TTIC 31190:
Natural Language Processing

Kevin Gimpel

Spring 2018

Lecture 16:
Machine Translation

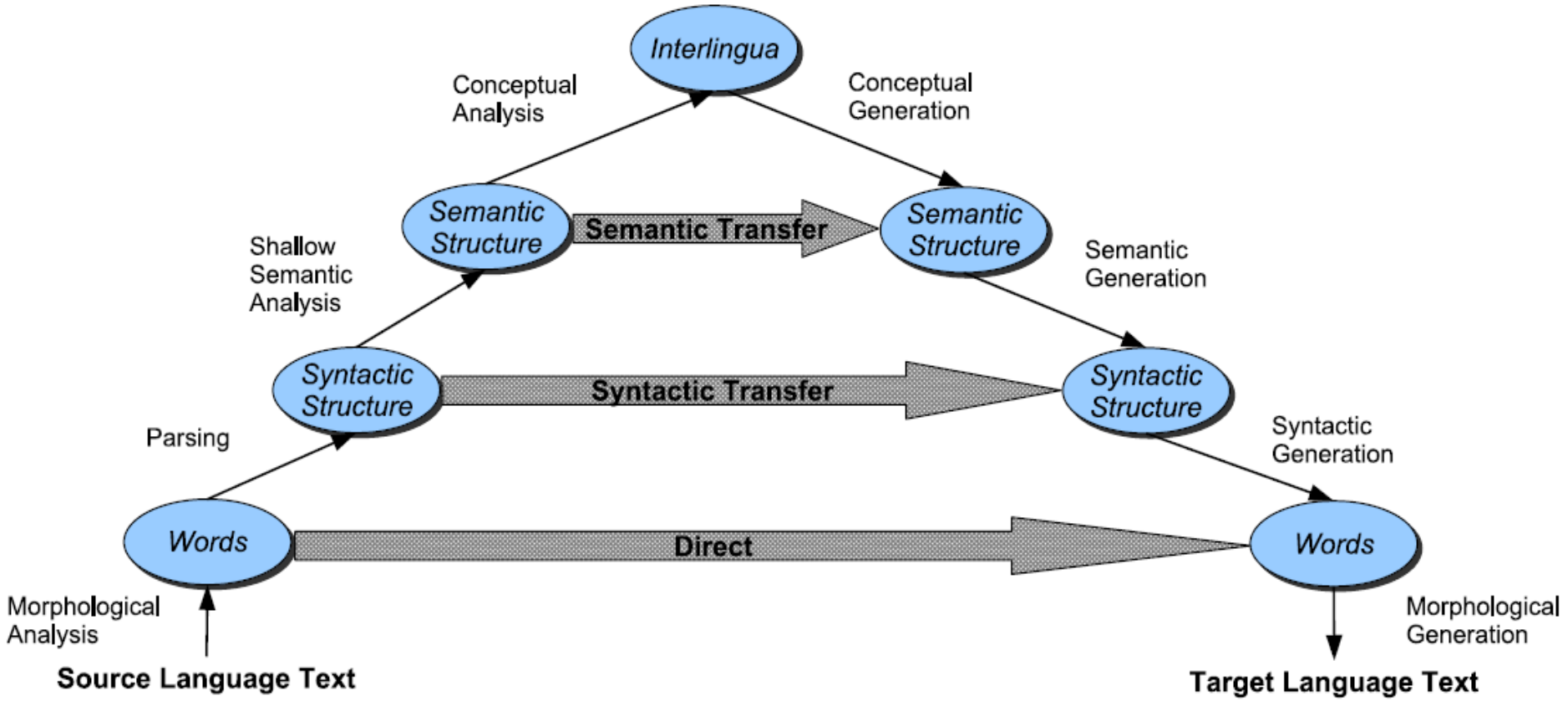
Roadmap

- words, morphology, lexical semantics
- text classification
- simple neural methods for NLP
- language modeling and word embeddings
- recurrent/recursive/convolutional networks in NLP
- sequence labeling, HMMs, dynamic programming
- syntax and syntactic parsing
- machine translation
- semantics

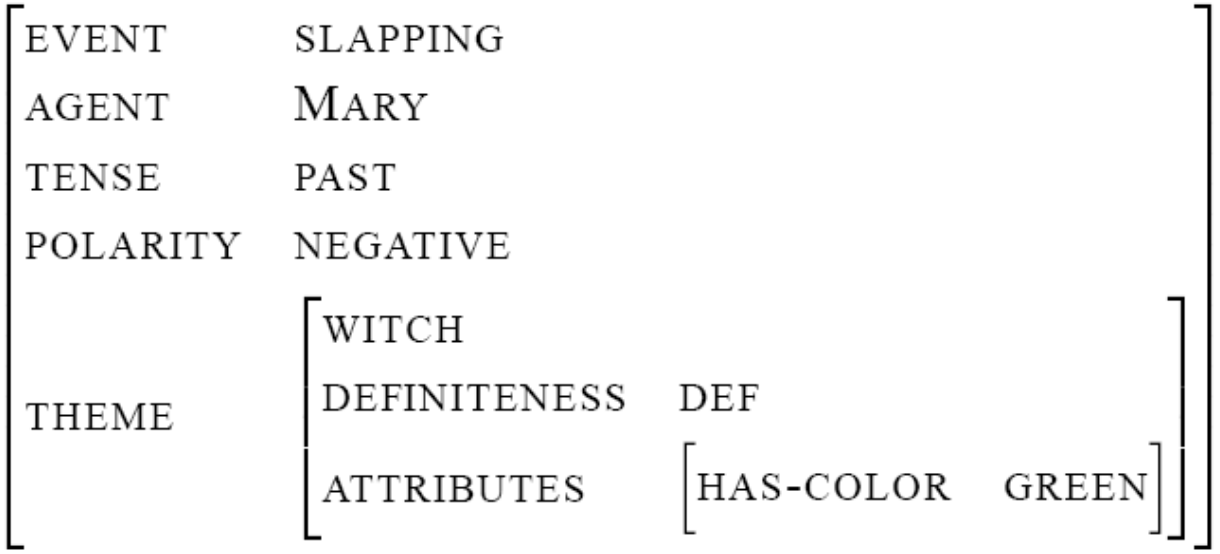
People rely on machine translation!

People rely on machine translation!

Approaches to Machine Translation: The Vauquois Triangle

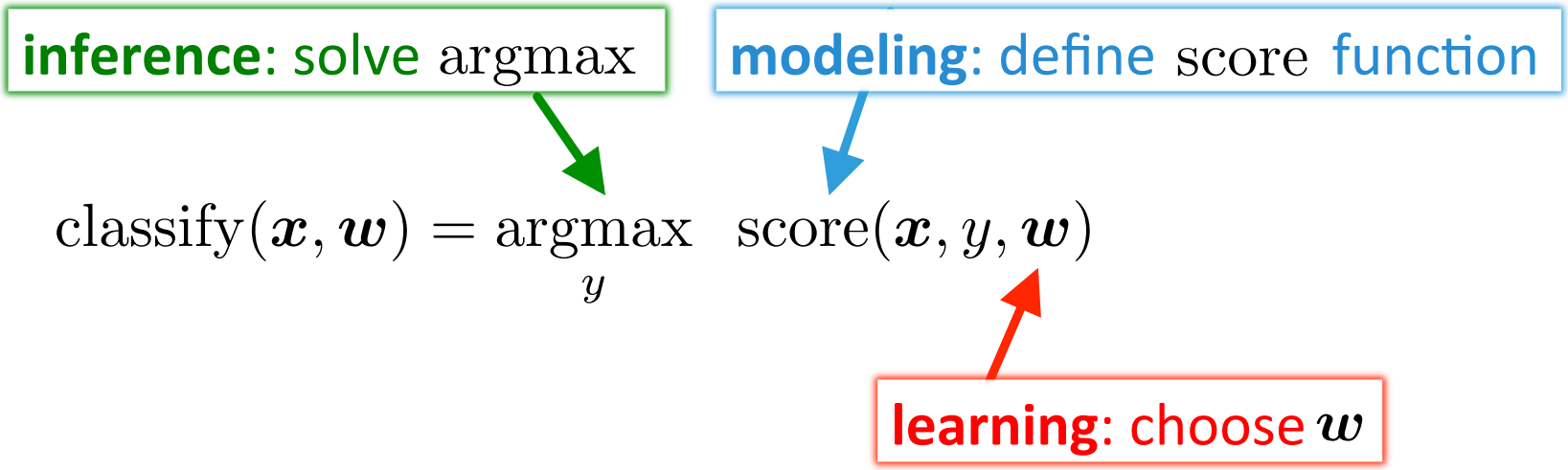


Interlingua Example



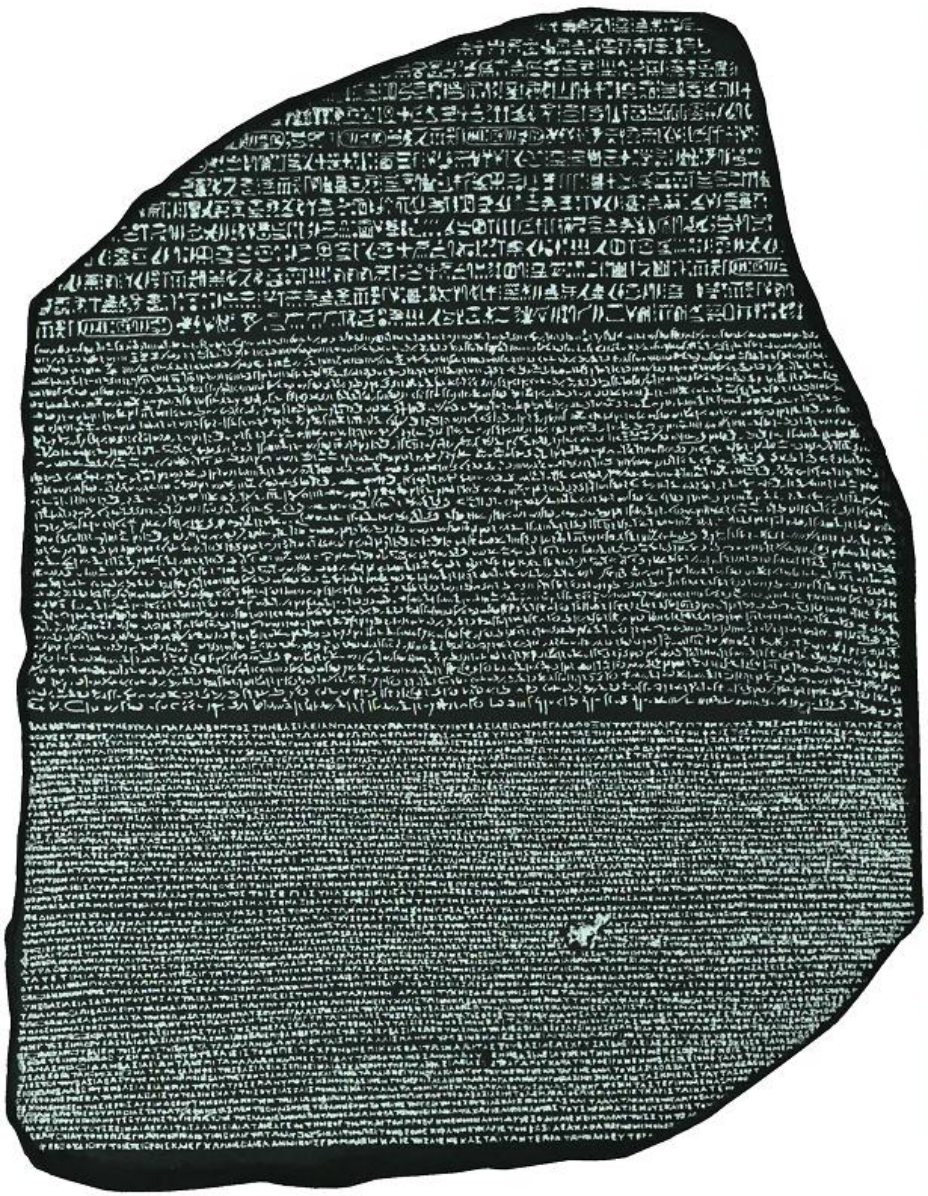
Interlingual representation of *Mary did not slap the green witch*.

