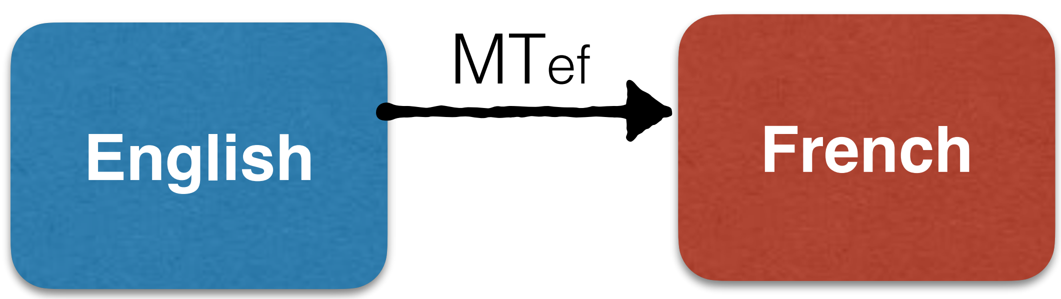
On Using Monolingual Corpora in Neural Machine Translation (Gulcehre et al. 2015)

Parallel English French

Monolingual

On Using Monolingual Corpora in Neural Machine Translation (Gulcehre et al. 2015)

Parallel

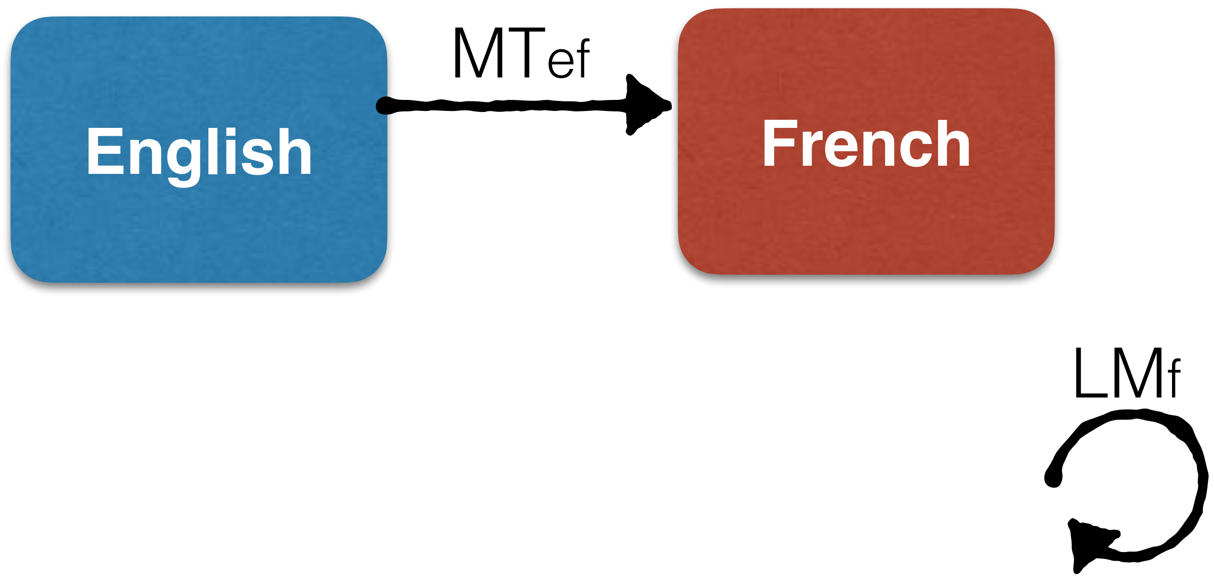


Train NMT

Monolingual

On Using Monolingual Corpora in Neural Machine Translation (Gulcehre et al. 2015)

Parallel



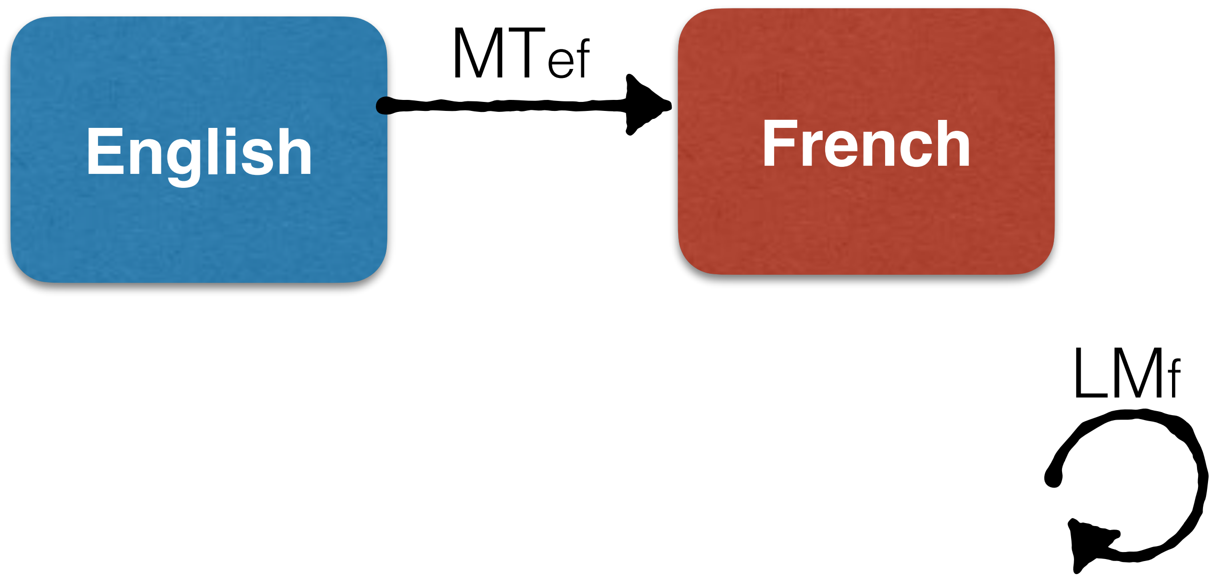
Train NMT

Monolingual

Train LM

On Using Monolingual Corpora in Neural Machine Translation (Gulcehre et al. 2015)

Parallel



Train NMT

Monolingual

Train LM

Combine the two!

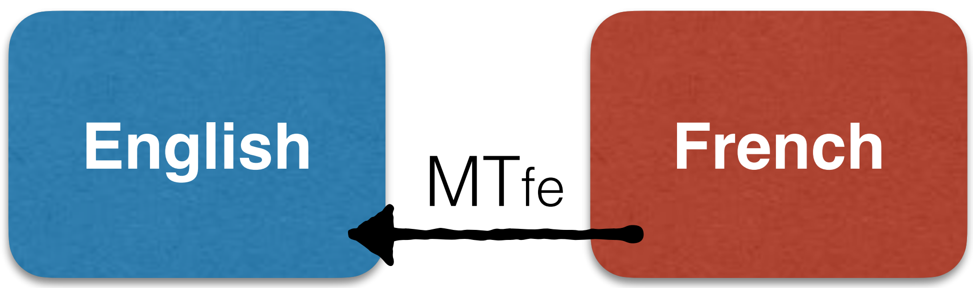
Back-translation (Sennrich et al. 2016)

Parallel English French

Monolingual

Back-translation (Sennrich et al. 2016)

Parallel

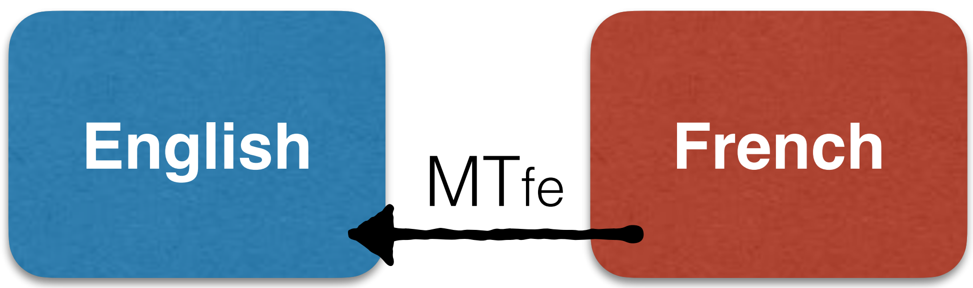


Train French->English

Monolingual

Back-translation (Sennrich et al. 2016)

Parallel



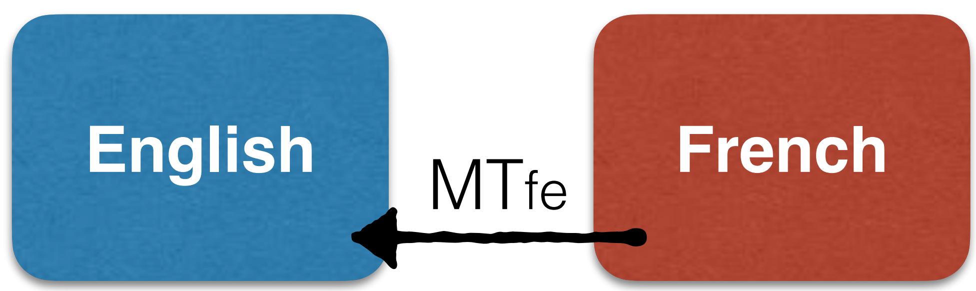
Train French->English

Back-Translate
Monolingual data

Monolingual

Back-translation (Sennrich et al. 2016)

Parallel



Train French->English

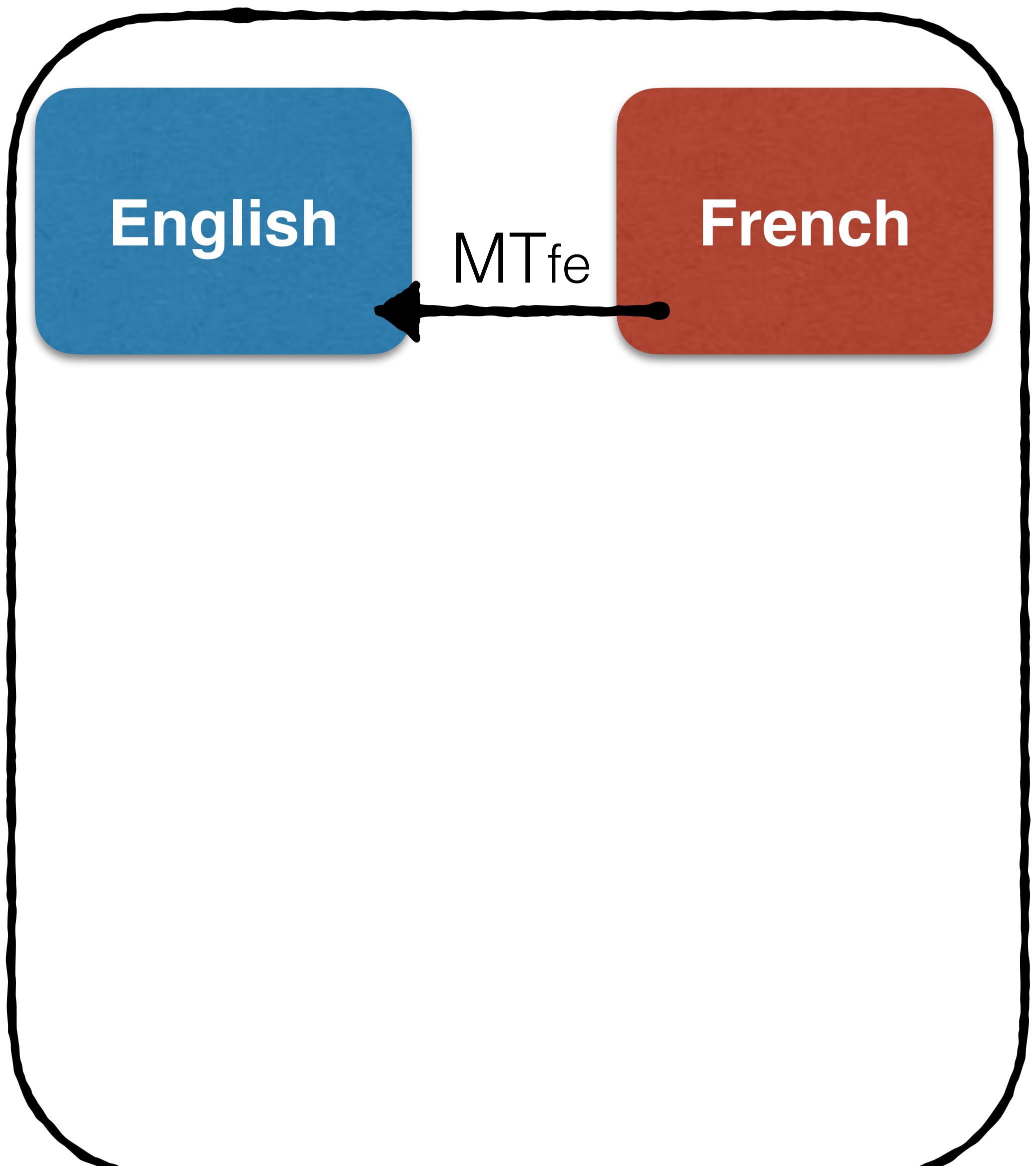
Back-Translate Monolingual data

Monolingual

Back-translation (Sennrich et al. 2016)