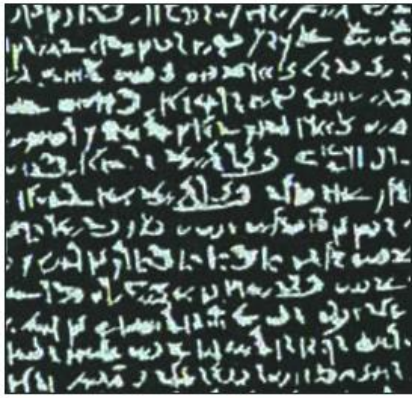
Our Classification Framework for Machine Translation



- modern MT systems are **data-driven**
- first we need data!

Data?





Data?

碟頭飯 RICE PLATE

揚州炒飯	Yang Chow Fried Rice	7.95
咸魚雞粒炒飯	Salted Fish w/ Chicken Fried Rice	8.95
油雞飯	Soy Chicken Rice	5.95
滑雞菜遠飯	Chicken with Vegetable on Rice	5.95
粟米雞扒飯	Chicken with Cream Corn on Rice	5.95
豉椒雞球飯	Chicken W/ Black Bean Sauce	5.95
涼瓜牛肉飯	Beef with Bitter Melon on Rice	5.95
菜遠牛肉飯	Beef with Vegetable on Rice	5.95
牛腩飯	Beef Stew on Rice	6.95
滑蛋牛肉飯	Beef with Egg on Rice	5.95
滑蛋蝦仁飯	Shrimp with Egg on Rice	6.95
鮮蝦菜遠飯	Shrimp with Vegetable on Rice	6.95
魚片菜遠飯	Fish with Vegetable on Rice	6.95
咖哩尤魚飯	Curry Squid on Rice	6.95
滑蛋叉燒飯	BBQ Pork with Egg on Rice	5.95
肉片豆腐飯	Pork with Tofu on Rice	5.95

粥品 CONGEE

白粥	Plain Congee	2.50
皮蛋肉片粥	Preserve Egg w/ Pork Congee	5.50
生滾牛肉粥	Beef Congee	5.50
魚片粥	Fish Congee	5.95
滑雞粥	Chicken Congee	5.50

Chinese Menu

Kings Garden

球記-皇家園

Authentic Chinese Food

TEL: (614) 793-2234

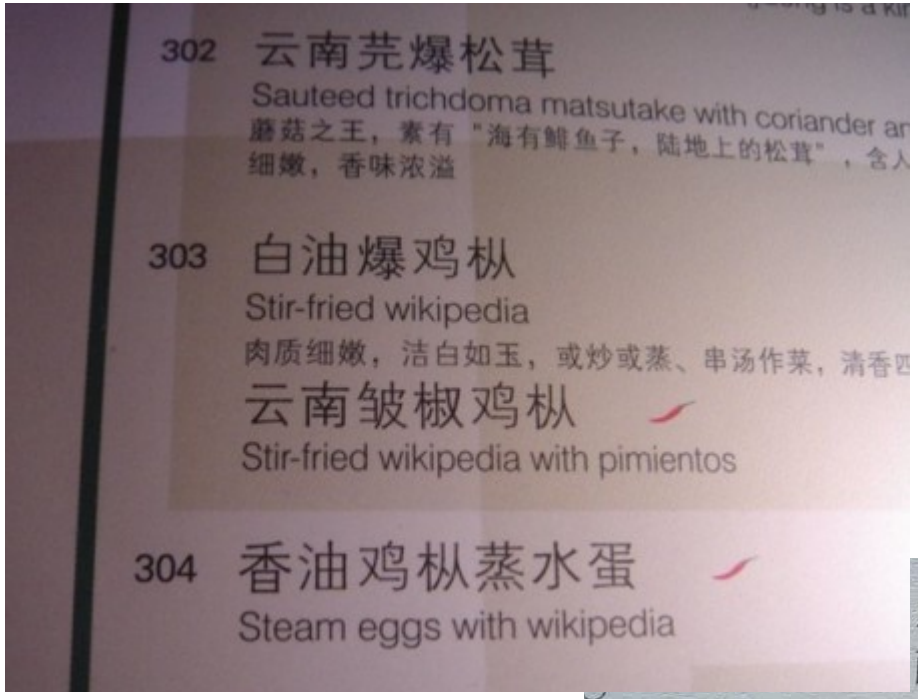
7726 Sawmill Rd.
Dublin, Ohio 43017
(Old Sawmill Sq. Shopping Center)

OPEN HOUR

Mon	Close
Tues - Sat	11:00 am to 10:00 pm
Sun	11:00 am to 9:00 pm

Catering available.

Data?


302 云南芫爆松茸
Sauteed trichdoma matsutake with coriander ar
蘑菇之王，素有"海有鲱鱼子，陆地上的松茸"，含人
细嫩，香味浓溢
303 白油爆鸡枞
Stir-fried wikipedia
肉质细嫩，洁白如玉，或炒或蒸、串汤作菜，清香
云南皱椒鸡枞
Stir-fried wikipedia with pimientos
304 香油鸡枞蒸水蛋
Steam eggs with wikipedia


potato wedges ¥ 15/例
濃湯 Gream of pumpkin soup ¥ 15/例
薩角 India samosa ¥ 25/例
6. 意式火腿面包棒 Italian ham bread ¥ 15/例
7. 田園沙拉 Garden salad ¥ 15/例
8. 三明治 Sand wiches ¥ 15/和
(培根/薩拉米/吞拿魚/火腿) (Bacon/Salami/Tuna/Ham)
9. 香烤魷魚圈 BBQ wikipedia ¥ 20/例
10. 串烤牛小排 BBQ beef and vegetables ¥ 20/例
11. 水牛城香辣鷄翅 Kookaburra wings ¥ 25/6
12. 德國烤腸 German BBQ Sausage ¥ 30/例
13. 香蒜面包 Garlic butter bread ¥ 10/3

Data?

United Nations

EUROPEAN PARLIAMENT

PARLIAMENT *of* CANADA

Also:

- news articles
- company websites
- laws & patents
- subtitles

Parallel Data

- **parallel data**: bilingual data that is naturally aligned at some level
- usually aligned at the document level
- sentence-level alignments are generated automatically
  - how might you design an algorithm for this?
  - it can be done well without dictionaries!
  - can throw out sentences that don't align with anything

Learning from Parallel Sentences

Chickasaw	English
1. Ofi 'at kowi 'ã lhiyohli	1. The dog chases the cat
2. Kowi 'at ofi 'ã lhiyohli	2. The cat chases the dog
3. Ofi 'at shoha	3. The dog stinks

Learning from Parallel Sentences

Chickasaw	English
1. Ofi 'at kowi 'ã lhiyohli	1. The dog chases the cat
2. Kowi 'at ofi 'ã lhiyohli	2. The cat chases the dog
3. Ofi 'at shoha	3. The dog stinks

Machine Translation Evaluation

- human judgments are ideal, but expensive
  - what other problems are there with human judgments?

- we need automatic evaluation metrics
  - BLEU (BiLingual Evaluation Understudy), Papineni et al. (2002)
    - compare *n*-gram overlap between system output and human-produced translation
    - correlates with human judgments surprisingly well, but only at the document level (not sentence level!)
  - other metrics do soft matching based on stemming and synonyms from WordNet
  - this is not a solved problem!

Statistical Machine Translation

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

When I look at an article in Arabic, 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, 1947

A Statistical Approach to Machine Translation

Peter F. Brown, John Cocke, Stephen A. Della Pietra, Vincent J. Della Pietra, Fredrick Jelinek, John D. Lafferty, Robert L. Mercer, and Paul S. Roossin

IBM

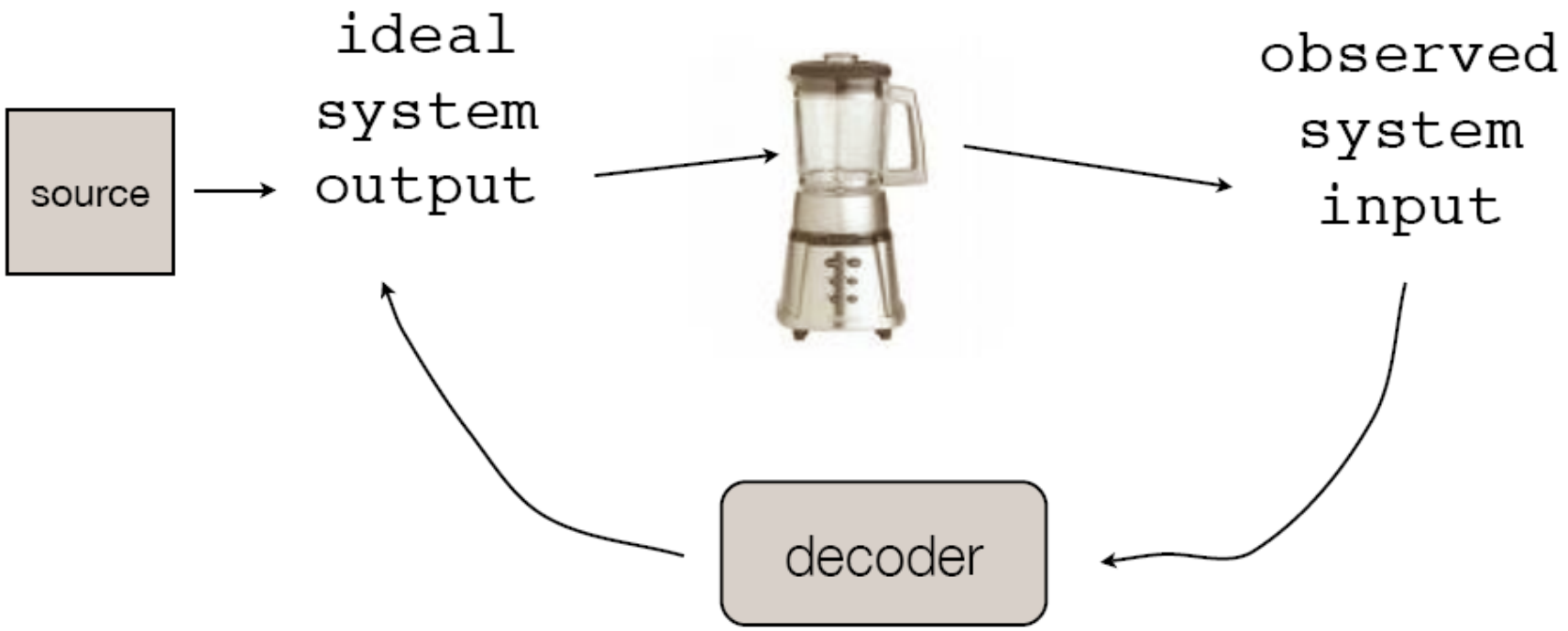
Thomas J. Watson Research Center

Yorktown Heights, NY

In this paper, we present a statistical approach to machine translation. We describe the application of our approach to translation from French to English and give preliminary results.

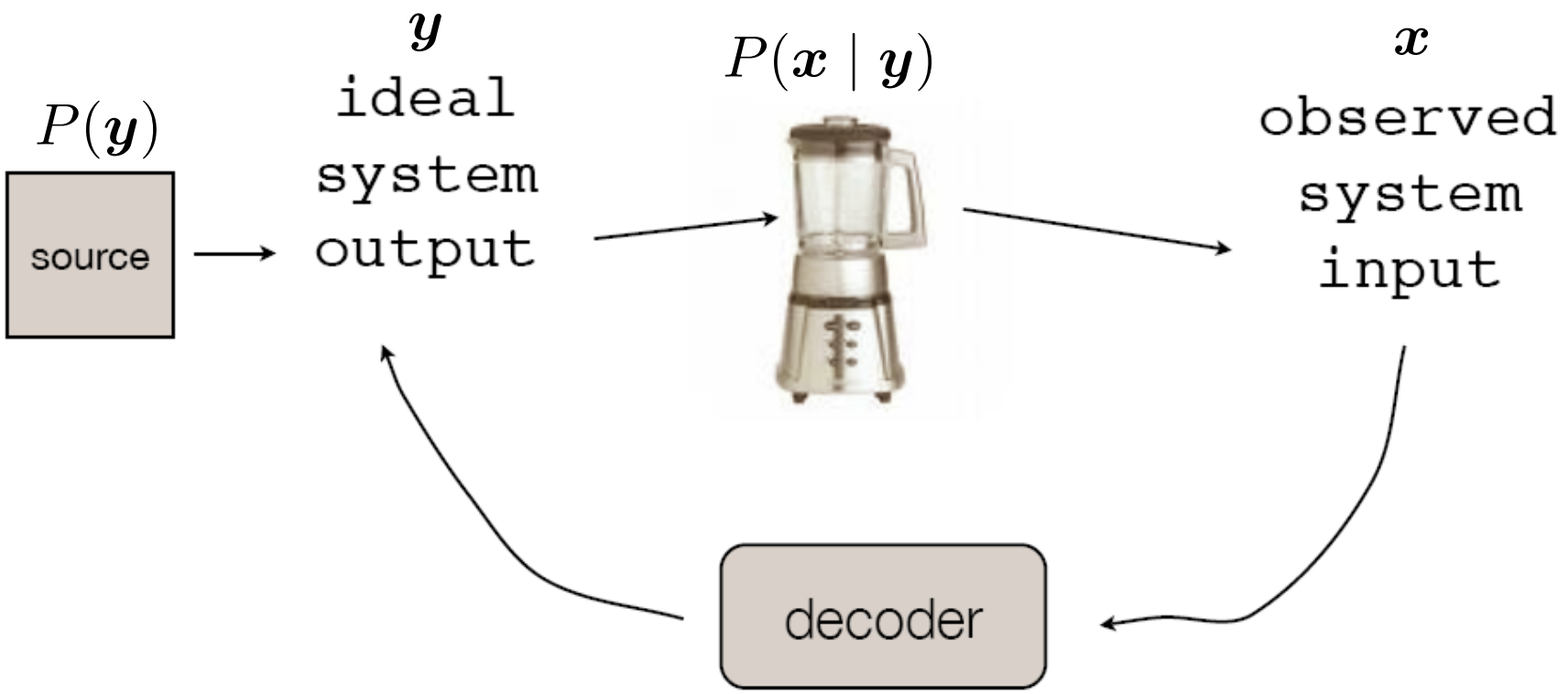
Brown et al. (1990)

Noisy Channel Model



from Noah Smith

Noisy Channel Model for Translating French (***x***) to English (***y***)



$$y^* = \operatorname*{argmax}_y P(y \mid x)$$

$$= \operatorname*{argmax}_y \frac{P(x \mid y)P(y)}{P(x)}$$

$$= \operatorname*{argmax}_y P(x \mid y)P(y)$$

from Noah Smith

Modeling for the Noisy Channel

- we need to model two probability distributions:
  - $P(\boldsymbol{y})$ should favor fluent translations
  - $P(\boldsymbol{x} \mid \boldsymbol{y})$ should favor accurate/faithful translations

Modeling for the Noisy Channel

- we need to model two probability distributions:
  - $P(\boldsymbol{y})$ should favor fluent translations
  - $P(\boldsymbol{x} \mid \boldsymbol{y})$ should favor accurate/faithful translations

- let's start with $P(\boldsymbol{y})$
  - how do we compute the probability of an English sentence?
  - language modeling is an important part of MT

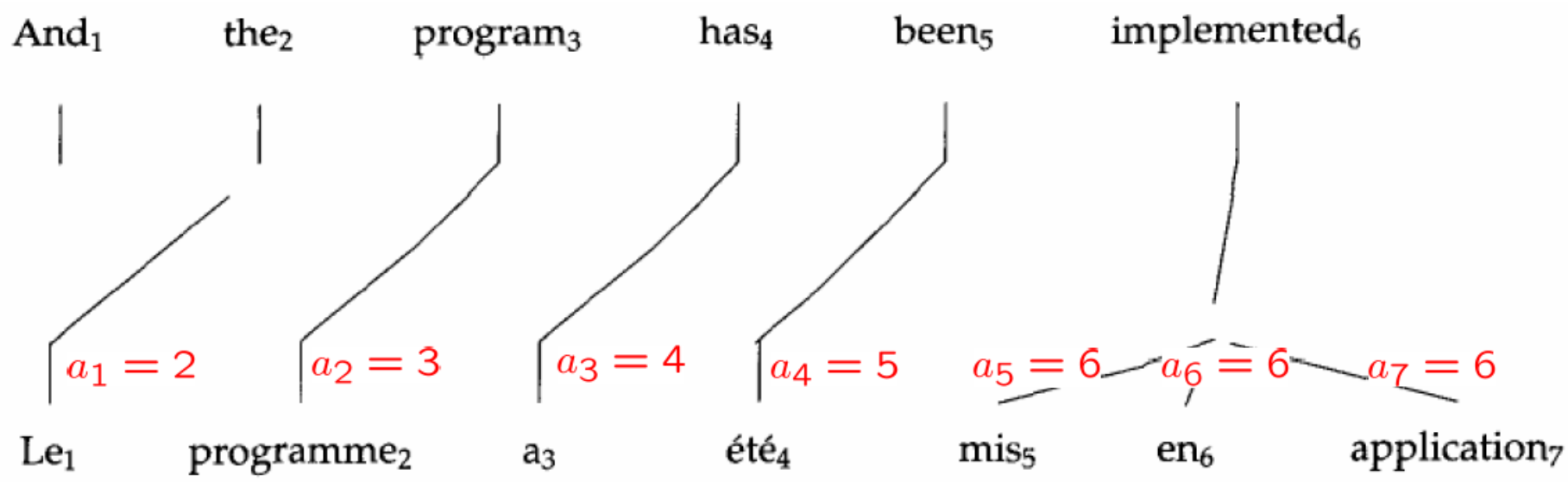
Word Alignments

And$_1$ the$_2$ program$_3$ has$_4$ been$_5$ implemented$_6$

Le$_1$ programme$_2$ a$_3$ été$_4$ mis$_5$ en$_6$ application$_7$

Word Alignments

$$\boldsymbol{a} = \langle a_1, \ldots, a_{|\boldsymbol{x}|} \rangle$$



- a is a "hidden" variable (not part of training data)
- for each French word, it holds the index of the aligned English word (or NULL)

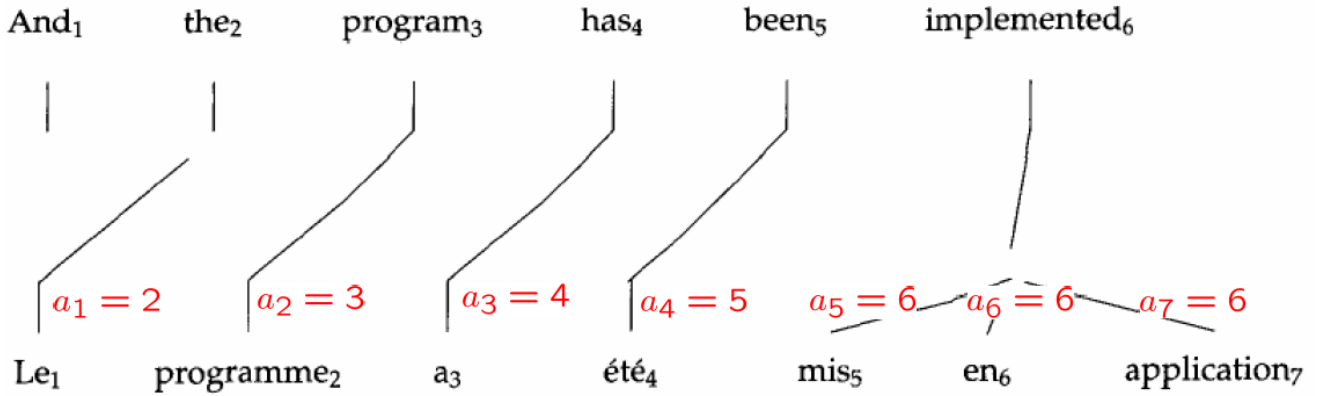
- remember: our goal was to model $P(\boldsymbol{x} \mid \boldsymbol{y})$
- why would we introduce a hidden variable?
  - to make it "easier" to define the model
  - we often want to share information across instances in our data
    - latent variables are a natural way to capture this
    - think of clustering (some points come from the same cluster)

Alignments as Hidden Variables

- for simplicity, assume that each French word aligns to 1 English word (or to NULL)

- analogy to clustering:
  - each data point has 1 vote which it can distribute among all the clusters
  - here, each French word has 1 vote which it can distribute among all the English words or NULL