Parallel

Monolingual



Train French->English

Back-Translate Monolingual data

Train English->French

Dual Learning
(He et al. 2016)

Parallel English French

Monolingual

Dual Learning (He et al. 2016)

Assume MT_{ef}, MT_{fe}, LM_e, LM_f

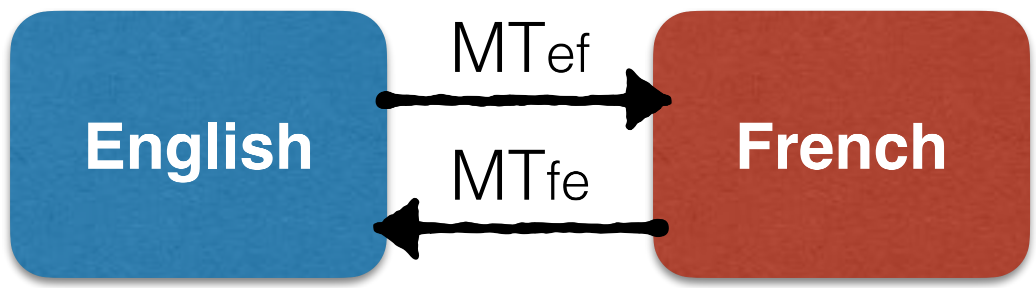
Parallel English French

Monolingual

Dual Learning
(He et al. 2016)

Assume MT_{ef}, MT_{fe}, LM_e, LM_f

Parallel



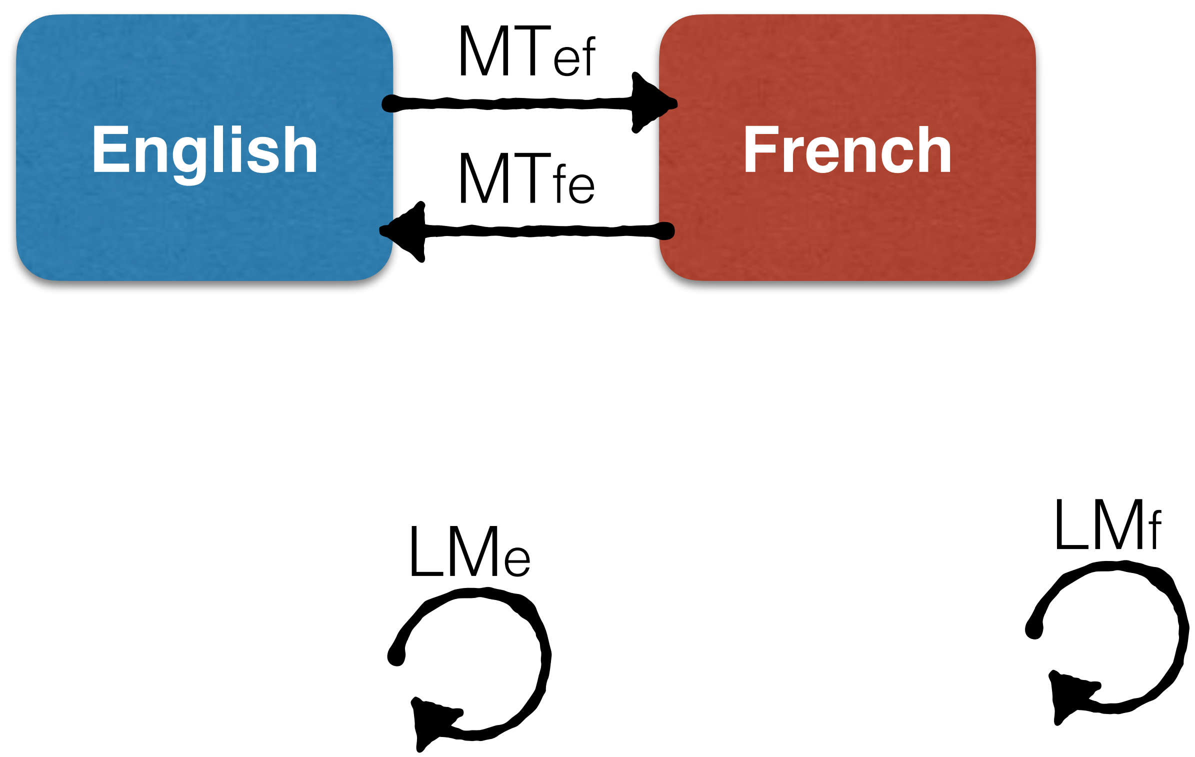
Monolingual

Dual Learning
(He et al. 2016)

Assume MT_{ef}, MT_{fe}, LM_e, LM_f

Parallel

Monolingual



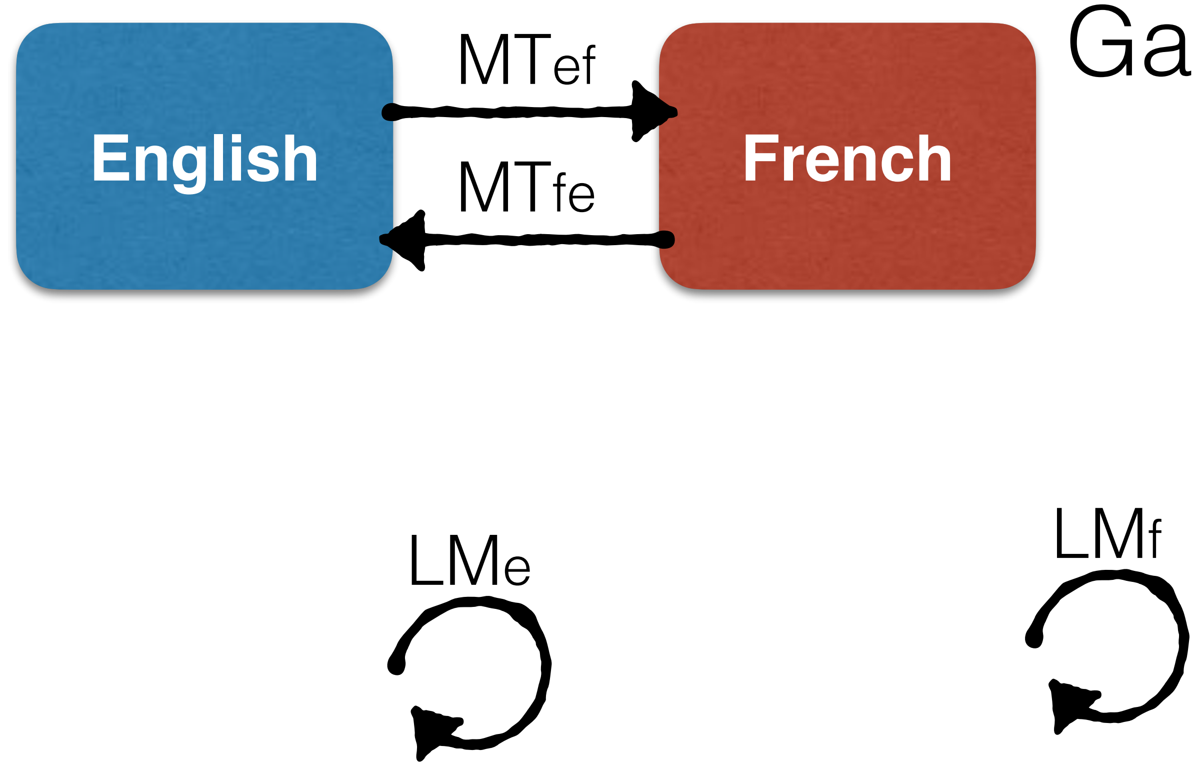
Dual Learning
(He et al. 2016)

Assume MT_{ef}, MT_{fe}, LM_e, LM_f

Game:

Parallel

Monolingual



Dual Learning (He et al. 2016)

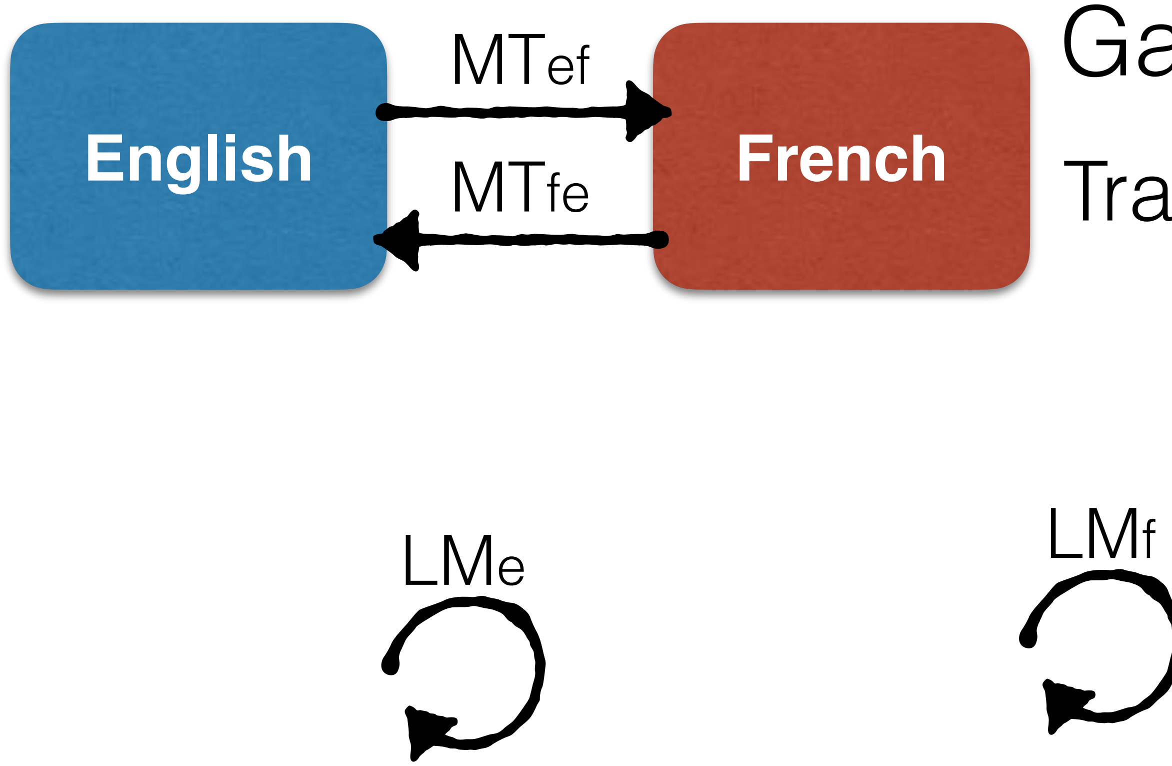
Assume MT_{ef}, MT_{fe}, LM_e, LM_f

Game:

Translate sample with MT_{ef}

Parallel

Monolingual



Dual Learning
(He et al. 2016)