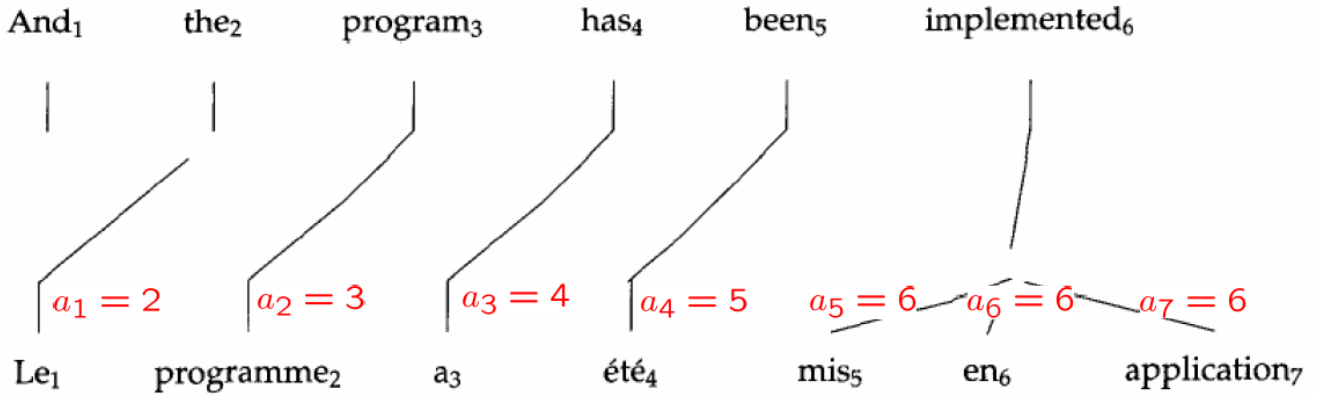
Modeling Alignments: IBM Model 1



$$P(\boldsymbol{x}, \boldsymbol{a} \mid \boldsymbol{y}) = \prod_{j=1}^{|\boldsymbol{x}|} P(a_j) P(x_j \mid y_{a_j})$$

$$= \prod_{j=1}^{|\boldsymbol{x}|} \frac{1}{|\boldsymbol{y}| + 1} P(x_j \mid y_{a_j})$$

Modeling Alignments: IBM Model 1

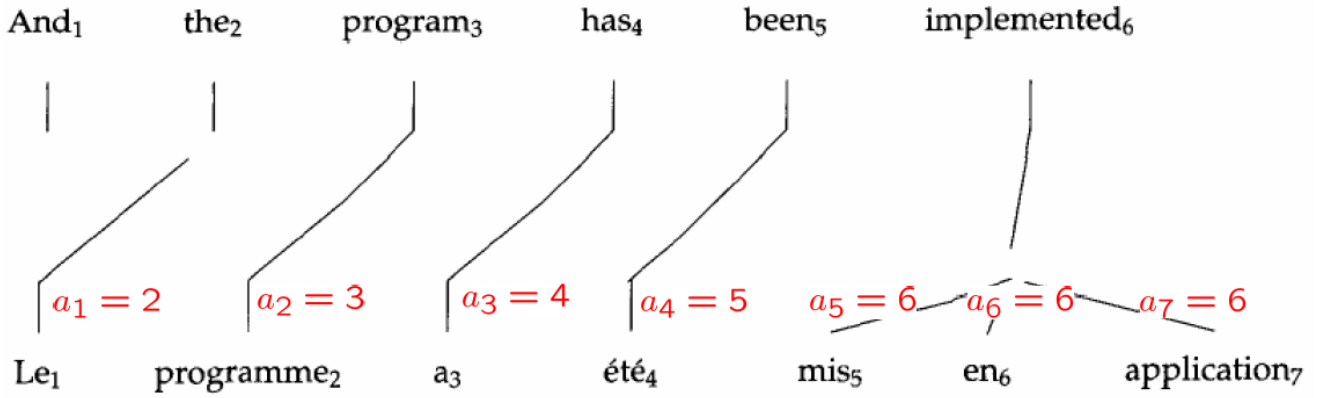


$$P(\boldsymbol{x}, \boldsymbol{a} \mid \boldsymbol{y}) = \prod_{j=1}^{|\boldsymbol{x}|} P(a_j) P(x_j \mid y_{a_j})$$

$$= \prod_{j=1}^{|\boldsymbol{x}|} \frac{1}{|\boldsymbol{y}| + 1} P(x_j \mid y_{a_j})$$

- How do we obtain $P(\boldsymbol{x} \mid \boldsymbol{y})$?

Modeling Alignments: IBM Model 1

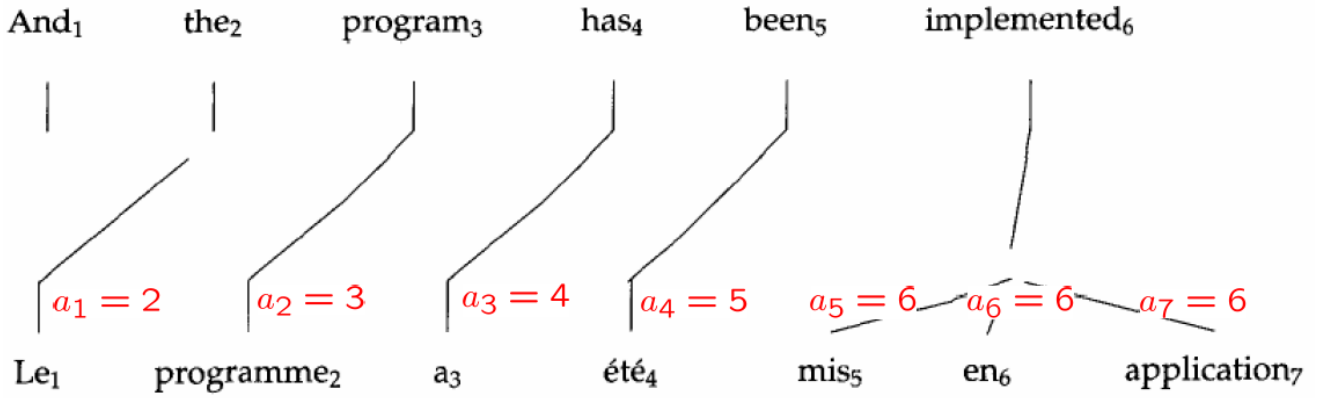


$$P(\boldsymbol{x}, \boldsymbol{a} \mid \boldsymbol{y}) = \prod_{j=1}^{|\boldsymbol{x}|} P(a_j) P(x_j \mid y_{a_j})$$

$$= \prod_{j=1}^{|\boldsymbol{x}|} \frac{1}{|\boldsymbol{y}| + 1} P(x_j \mid y_{a_j})$$

- How do we obtain $P(\boldsymbol{x} \mid \boldsymbol{y})$?
- Sum over all alignments: $P(\boldsymbol{x} \mid \boldsymbol{y}) = \sum_{\boldsymbol{a}} P(\boldsymbol{x}, \boldsymbol{a} \mid \boldsymbol{y})$

Modeling Alignments: IBM Model 1



$$P(\boldsymbol{x}, \boldsymbol{a} \mid \boldsymbol{y}) = \prod_{j=1}^{|\boldsymbol{x}|} P(a_j) P(x_j \mid y_{a_j})$$

$$= \prod_{j=1}^{|\boldsymbol{x}|} \frac{1}{|\boldsymbol{y}| + 1} P(x_j \mid y_{a_j})$$

Parameters in the model,
learned using expectation maximization

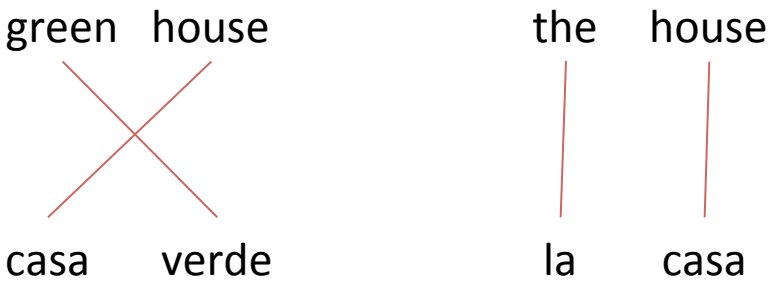
Aside: are alignments always hidden?

- certain small parallel corpora have been hand-aligned
- issues with this?
  - annotators don't agree
  - we have lots of parallel text, very little is hand-aligned
  - for some language pairs, we will never have manual alignments

- word alignment has become a fundamental part of MT, and we need unsupervised learning to solve it!

IBM Model 1 Example

- Consider a training set of two sentence pairs:



Initial Parameter Estimates:

t(casa|green) = $\frac{1}{3}$	t(verde|green) = $\frac{1}{3}$	t(la|green) = $\frac{1}{3}$
t(casa|house) = $\frac{1}{3}$	t(verde|house) = $\frac{1}{3}$	t(la|house) = $\frac{1}{3}$
t(casa|the) = $\frac{1}{3}$	t(verde|the) = $\frac{1}{3}$	t(la|the) = $\frac{1}{3}$

$t(f \mid e)$

= probability of translating *e* into *f*

After 1 iteration of EM:

t(casa|green) = $\frac{1/2}{1} = \frac{1}{2}$	t(verde|green) = $\frac{1/2}{1} = \frac{1}{2}$	t(la|green) = $\frac{0}{1} = 0$
t(casa|house) = $\frac{1}{2} = \frac{1}{2}$	t(verde|house) = $\frac{1/2}{2} = \frac{1}{4}$	t(la|house) = $\frac{1/2}{2} = \frac{1}{4}$
t(casa|the) = $\frac{1/2}{1} = \frac{1}{2}$	t(verde|the) = $\frac{0}{1} = 0$	t(la|the) = $\frac{1/2}{1} = \frac{1}{2}$

The Mathematics of Statistical Machine Translation: Parameter Estimation

Peter F. Brown*
IBM T.J. Watson Research Center

Stephen A. Della Pietra*
IBM T.J. Watson Research Center

Vincent J. Della Pietra*
IBM T.J. Watson Research Center

Robert L. Mercer*
IBM T.J. Watson Research Center

We describe a series of five statistical models of the translation process and give algorithms for estimating the parameters of these models given a set of pairs of sentences that are translations of one another. We define a concept of word-by-word alignment between such pairs of sentences. For any given pair of such sentences each of our models assigns a probability to each of the possible word-by-word alignments. We give an algorithm for seeking the most probable of these alignments. Although the algorithm is suboptimal, the alignment thus obtained accounts well for the word-by-word relationships in the pair of sentences. We have a great deal of data in French and English from the proceedings of the Canadian Parliament. Accordingly, we have restricted our work to these two languages; but we feel that because our algorithms have minimal linguistic content they would work well on other pairs of languages. We also feel, again because of the minimal linguistic content of our algorithms, that it is reasonable to argue that word-by-word alignments are inherent in any sufficiently large bilingual corpus.