Assume MT_{ef}, MT_{fe}, LM_e, LM_f

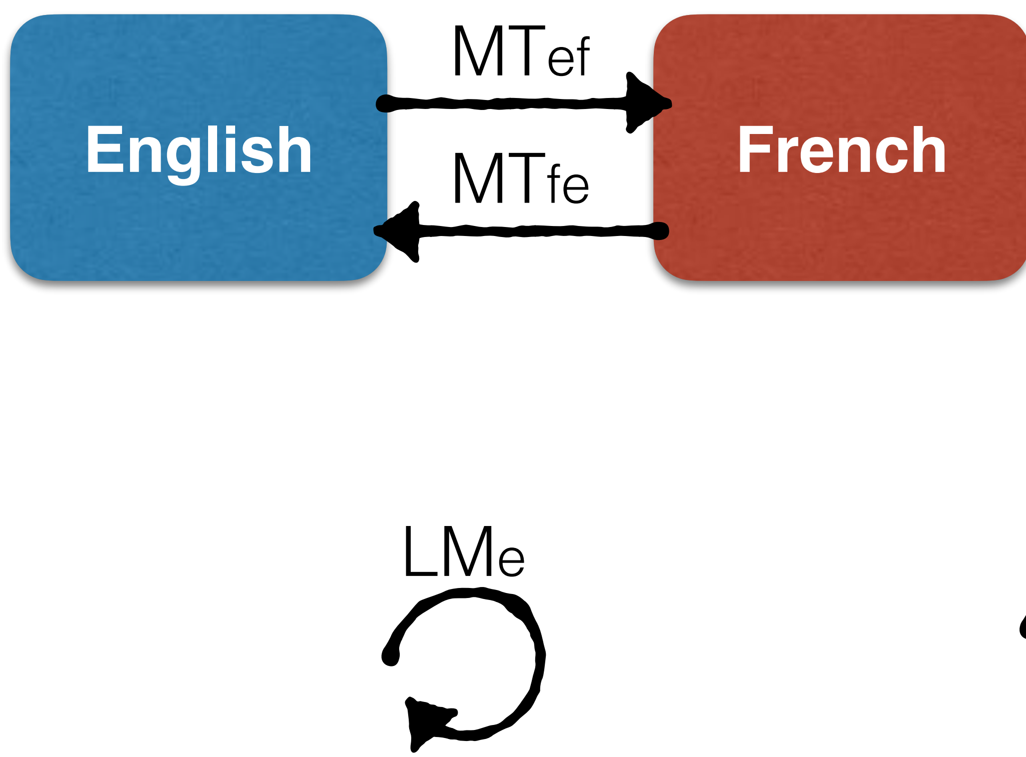
Game:

Translate sample with MT_{ef}

Get reward with LM_f

Parallel

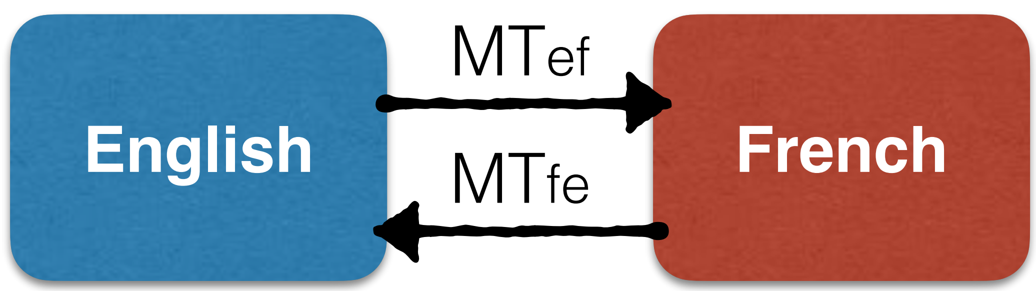
Monolingual



Dual Learning
(He et al. 2016)

Assume MT_{ef}, MT_{fe}, LM_e, LM_f

Parallel



Game:

Translate sample with MT_{ef}

Get reward with LM_f

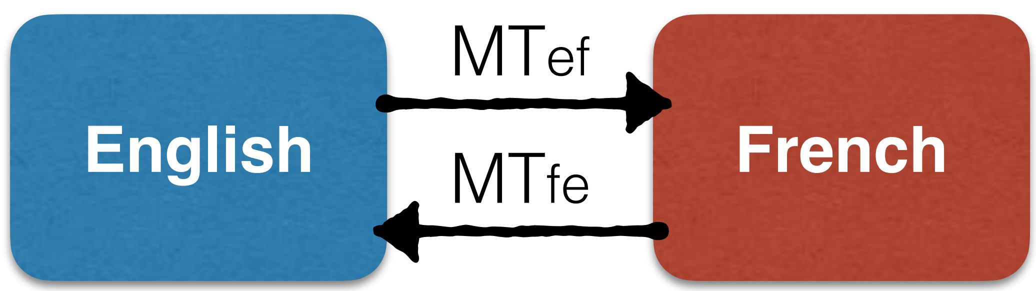
Monolingual

Translate sample with MT_{fe}

Dual Learning (He et al. 2016)

Assume MT_{ef}, MT_{fe}, LM_e, LM_f

Parallel

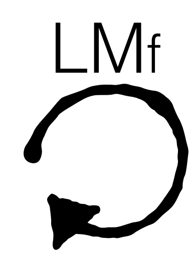


Game:

Translate sample with MT_{ef}

Get reward with LM_f

Monolingual



Translate sample with MT_{fe}

Get reward with LM_e

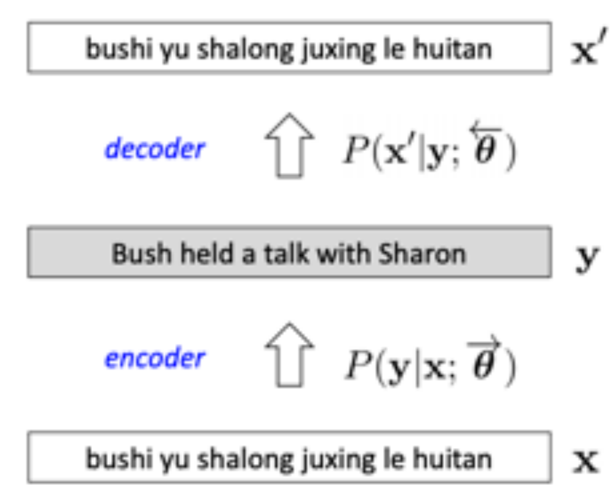
Semi-Supervised Learning for MT (Cheng et al. 2016)

Parallel

English

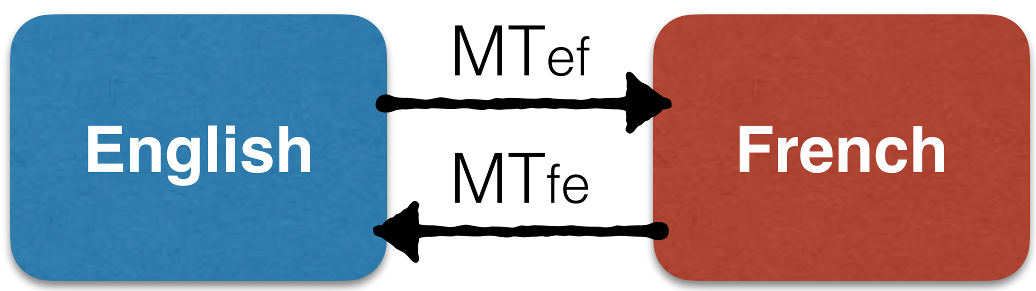
French

Monolingual

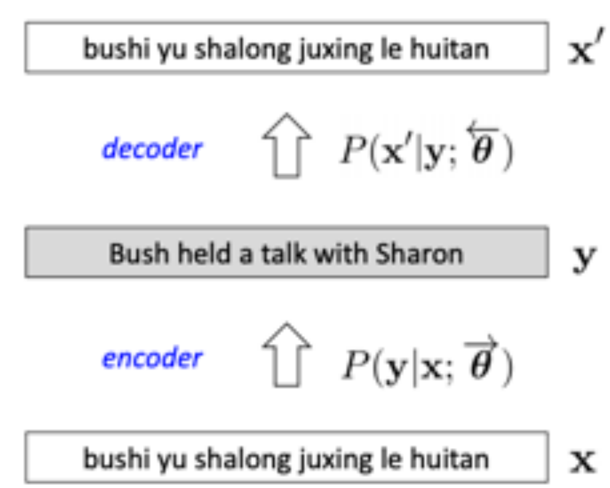


Semi-Supervised Learning for MT (Cheng et al. 2016)

Parallel



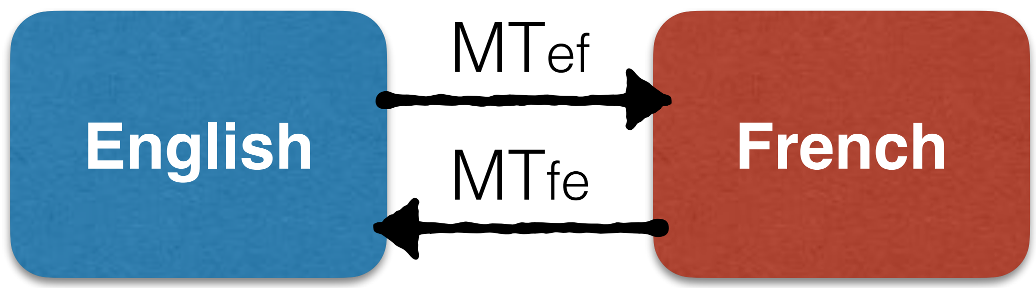
Monolingual



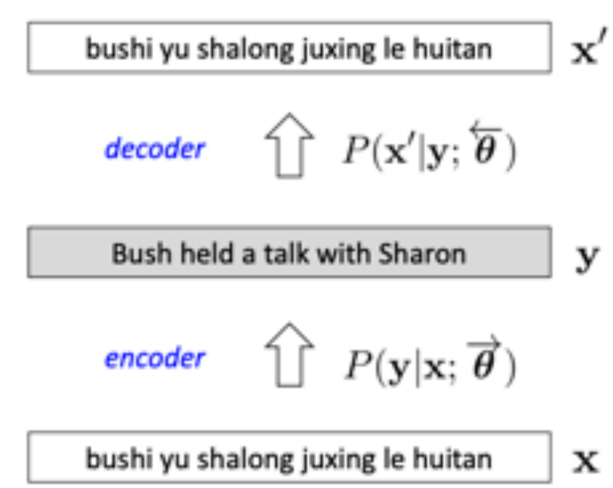
Semi-Supervised Learning for MT (Cheng et al. 2016)

Round-trip translation for supervision

Parallel



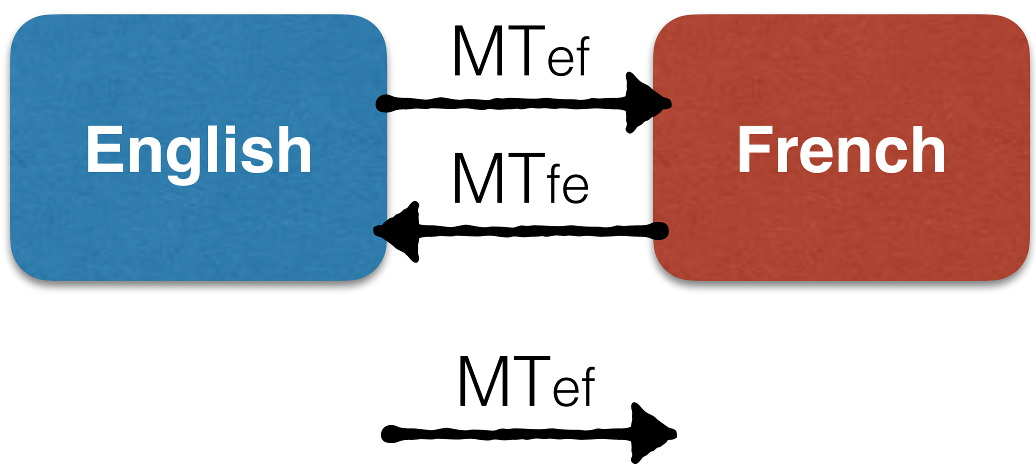
Monolingual



Semi-Supervised Learning for MT (Cheng et al. 2016)

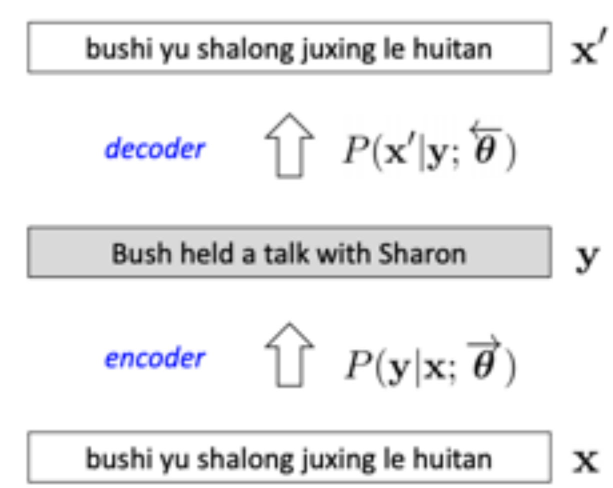
Round-trip translation for supervision

Parallel



Translate *e* to *f'* with MT_{ef}

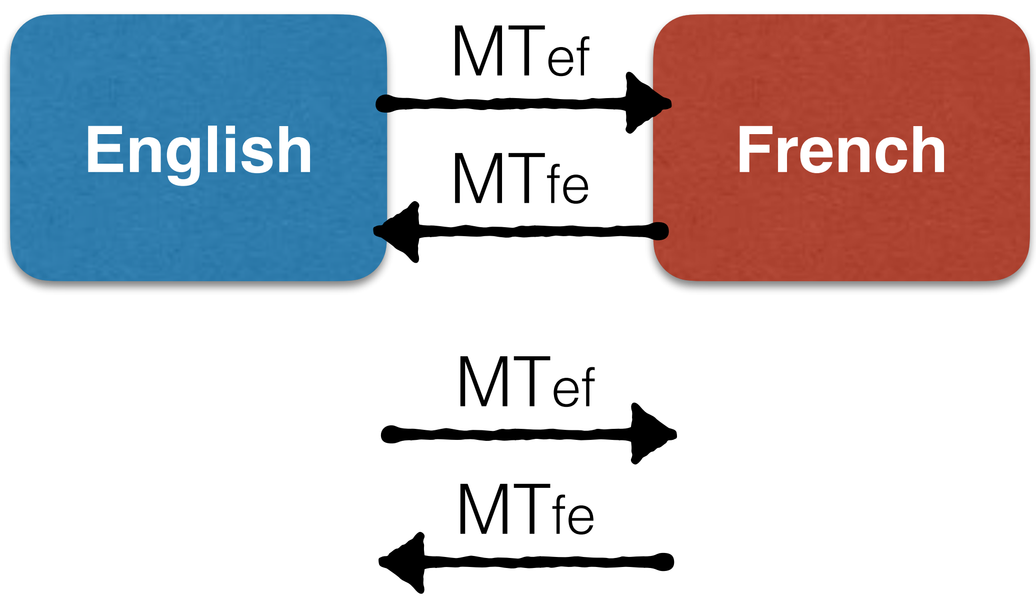
Monolingual



Semi-Supervised Learning for MT (Cheng et al. 2016)

Round-trip translation for supervision

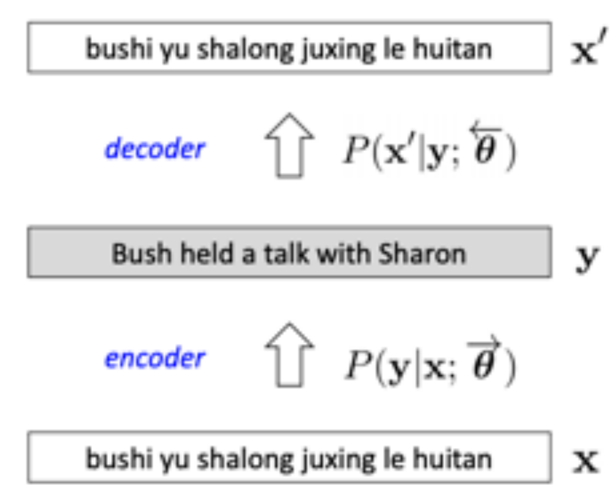
Parallel



Translate *e* to *f'* with MT_{ef}

Translate *f'* to *e'* with MT_{fe}

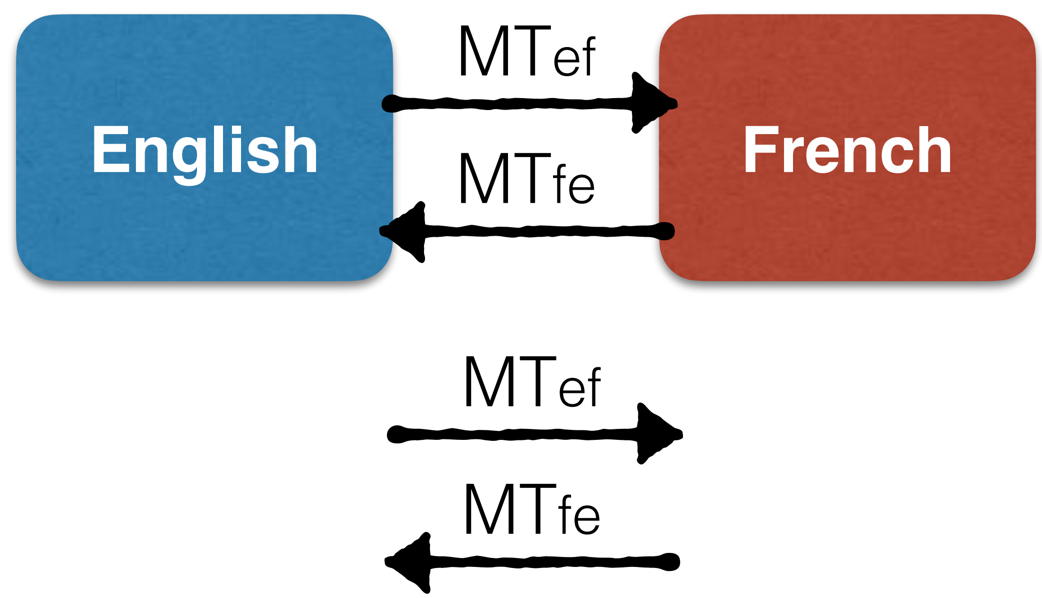
Monolingual



Semi-Supervised Learning for MT (Cheng et al. 2016)

Round-trip translation for supervision

Parallel

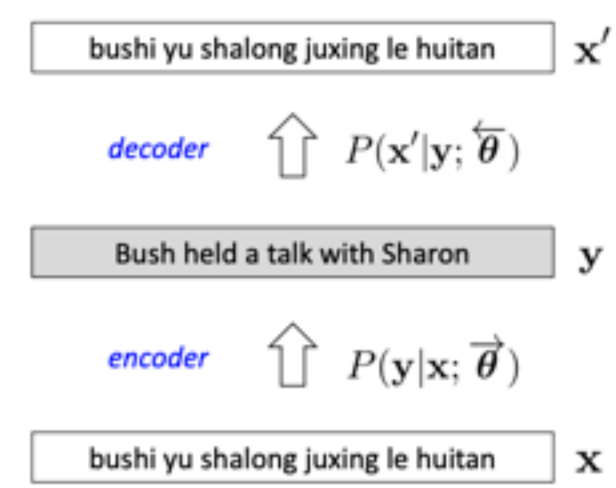


Translate *e* to *f'* with MT_{ef}

Translate *f'* to *e'* with MT_{fe}

Loss from *e* and *e'*

Monolingual



Another idea: use monolingual data to pretrain model components

Parallel

English

French

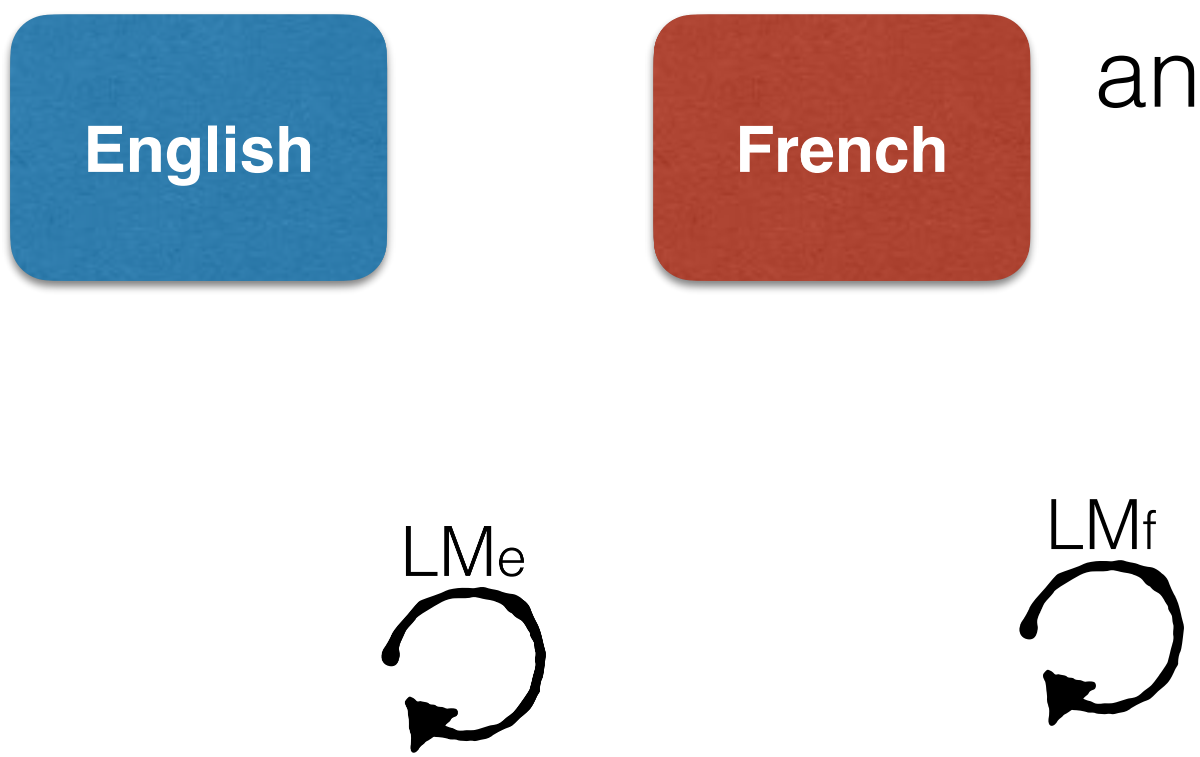
Monolingual

Another idea: use monolingual data to pretrain model components

Use the monolingual data to train the encoder and the decoder.

Parallel

Monolingual



Another idea: use monolingual data to pretrain model components

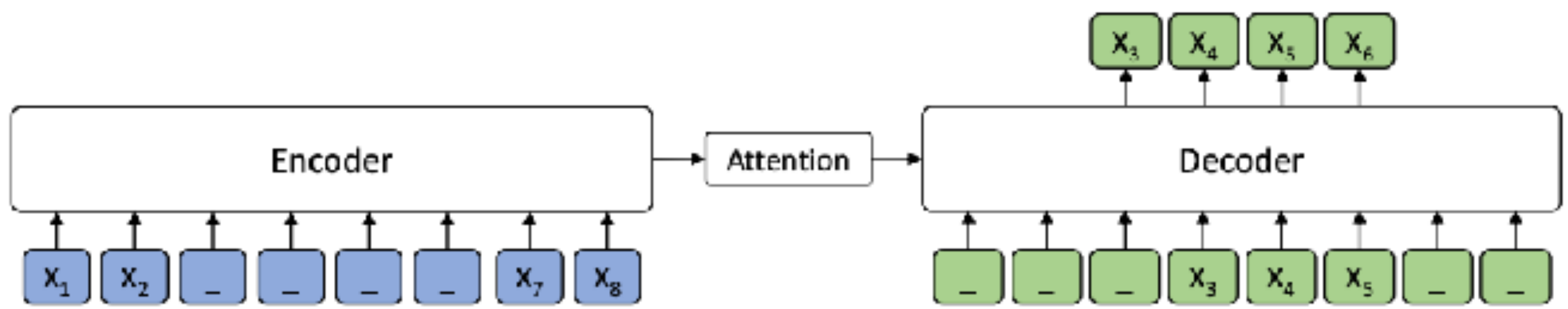
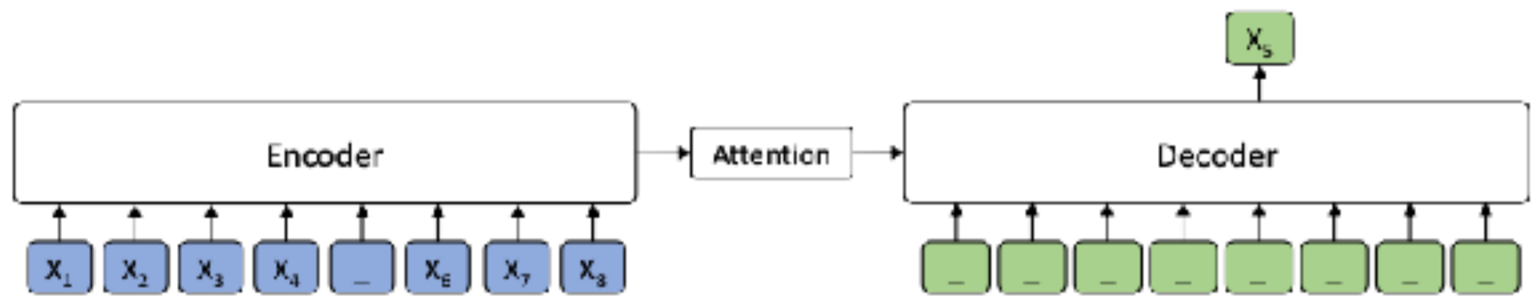


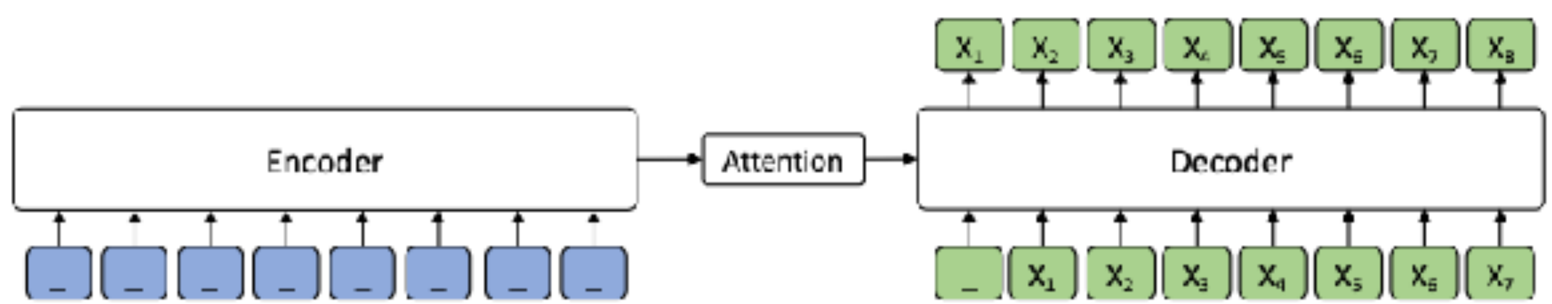
Figure 1. The encoder-decoder framework for our proposed MASS. The token "_" represents the mask symbol [M].