Brown et al. (1993)

IBM Model 1

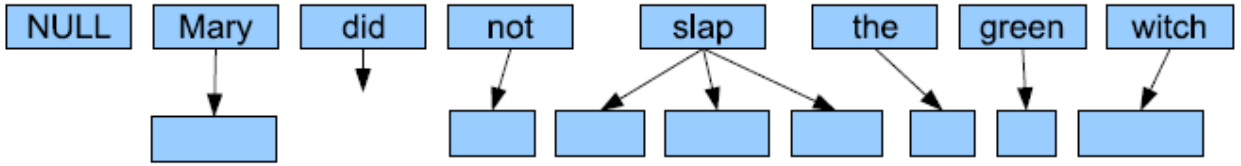
$$P(\boldsymbol{x}, \boldsymbol{a} \mid \boldsymbol{y}) = \prod_{j=1}^{|\boldsymbol{x}|} \frac{1}{|\boldsymbol{y}| + 1} P(x_j \mid y_{a_j})$$

IBM Model 2

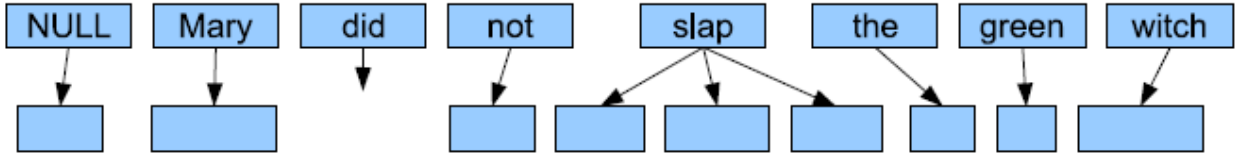
$$P(\boldsymbol{x}, \boldsymbol{a} \mid \boldsymbol{y}) = \prod_{j=1}^{|\boldsymbol{x}|} P(a_j \mid j, |\boldsymbol{x}|, |\boldsymbol{y}|) P(x_j \mid y_{a_j})$$

IBM Model 3

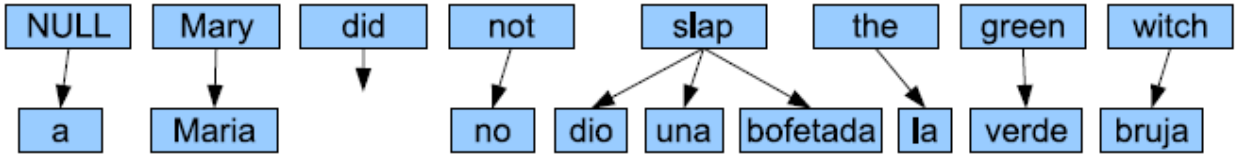
Step 1: Choose fertility for each English word



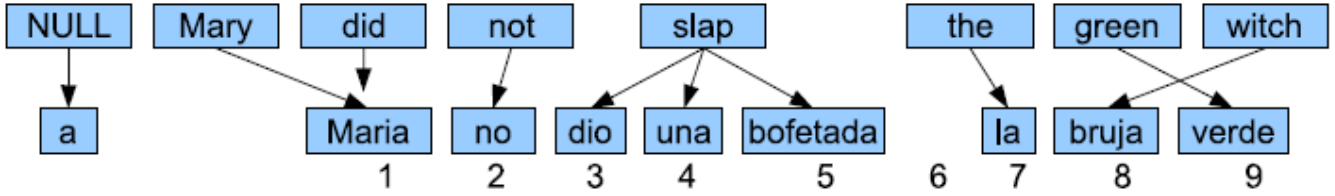
Step 2: Choose fertility for NULL



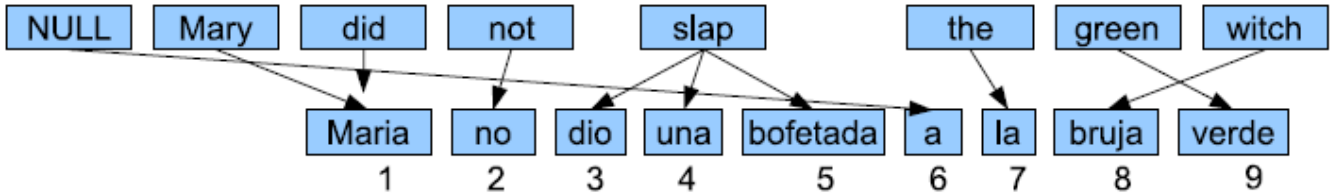
Step 3: Create Spanish words by translating aligned English word



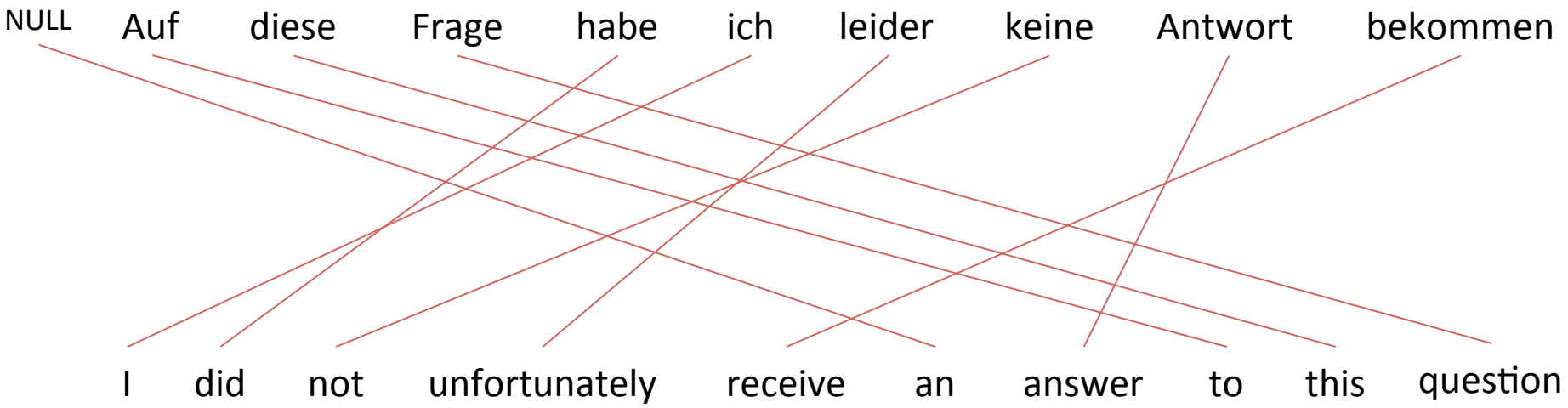
Step 4: Move the Spanish words into final slots



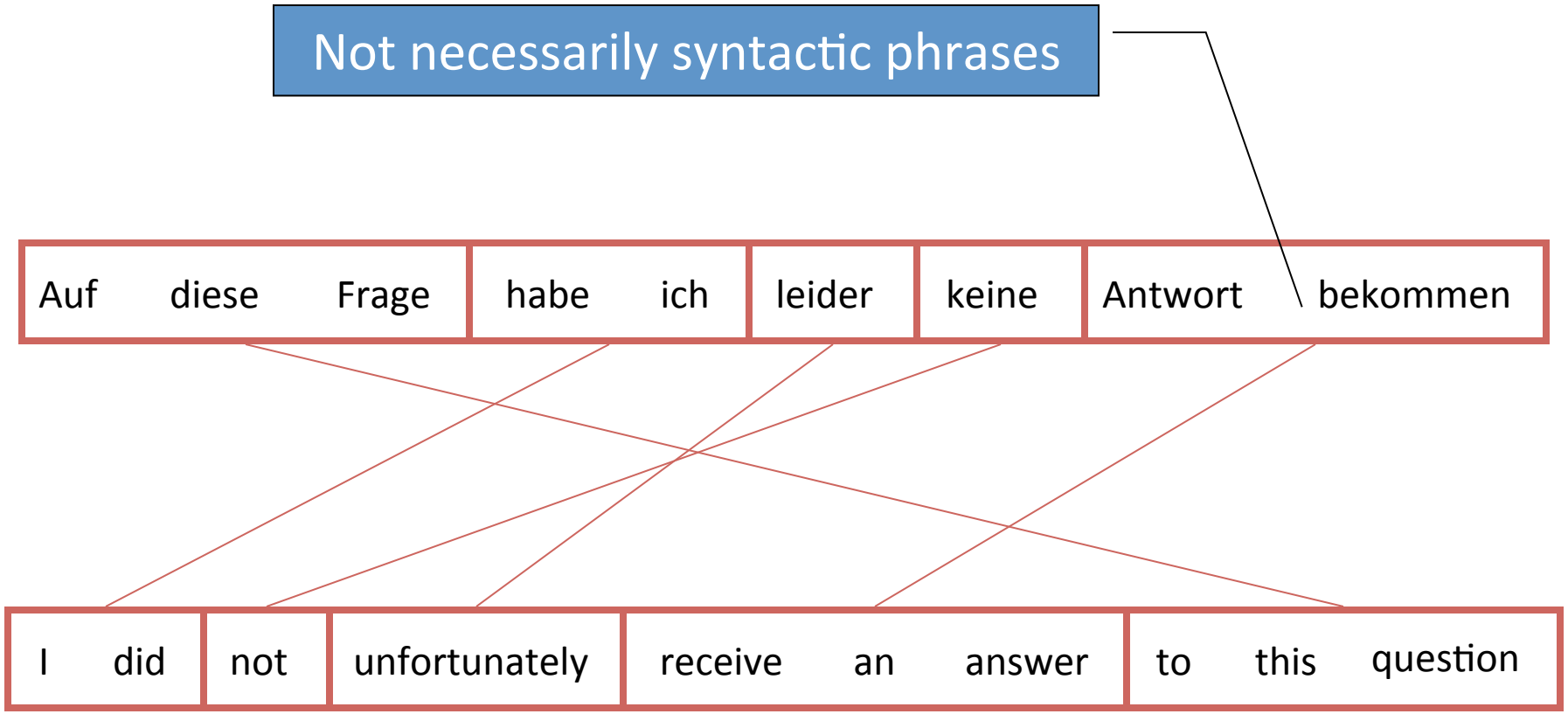
Step 4: Move spurious Spanish words into unclaimed slots



Moving to Phrases



Moving to Phrases

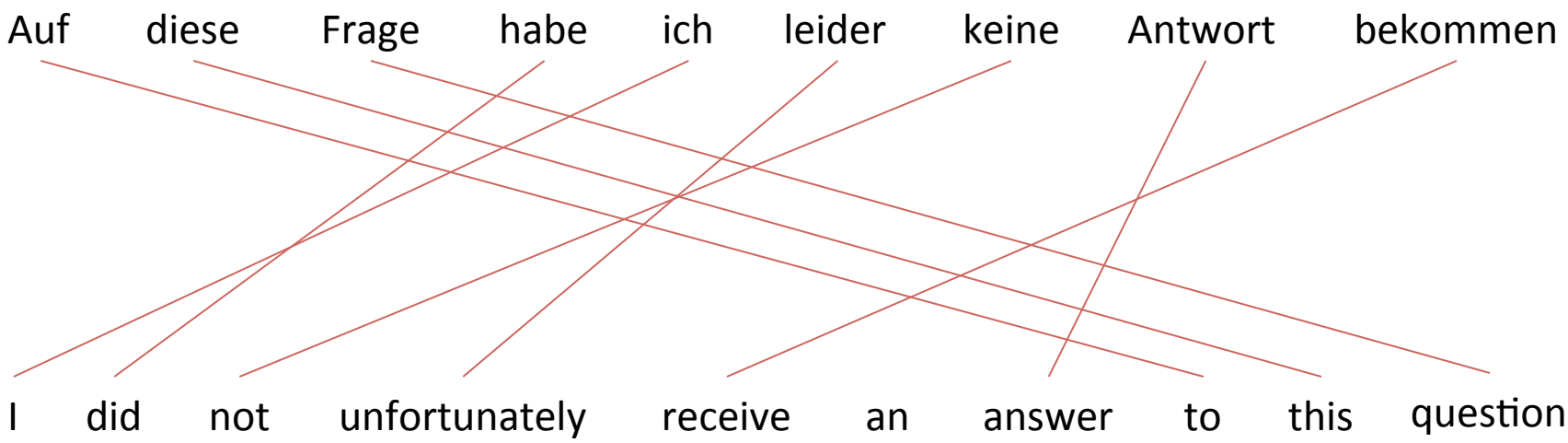


“Phrase-Based” Translation

- Relies on a **phrase table**
  - massive bilingual phrase dictionary, with probabilities
- To build:
  - Find the best word alignment for each sentence pair
  - Extract all phrase pairs **consistent** with the word alignment
  - Compute probabilities using relative frequency estimation

Phrase-Based Translation

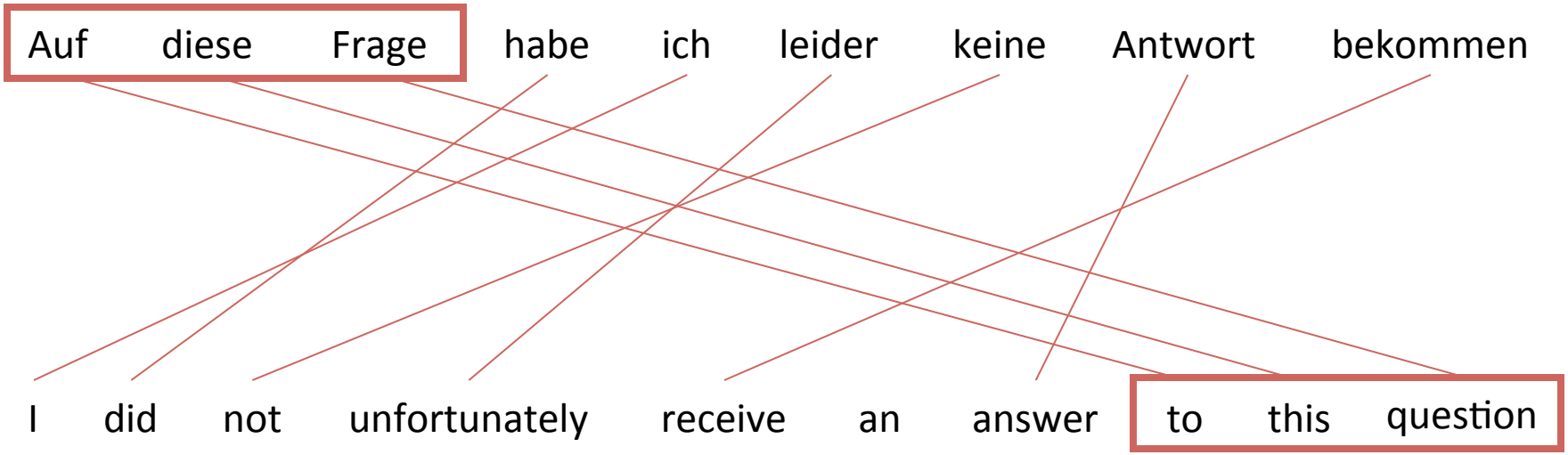
- Relies on a **phrase table**
  - massive bilingual phrase dictionary, with probabilities
- To build:
  - Find the best word alignment for each sentence pair
  - Extract all phrase pairs **consistent** with the word alignment
  - Compute probabilities using MLE



Phrase-Based Translation

- Relies on a **phrase table**
  - massive bilingual phrase dictionary, with probabilities
- To build:
  - Find the best word alignment for each sentence pair
  - Extract all phrase pairs **consistent** with the word alignment
  - Compute probabilities

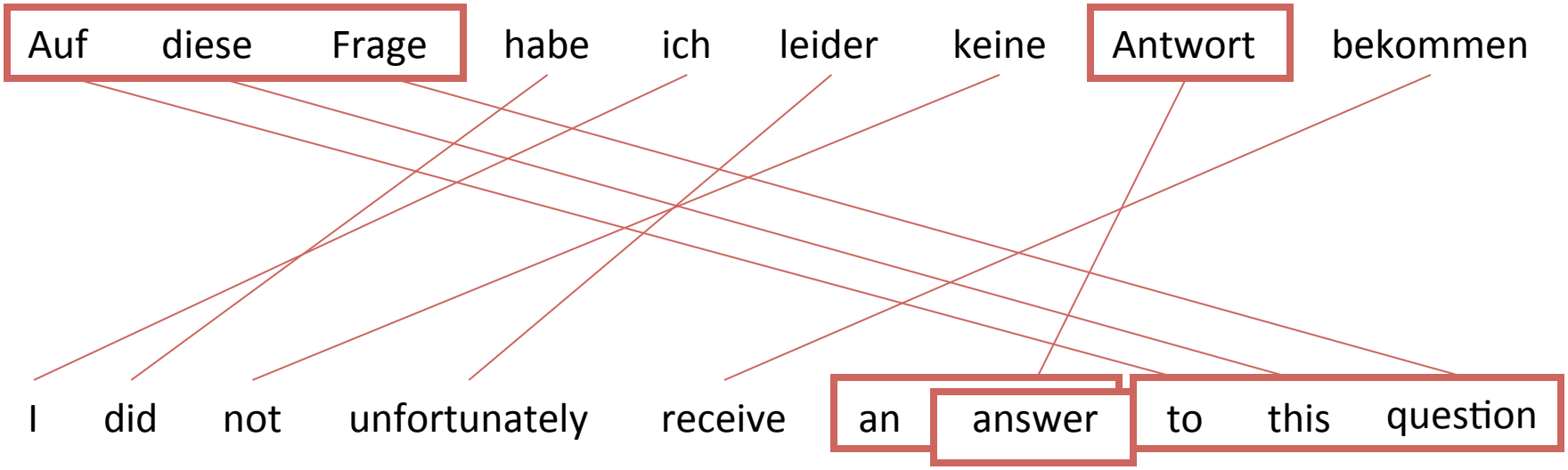
Auf diese Frage	to this question	1.0



Phrase-Based Translation

- Relies on a **phrase table**
  - massive bilingual phrase dictionary, with probabilities
- To build:
  - Find the best word alignment for each sentence pair
  - Extract all phrase pairs **consistent** with the word alignment
  - Compute probabilities

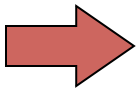
Auf diese Frage	to this question	1.0
Antwort	an answer	1.0
Antwort	answer	1.0
	…	



Phrase-Based Translation

- Relies on a **phrase table**
  - massive bilingual phrase dictionary, with probabilities
- To build:
  - Find the best word alignment for each sentence pair
  - Extract all phrase pairs **consistent** with the word alignment
  - Compute probabilities using MLE:

German	English	Count
Auf diese Frage	to this question	1.0
Antwort	an answer	1.0
Antwort	answer	1.0
…		



German	English	P(e | f)
Auf diese Frage	to this question	1.0
Antwort	an answer	0.5
Antwort	answer	0.5
…		

Statistical Phrase-Based Translation

Philipp Koehn, Franz Josef Och, Daniel Marcu
Information Sciences Institute
Department of Computer Science
University of Southern California
koehn@isi.edu, och@isi.edu, marcu@isi.edu

Koehn et al. (2003)

Adding Syntax: Synchronous Context-Free Grammars

CFG

$$\begin{aligned}
\text{NP} &\longrightarrow \text{DT NPB} \\
\text{NPB} &\longrightarrow \text{JJ NPB} \\
\text{NPB} &\longrightarrow \textbf{NN} \\
\text{DT} &\longrightarrow \text{the} \\
\text{JJ} &\longrightarrow \text{strong} \\
\text{JJ} &\longrightarrow \text{north} \\
\text{NN} &\longrightarrow \text{wind}
\end{aligned}$$

SCFG

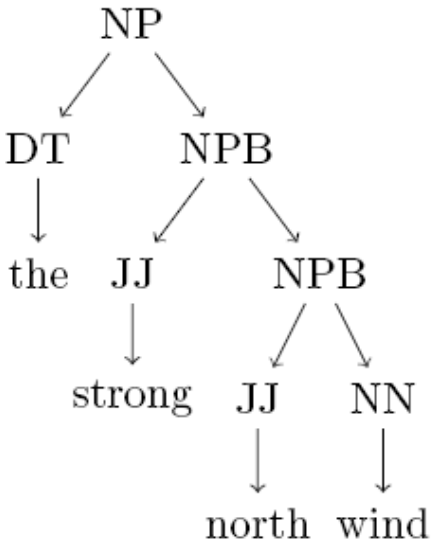
$$\begin{aligned}
\text{NP} &\longrightarrow \text{DT}_{\boxed{1}}\text{NPB}_{\boxed{2}} \;/\; \text{DT}_{\boxed{1}}\text{NPB}_{\boxed{2}} \\
\text{NPB} &\longrightarrow \text{JJ}_{\boxed{1}}\text{NPB}_{\boxed{2}} \;/\; \text{JJ}_{\boxed{1}}\text{NPB}_{\boxed{2}} \\
\text{NPB} &\longrightarrow \text{JJ}_{\boxed{1}}\text{NPB}_{\boxed{2}} \;/\; \text{NPB}_{\boxed{2}}\text{JJ}_{\boxed{1}} \\
\text{NPB} &\longrightarrow \text{NP}_{\boxed{1}} \;/\; \text{NP}_{\boxed{1}} \\
\text{DT} &\longrightarrow \text{the} \;/\; \varepsilon \\
\text{JJ} &\longrightarrow \text{strong} \;/\; \text{呼啸} \\
\text{JJ} &\longrightarrow \text{north} \;/\; \text{北} \\
\text{NN} &\longrightarrow \text{wind} \;/\; \text{风}
\end{aligned}$$

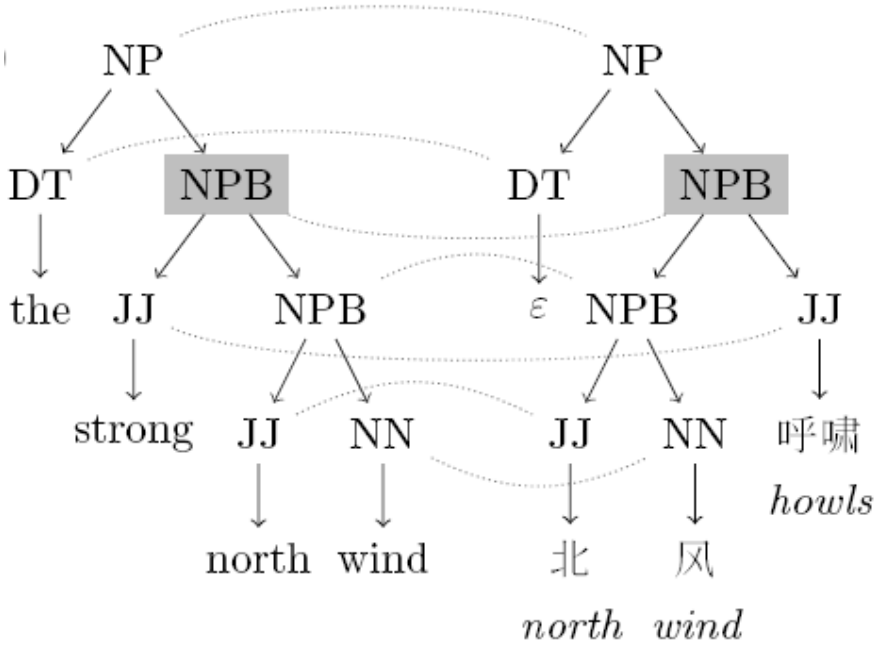
CFG

$$
\begin{aligned}
\text{NP} &\longrightarrow \text{DT NPB} \\
\text{NPB} &\longrightarrow \text{JJ NPB} \\
\text{NPB} &\longrightarrow \textbf{NN} \\
\text{DT} &\longrightarrow \text{the} \\
\text{JJ} &\longrightarrow \text{strong} \\
\text{JJ} &\longrightarrow \text{north} \\
\text{NN} &\longrightarrow \text{wind}
\end{aligned}
$$