From "MASS: Masked Sequence to Sequence Pre-training for Language Generation", Song et al. 2019.

Another idea: use monolingual data to pretrain model components



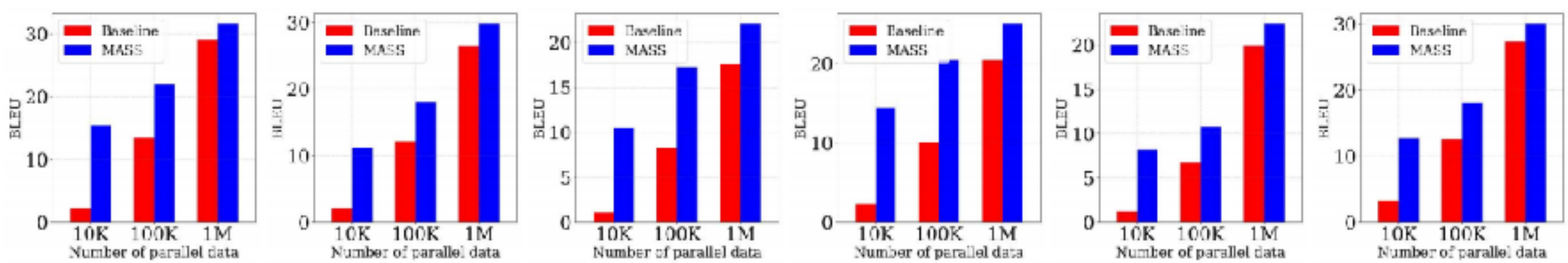
(a) Masked language modeling in BERT ($k = 1$)



(b) Standard language modeling ($k = m$)

From "MASS: Masked Sequence to Sequence Pre-training for Language Generation", Song et al. 2019.

Another idea: use monolingual data to pretrain model components



(a) en-fr (b) fr-en (c) en-de (d) de-en (e) en-ro (f) ro-en

Figure 3. The BLEU score comparisons between MASS and the baseline on low-resource NMT with different scales of paired data.

From "MASS: Masked Sequence to Sequence Pre-training for Language Generation", Song et al. 2019.

Unsupervised Translation

… at the core of it all: decipherment

French

$$\arg\max_{\theta} \prod_{f} P_{\theta}(f)$$

From "Deciphering Foreign Language", Ravi and Knight 2011.

… at the core of it all: decipherment

French

$$\arg\max_{\theta} \prod_{f} P_{\theta}(f)$$

Weaver (1955): *This is really English, encrypted in some strange symbols*

From "Deciphering Foreign Language", Ravi and Knight 2011.

… at the core of it all: decipherment

French

$$\arg\max_{\theta} \prod_{f} P_{\theta}(f)$$

Weaver (1955): *This is really English, encrypted in some strange symbols*

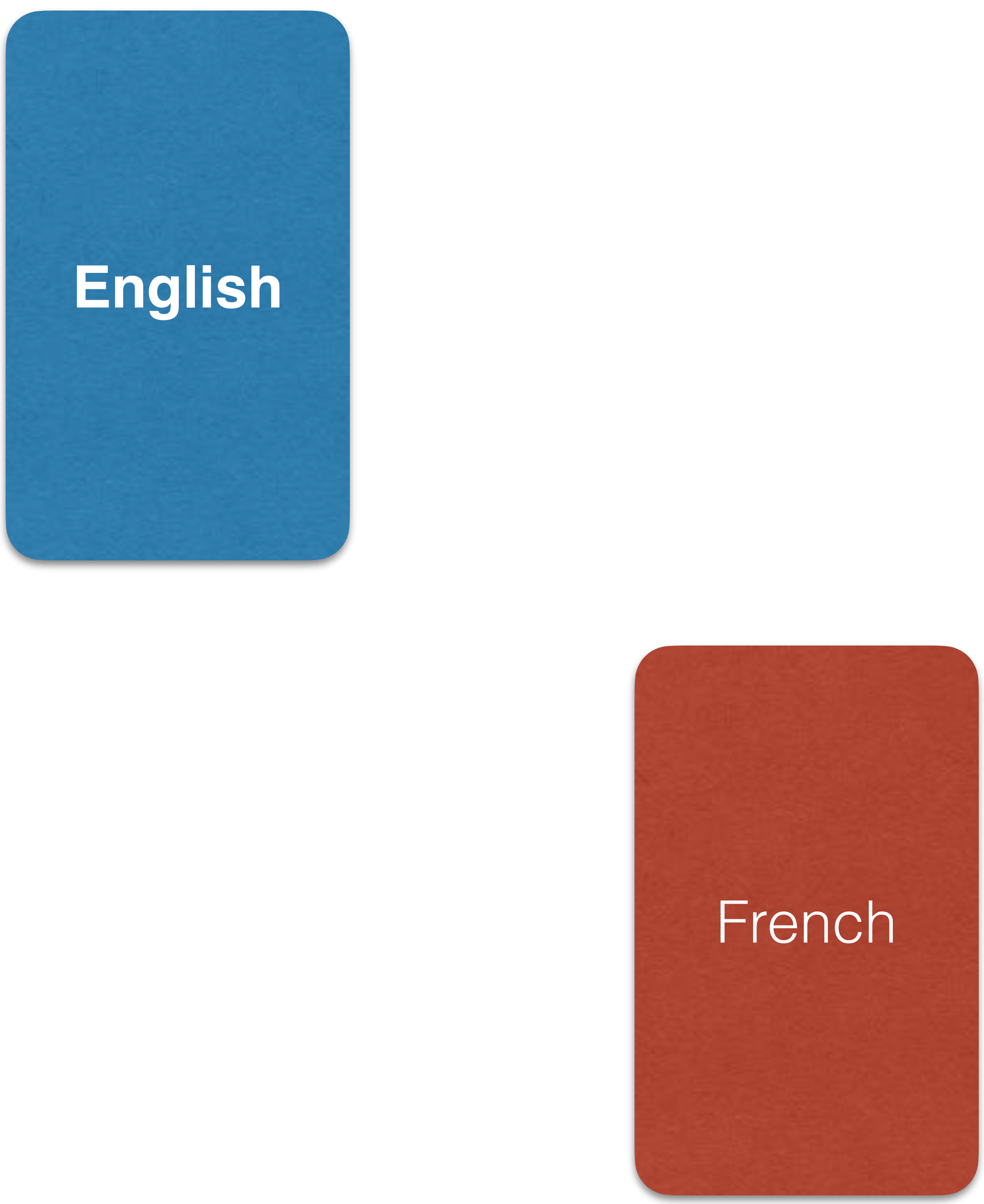
English → **French**

$$\arg\max_{\theta} \prod_{f} \sum_{e} P(e) \cdot P_{\theta}(f|e)$$

From "Deciphering Foreign Language", Ravi and Knight 2011.

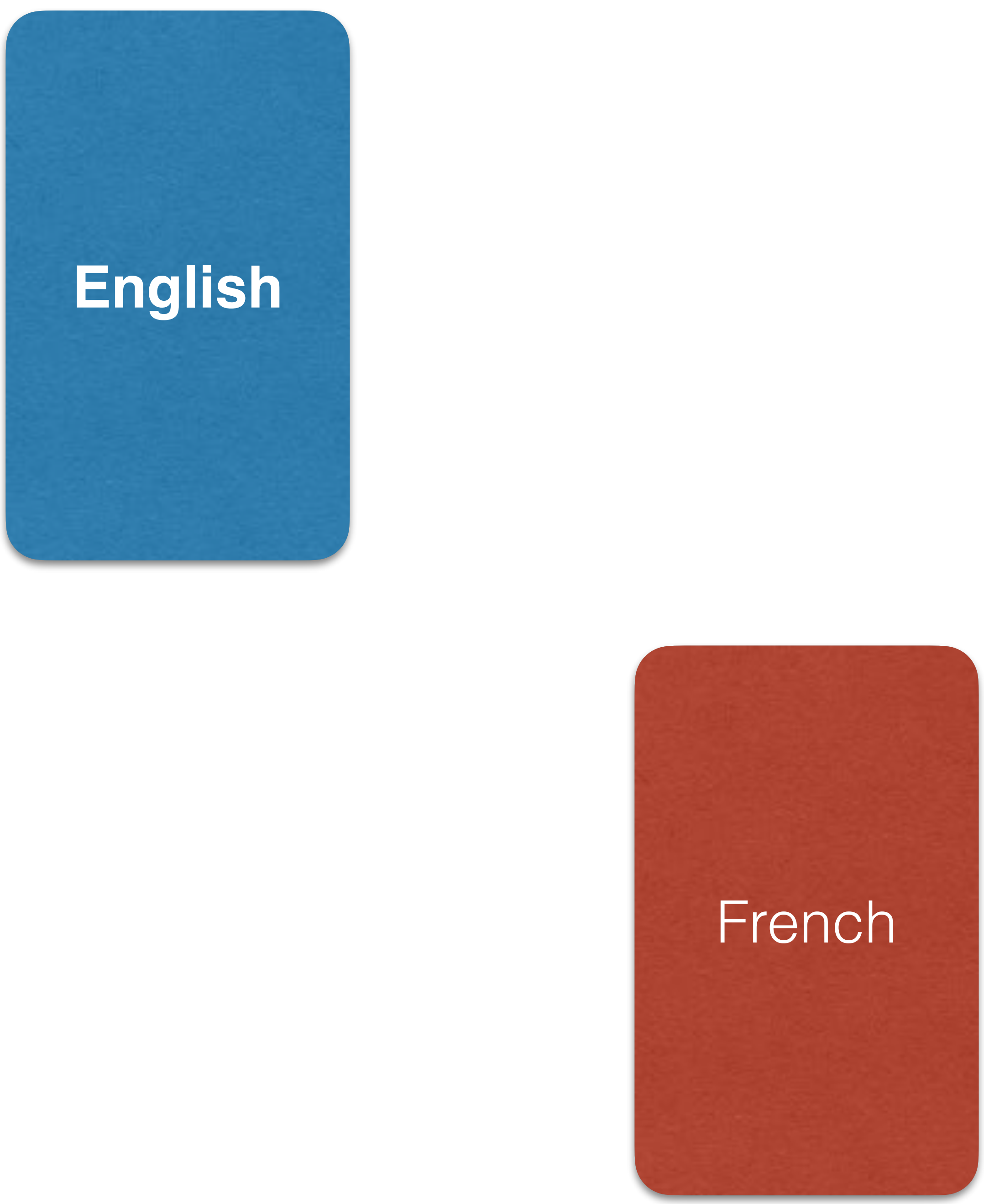
Unsupervised MT
(Lample et al. and Artetxe et al. 2018)

Unsupervised MT
(Lample et al. and Artetxe et al. 2018)

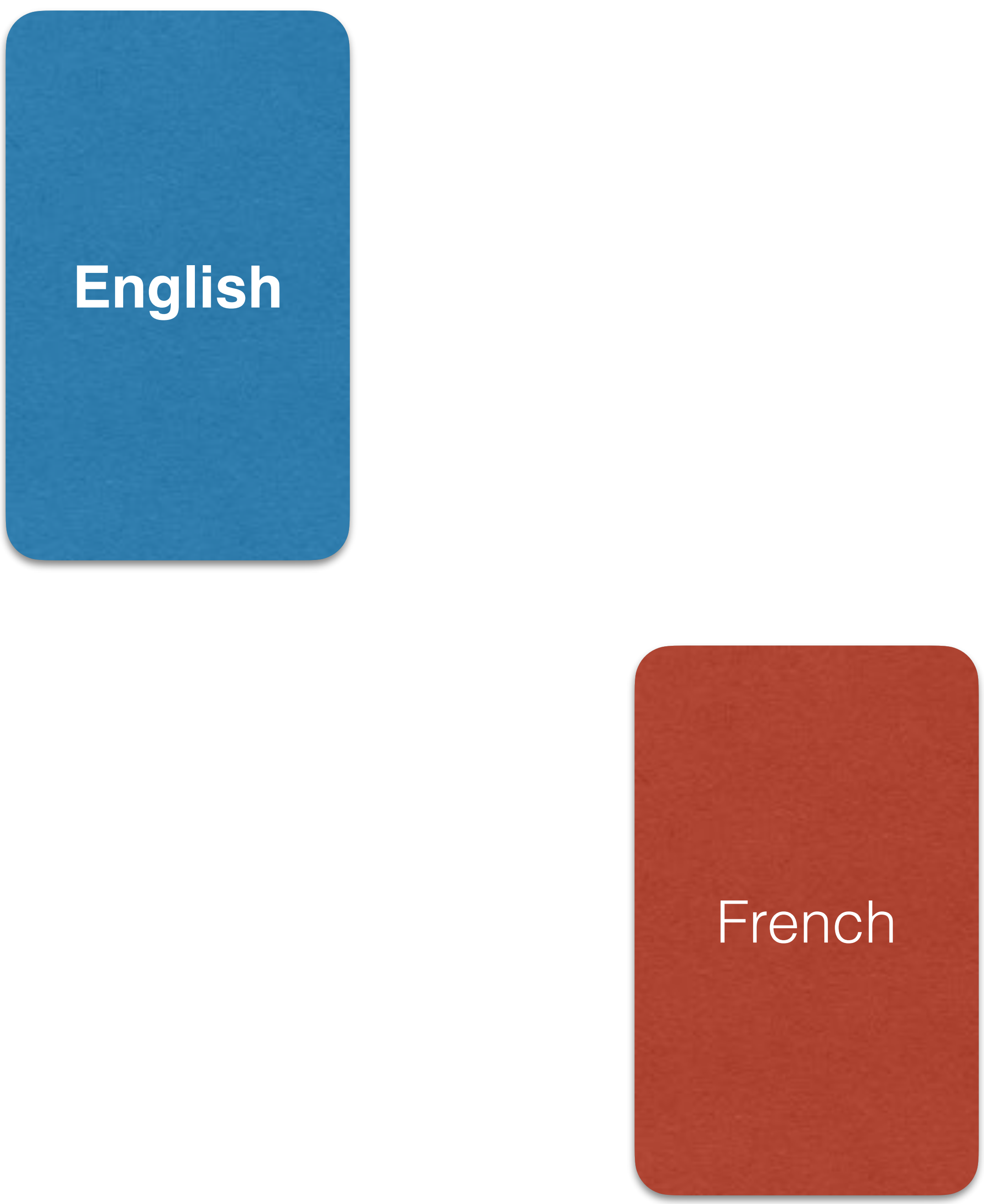


Unsupervised MT
(Lample et al. and Artetxe et al. 2018)

1. Embeddings + Unsup. BLI



Unsupervised MT
(Lample et al. and Artetxe et al. 2018)



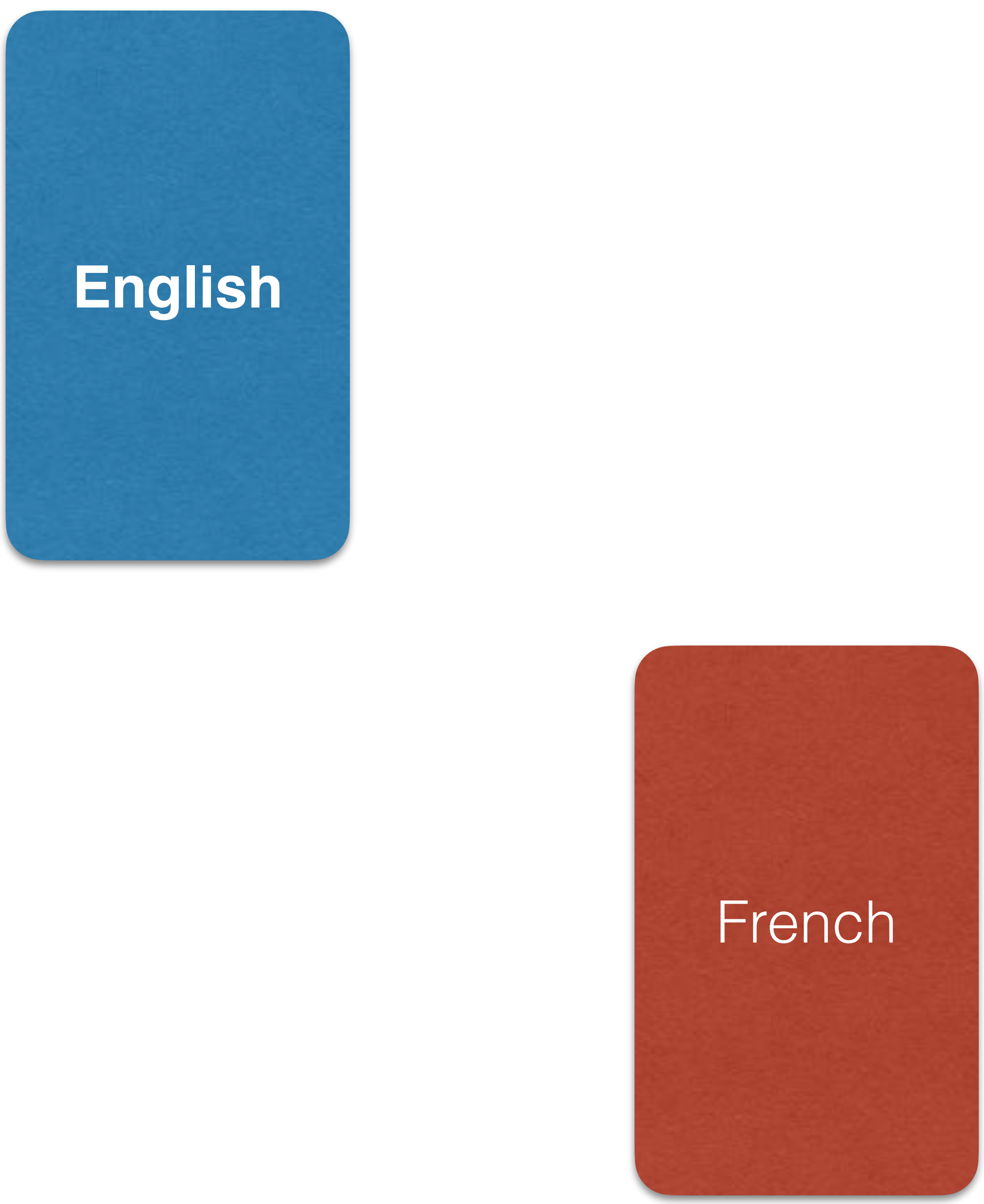
1. Embeddings + Unsup. BLI

2. BLI —> Word Translations

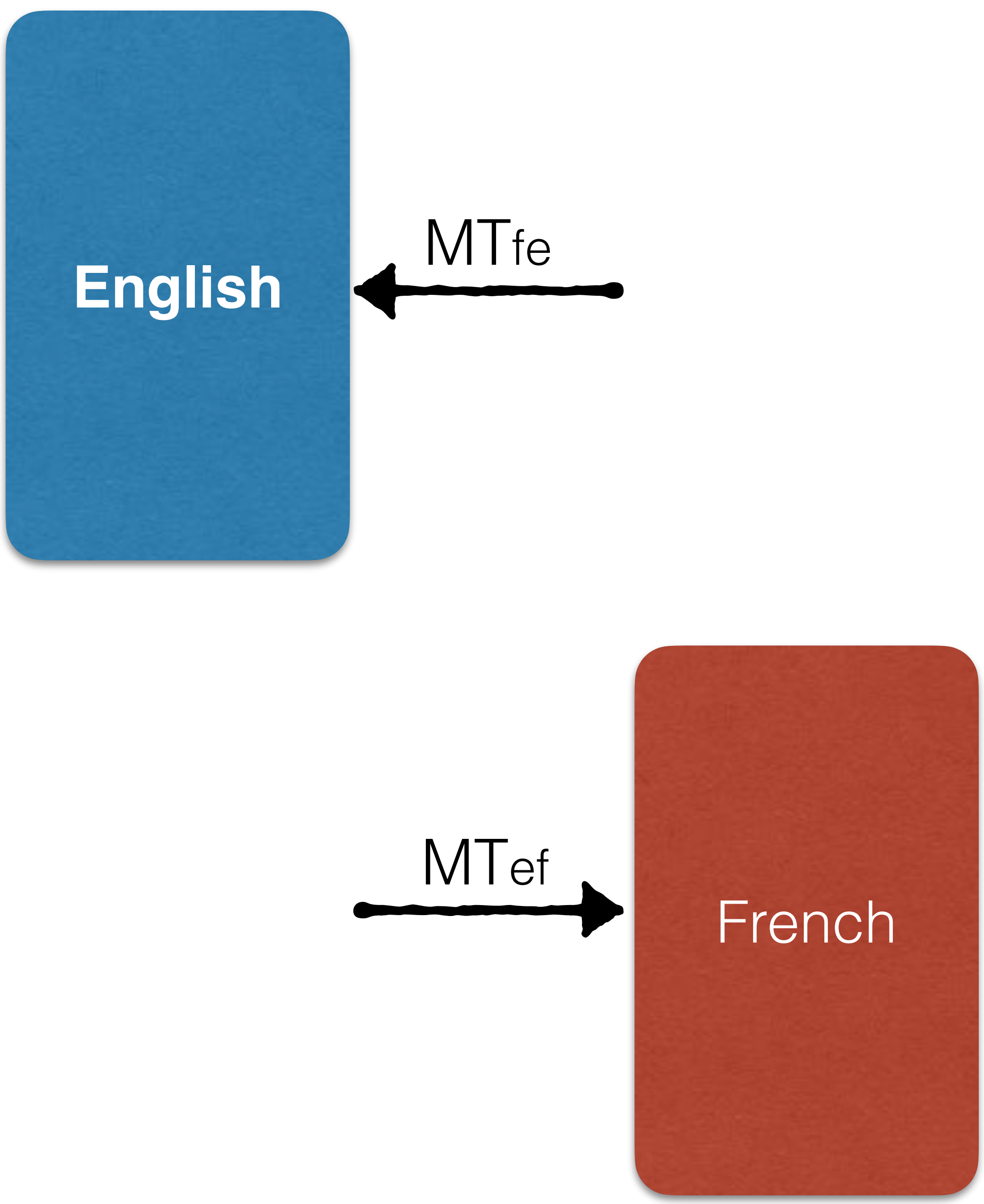
Unsupervised MT
(Lample et al. and Artetxe et al. 2018)