SCFG

$$
\begin{aligned}
\text{NP} &\longrightarrow \text{DT}_{\boxed{1}}\text{NPB}_{\boxed{2}} \;/\; \text{DT}_{\boxed{1}}\text{NPB}_{\boxed{2}} \\
\text{NPB} &\longrightarrow \text{JJ}_{\boxed{1}}\text{NPB}_{\boxed{2}} \;/\; \text{JJ}_{\boxed{1}}\text{NPB}_{\boxed{2}} \\
\text{NPB} &\longrightarrow \text{JJ}_{\boxed{1}}\text{NPB}_{\boxed{2}} \;/\; \text{NPB}_{\boxed{2}}\text{JJ}_{\boxed{1}} \\
\text{NPB} &\longrightarrow \text{NP}_{\boxed{1}} \;/\; \text{NP}_{\boxed{1}} \\
\text{DT} &\longrightarrow \text{the} \;/\; \varepsilon \\
\text{JJ} &\longrightarrow \text{strong} \;/\; \text{呼啸} \\
\text{JJ} &\longrightarrow \text{north} \;/\; \text{北} \\
\text{NN} &\longrightarrow \text{wind} \;/\; \text{风}
\end{aligned}
$$





Classification Framework for Machine Translation

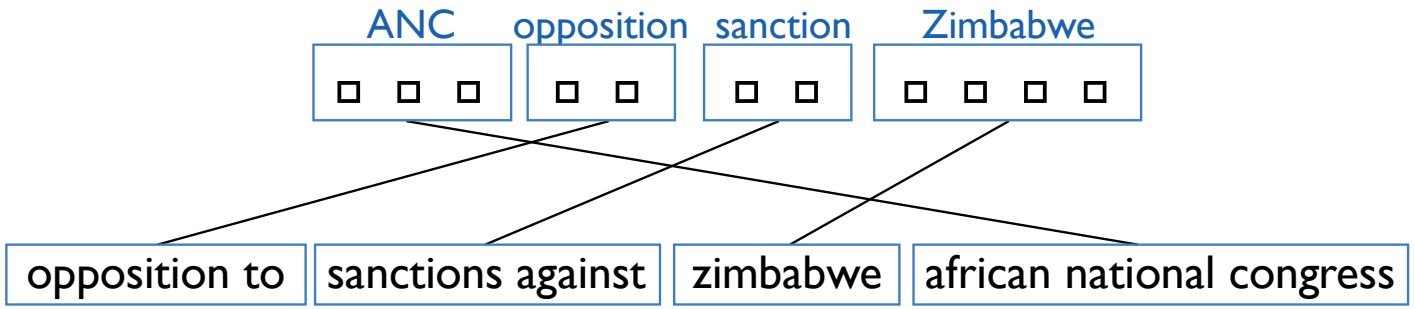
inference: solve argmax

$$\boldsymbol{y}^* = \text{classify}(\boldsymbol{x}, \boldsymbol{w}) = \underset{\boldsymbol{y}}{\text{argmax}}\ \text{score}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{w})$$

- we have a latent variable, so this becomes:

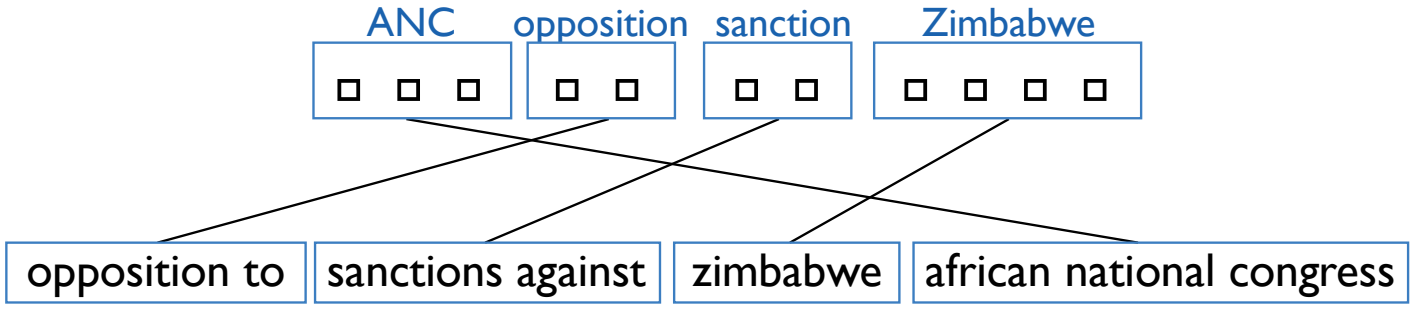
$$\langle \boldsymbol{y}^*, \boldsymbol{h}^* \rangle = \text{classify}(\boldsymbol{x}, \boldsymbol{w}) = \underset{\langle \boldsymbol{y}, \boldsymbol{h} \rangle}{\text{argmax}}\ \text{score}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{h}, \boldsymbol{w})$$

- we maximize over the latent variable *and* the output!
- *h* could be word alignments, phrase segmentations/ alignments, synchronous CFG derivations, etc.



Reference: african national congress opposes sanctions against zimbabwe

- For phrase-based translation, search over:
  - Segmentations into phrases
  - Translations for each phrase
  - Orderings of the translated phrases



Reference: african national congress opposes sanctions against zimbabwe

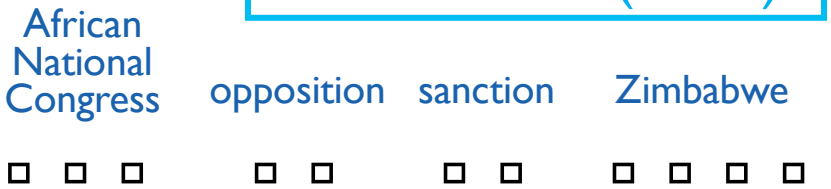
- For phrase-based translation, search over:
  - Segmentations into phrases
  - Translations for each phrase
  - Orderings of the translated phrases

This search problem is NP-hard (Knight, 1999)

Approximate beam search is used in practice

Phrase-Based Machine Translation

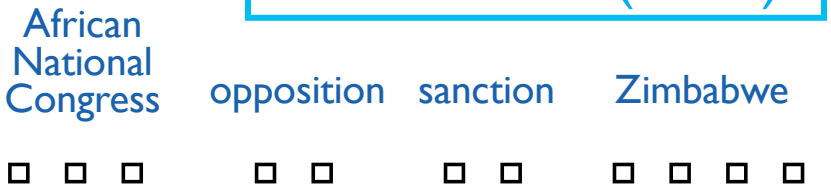
Koehn et al. (2003)



Reference translation:
African National Congress opposes sanctions against Zimbabwe

Phrase-Based Machine Translation

Koehn et al. (2003)



Reference translation:
African National Congress opposes sanctions against Zimbabwe

Phrase Table	
1	□ □ □ / African National Congress
2	□ □ / opposition to
3	□ □ / is opposed to
4	□ □ / sanctions
5	□ □ □ □ □ □ / sanctions against Zimbabwe
...	

Phrase-Based Machine Translation

Koehn et al. (2003)

African National Congress | opposition | sanction | Zimbabwe

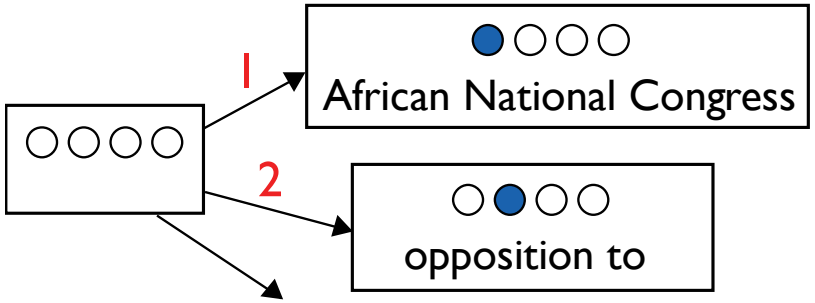
□ □ □ □ □ □ □ □ □ □

Reference translation:
African National Congress opposes sanctions against Zimbabwe

Phrase Table

1	□ □ □	/ African National Congress
2	□ □	/ opposition to
3	□ □	/ is opposed to
4	□ □	/ sanctions
5	□ □ □ □ □ □	/ sanctions against Zimbabwe

...



Phrase-Based Machine Translation

Koehn et al. (2003)

African National Congress | opposition | sanction | Zimbabwe

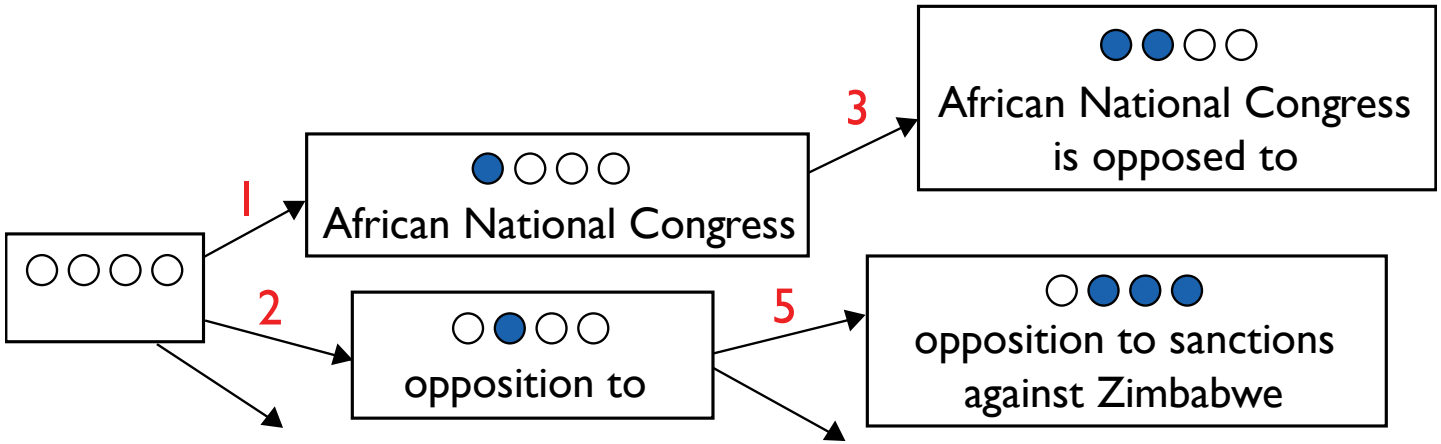
□ □ □ | □ □ | □ □ | □ □ □ □

Reference translation:
African National Congress opposes sanctions against Zimbabwe

Phrase Table

1	□ □ □	/ African National Congress
2	□ □	/ opposition to
3	□ □	/ is opposed to
4	□ □	/ sanctions
5	□ □	□ □ □ □ / sanctions against Zimbabwe

...