1. Embeddings + Unsup. BLI

2. BLI —> Word Translations

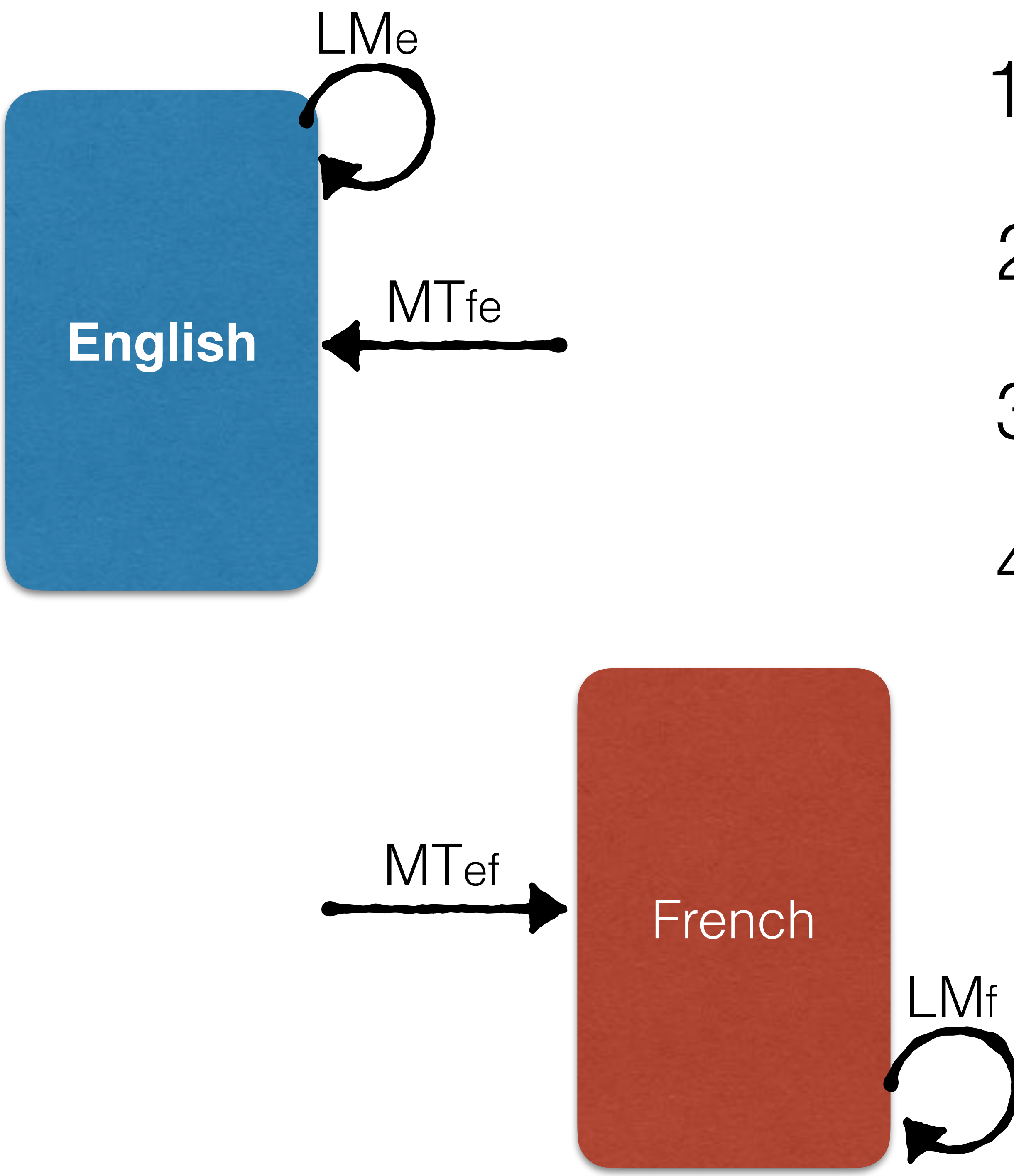


Unsupervised MT
(Lample et al. and Artetxe et al. 2018)



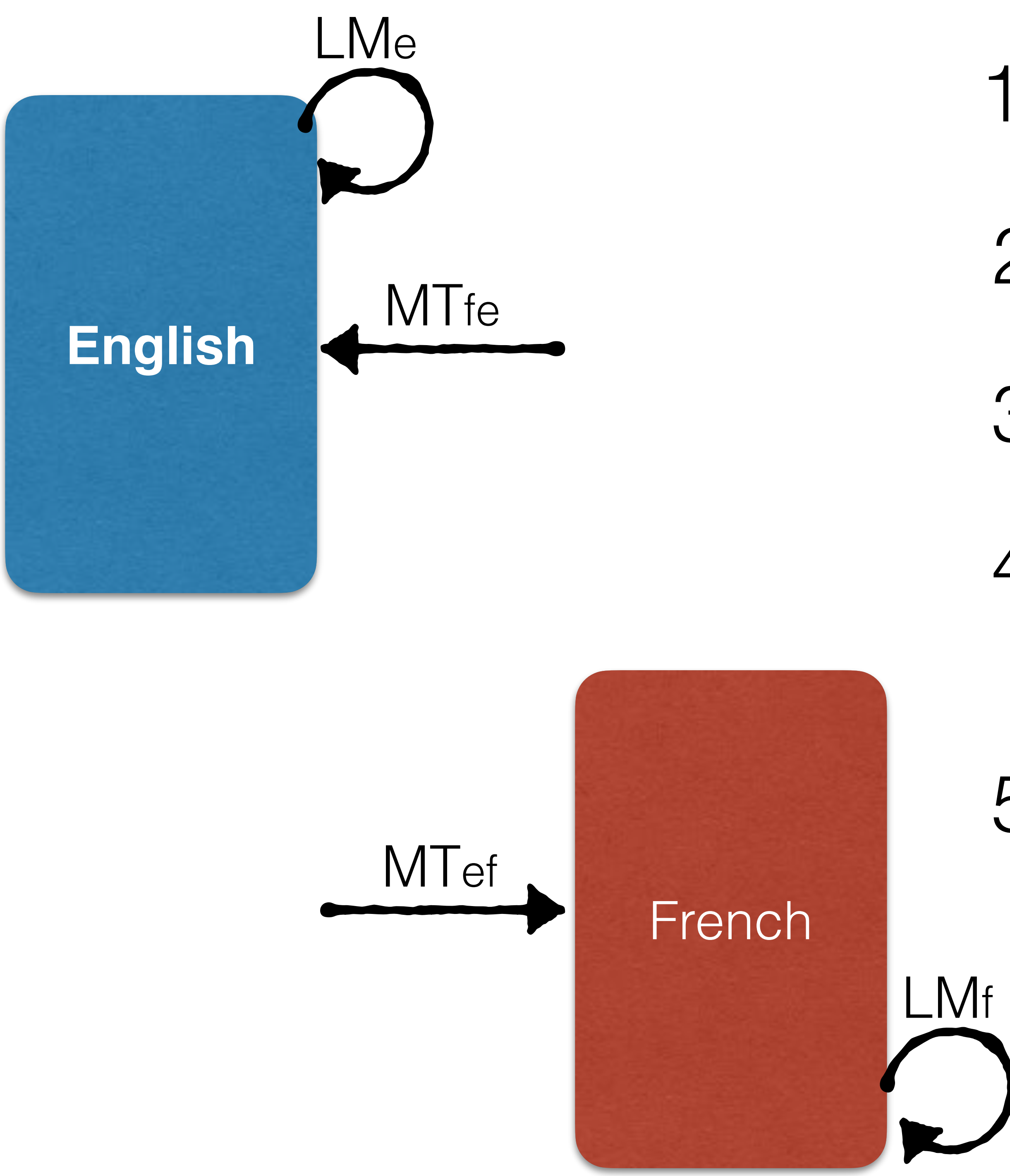
1. Embeddings + Unsup. BLI
2. BLI —> Word Translations
3. Train MT_{fe} and MT_{ef} systems

Unsupervised MT
(Lample et al. and Artetxe et al. 2018)



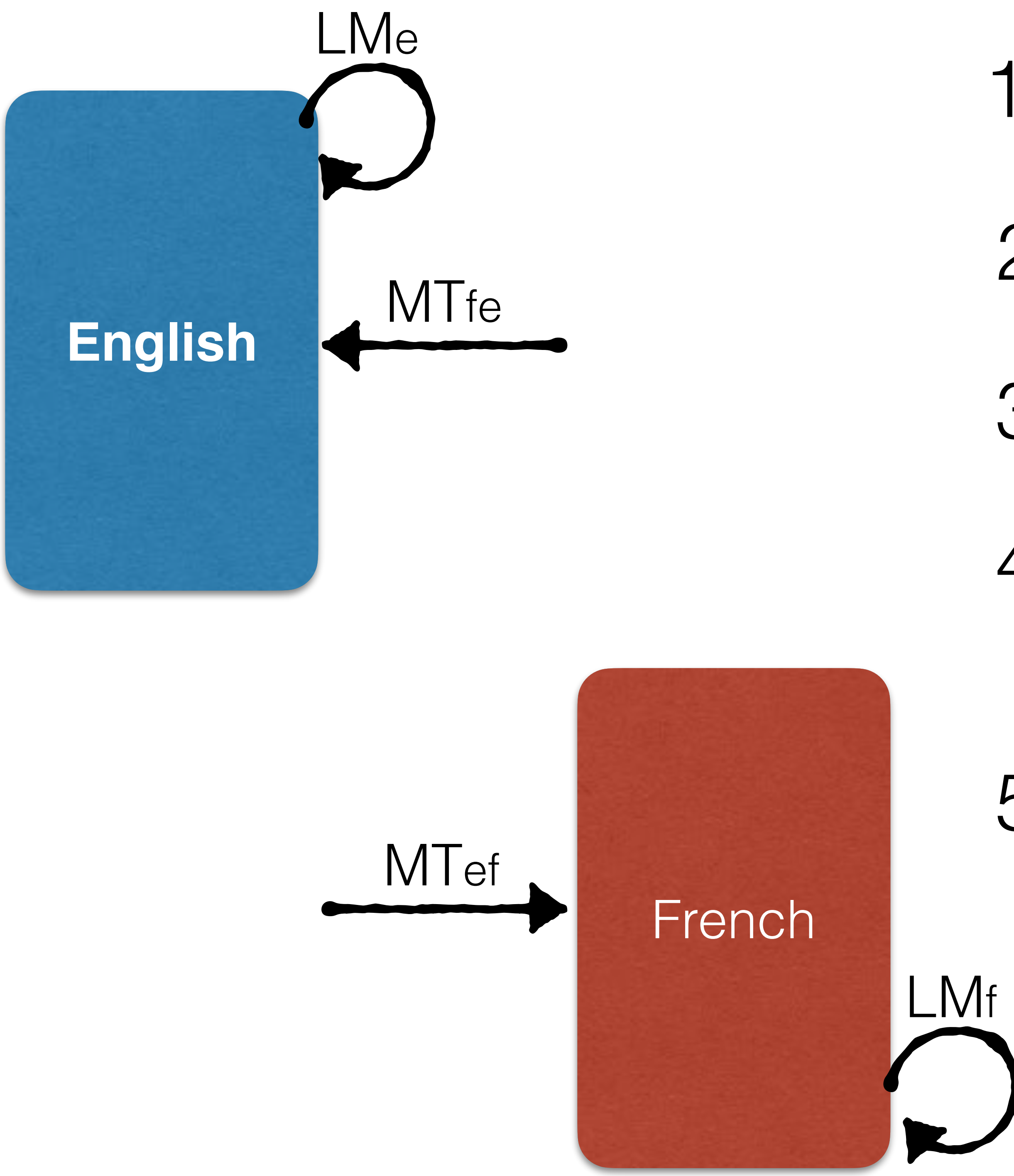
1. Embeddings + Unsup. BLI
2. BLI —> Word Translations
3. Train MT_{fe} and MT_{ef} systems
4. Meanwhile, use unsupervised objectives (denoising LM)

Unsupervised MT
(Lample et al. and Artetxe et al. 2018)



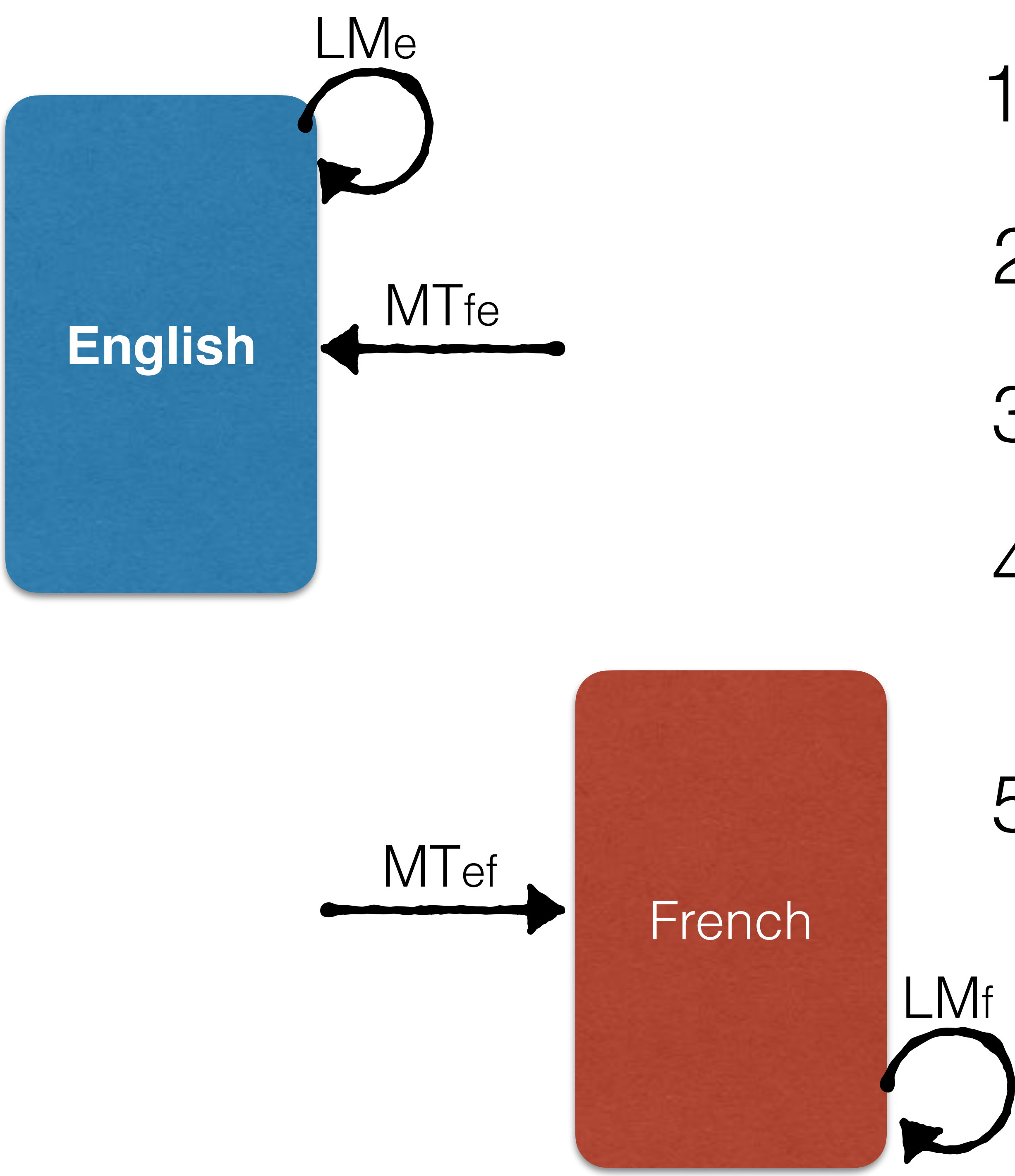
1. Embeddings + Unsup. BLI
2. BLI —> Word Translations
3. Train MT$_{fe}$ and MT$_{ef}$ systems
4. Meanwhile, use unsupervised objectives (denoising LM)
5. Iterate

Unsupervised MT
(Lample et al. and Artetxe et al. 2018)



1. Embeddings + Unsup. BLI
2. BLI —> Word Translations
3. Train MT_{fe} and MT_{ef} systems
4. Meanwhile, use unsupervised objectives (denoising LM)
5. Iterate

Unsupervised MT
(Lample et al. and Artetxe et al. 2018)



1. Embeddings + Unsup. BLI
2. BLI —> Word Translations
3. Train MT_{fe} and MT_{ef} systems
4. Meanwhile, use unsupervised objectives (denoising LM)
5. Iterate

NMT: the biggest success story of NLP Deep Learning

NMT: the biggest success story of NLP Deep Learning

NMT: the biggest success story of NLP Deep Learning

Neural Machine Translation went from a fringe research activity in 2014 to the leading standard method in 2016

NMT: the biggest success story of NLP Deep Learning

Neural Machine Translation went from a fringe research activity in 2014 to the leading standard method in 2016

- 2014: First seq2seq paper published

NMT: the biggest success story of NLP Deep Learning

Neural Machine Translation went from a fringe research activity in 2014 to the leading standard method in 2016

- 2014: First seq2seq paper published
- 2016: Google Translate switches from SMT to NMT

NMT: the biggest success story of NLP Deep Learning

Neural Machine Translation went from a fringe research activity in 2014 to the leading standard method in 2016

- 2014: First seq2seq paper published
- 2016: Google Translate switches from SMT to NMT
- **This is amazing!**

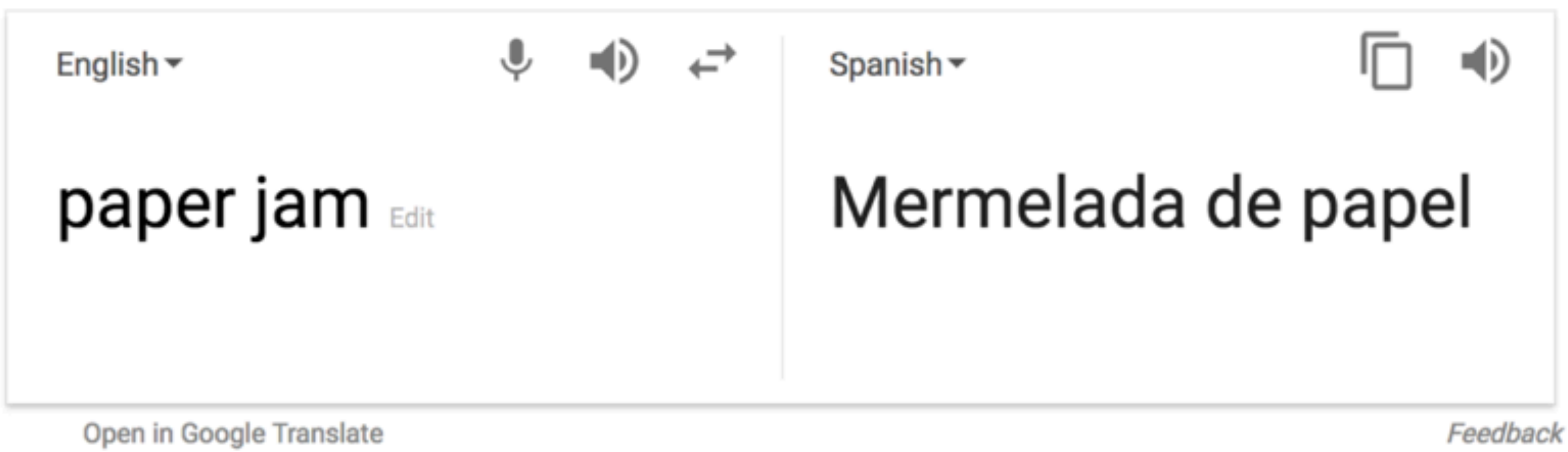
NMT: the biggest success story of NLP Deep Learning

Neural Machine Translation went from a fringe research activity in 2014 to the leading standard method in 2016

- 2014: First seq2seq paper published
- 2016: Google Translate switches from SMT to NMT
- **This is amazing!**
- SMT systems, built by hundreds of engineers over many years, outperformed by NMT systems trained by a handful of engineers in a few months

So is Machine Translation solved?

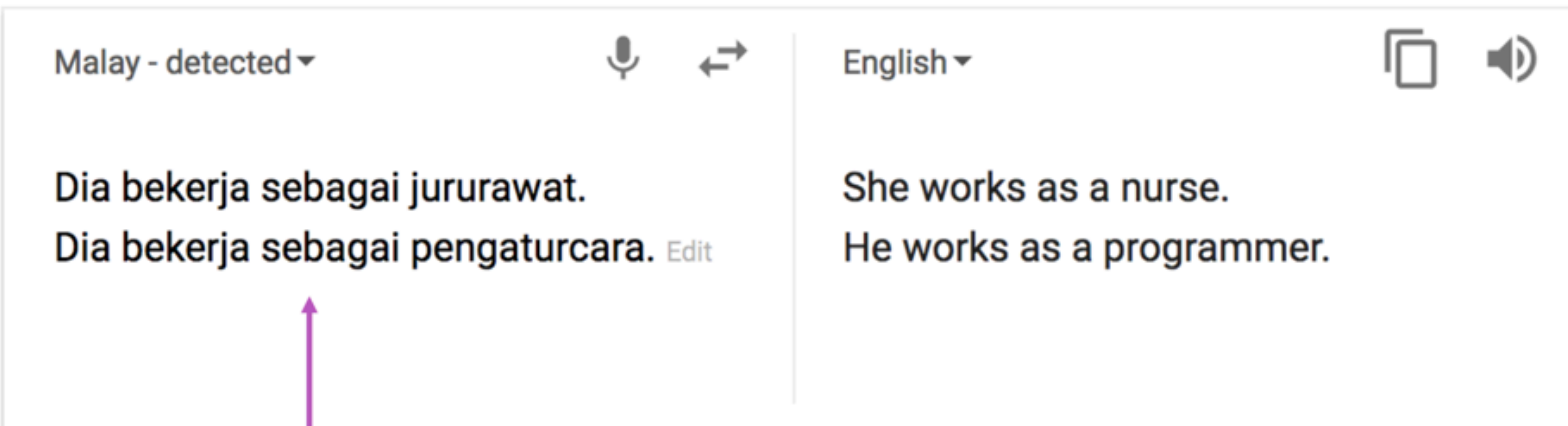
- Nope!
- Using common sense is still hard



?

So is Machine Translation solved?

- Nope!
- NMT picks up biases in training data

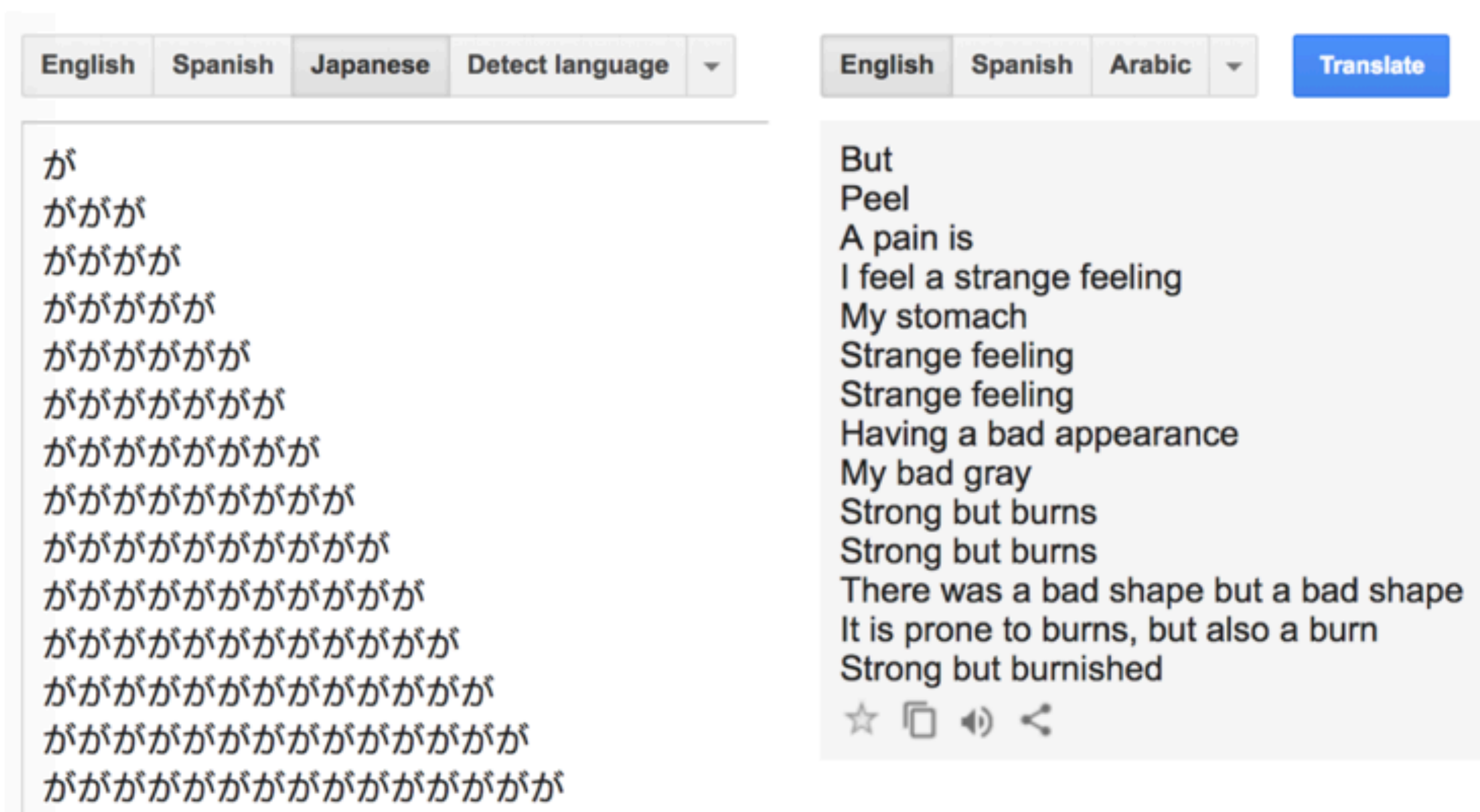


Didn't specify gender

Source: https://hackernoon.com/bias-sexist-or-this-is-the-way-it-should-be-ce1f7c8c683c

So is Machine Translation solved?

- Nope!
- Uninterpretable systems do strange things



Source: http://languagelog.ldc.upenn.edu/nll/?p=35120#more-35120