Phrase-Based Machine Translation

Koehn et al. (2003)

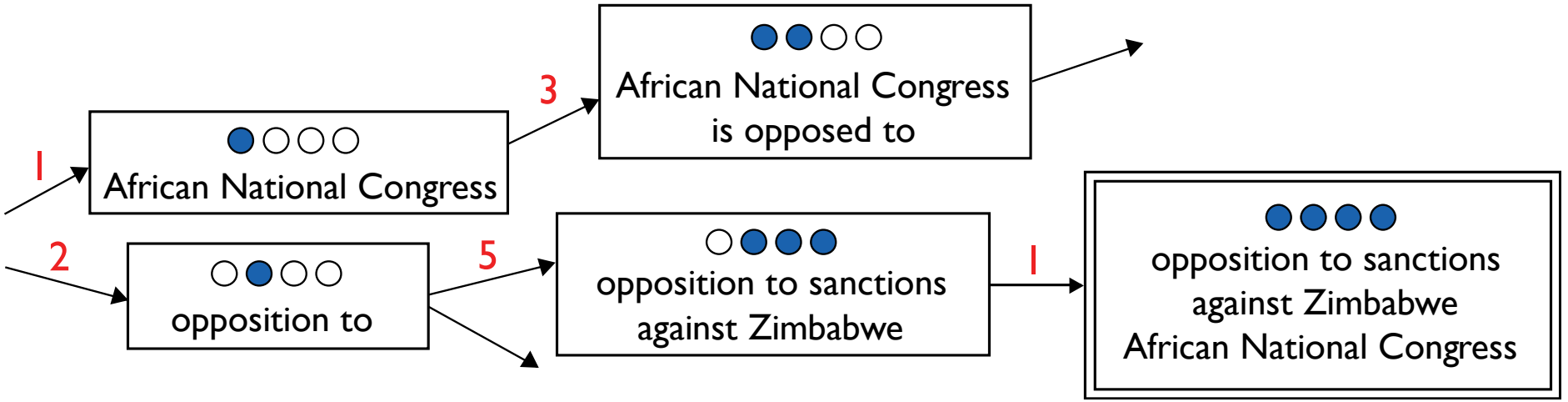
African National Congress | opposition | sanction | Zimbabwe

□ □ □ □ □ □ □ □ □

Reference translation:
African National Congress opposes sanctions against Zimbabwe

Phrase Table

1	□ □ □	/ African National Congress
2	□ □	/ opposition to
3	□ □	/ is opposed to
4	□ □	/ sanctions
5	□ □ □ □ □ □	/ sanctions against Zimbabwe
...		



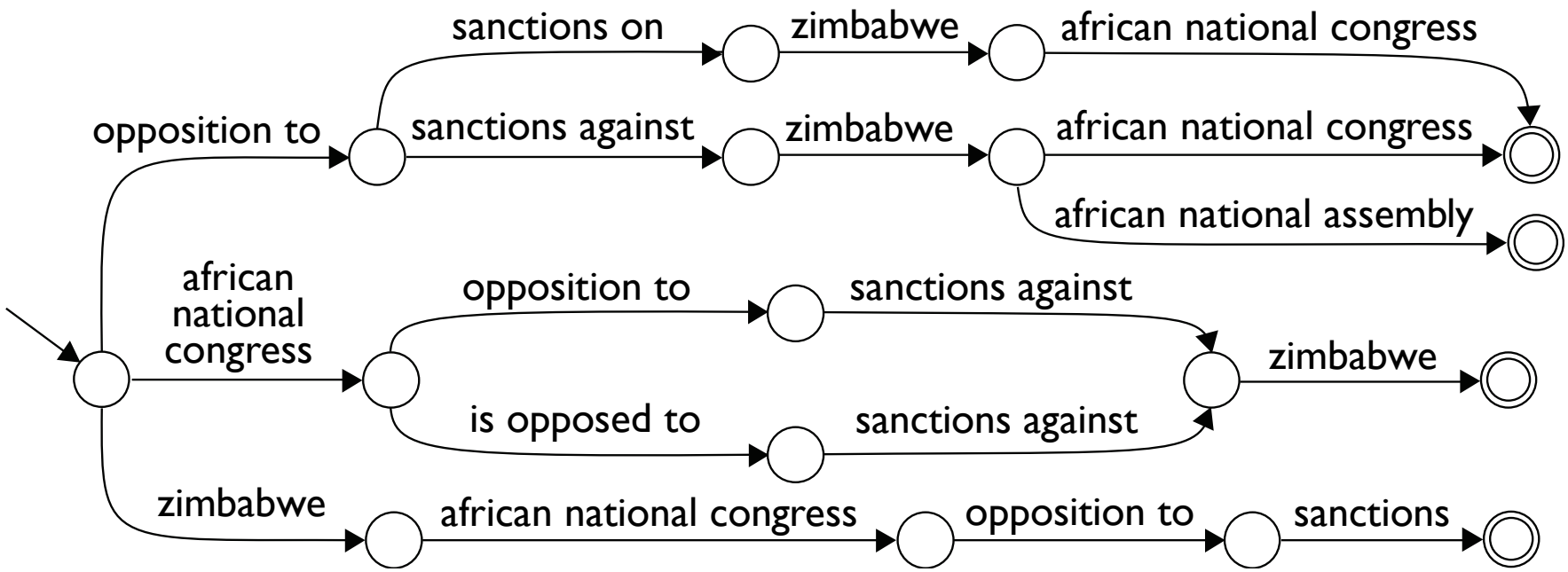
other useful inference tasks:

- find k-best translations

Rank	Score	
1	-11.8	opposition to | sanctions against | zimbabwe | african national congress
2	-12.1	african national congress | opposition to | sanctions against | zimbabwe
3	-12.4	african national congress | oppose | sanctions against | zimbabwe
4	-12.9	zimbabwe | african national congress | opposition to | sanctions
5	-13.5	opposition to | sanctions on | zimbabwe | african national congress

other useful inference tasks:

- find **phrase lattice** of translations



typical lattices contain up to 10^{80} paths!

(but not all are unique translations)

Neural Networks and Machine Translation

- current trend in MT research is to use neural networks for everything
- "neural MT" typically refers to approaches that **only** use neural networks
- but most MT systems combine traditional phrase-based models with features based on neural networks

Fast and Robust Neural Network Joint Models for Statistical Machine Translation

ACL 2014 (best paper award)

Jacob Devlin, Rabih Zbib, Zhongqiang Huang,
Thomas Lamar, Richard Schwartz, and **John Makhoul**
Raytheon BBN Technologies, 10 Moulton St, Cambridge, MA 02138, USA
{jdevlin,rzbib,zhuang,tlamar,schwartz,makhoul}@bbn.com

Abstract

Recent work has shown success in using neural network language models (NNLMs) as features in MT systems. Here, we present a novel formulation for a neural network *joint* model (NNJM), which augments the NNLM with a source context window. Our model is purely lexicalized and can be integrated into any MT decoder. We also present several variations of the NNJM which provide significant additive improvements.

Although the model is quite simple, it yields strong empirical results. On the NIST OpenMT12 Arabic-English condition, the NNJM features produce a gain of +3.0 BLEU on top of a powerful, feature-rich baseline which already includes a target-only NNLM. The NNJM features also produce a gain of +6.3 BLEU on top of a simpler baseline equivalent to Chiang's (2007) original Hiero implementation.

Fast and Robust Neural Network Joint Models for Statistical Machine Translation

ACL 2014

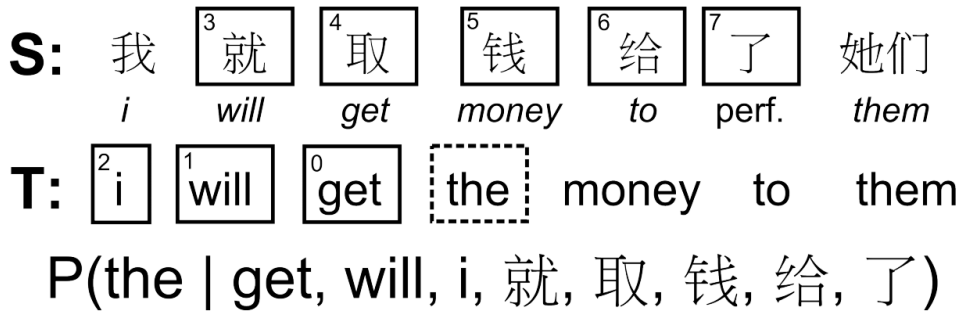


Figure 1: Context vector for target word "the", using a 3-word target history and a 5-word source window (i.e., $n = 4$ and $m = 5$). Here, "the" inherits its affiliation from "money" because this is the first aligned word to its right. The number in each box denotes the index of the word in the context vector. This indexing must be consistent across samples, but the absolute ordering does not affect results.

Fast and Robust Neural Network Joint Models for Statistical Machine Translation

ACL 2014

NIST MT12 Test		
	Ar-En	**Ch-En**
	BLEU	BLEU
OpenMT12 - 1st Place	49.5	32.6
OpenMT12 - 2nd Place	47.5	32.2
OpenMT12 - 3rd Place	47.4	30.8
...	...	...
OpenMT12 - 9th Place	44.0	27.0
OpenMT12 - 10th Place	41.2	25.7
Baseline (w/o RNNLM)	48.9	33.0
Baseline (w/ RNNLM)	49.8	33.4
+ S2T/L2R NNJM (Dec)	51.2	34.2
+ S2T NNLTM (Dec)	52.0	34.2
+ T2S NNLTM (Resc)	51.9	34.2
+ S2T/R2L NNJM (Resc)	52.2	34.3
+ T2S/L2R NNJM (Resc)	52.3	34.5
+ T2S/R2L NNJM (Resc)	52.8	34.7

Neural MT

Recurrent Continuous Translation Models

EMNLP 2013

Nal Kalchbrenner **Phil Blunsom**
Department of Computer Science
University of Oxford

Abstract

We introduce a class of probabilistic continuous translation models called Recurrent Continuous Translation Models that are purely based on continuous representations for words, phrases and sentences and do not rely on alignments or phrasal translation units. The models have a generation and a conditioning aspect. The generation of the translation is modelled with a target Recurrent Language Model, whereas the conditioning on the source sentence is modelled with a Convolutional Sentence Model. Through various experiments, we show first that our models obtain a perplexity with respect to gold translations that is $> 43\%$ lower than that of state-of-the-art alignment-based translation models.

Recurrent Continuous Translation Models

EMNLP 2013

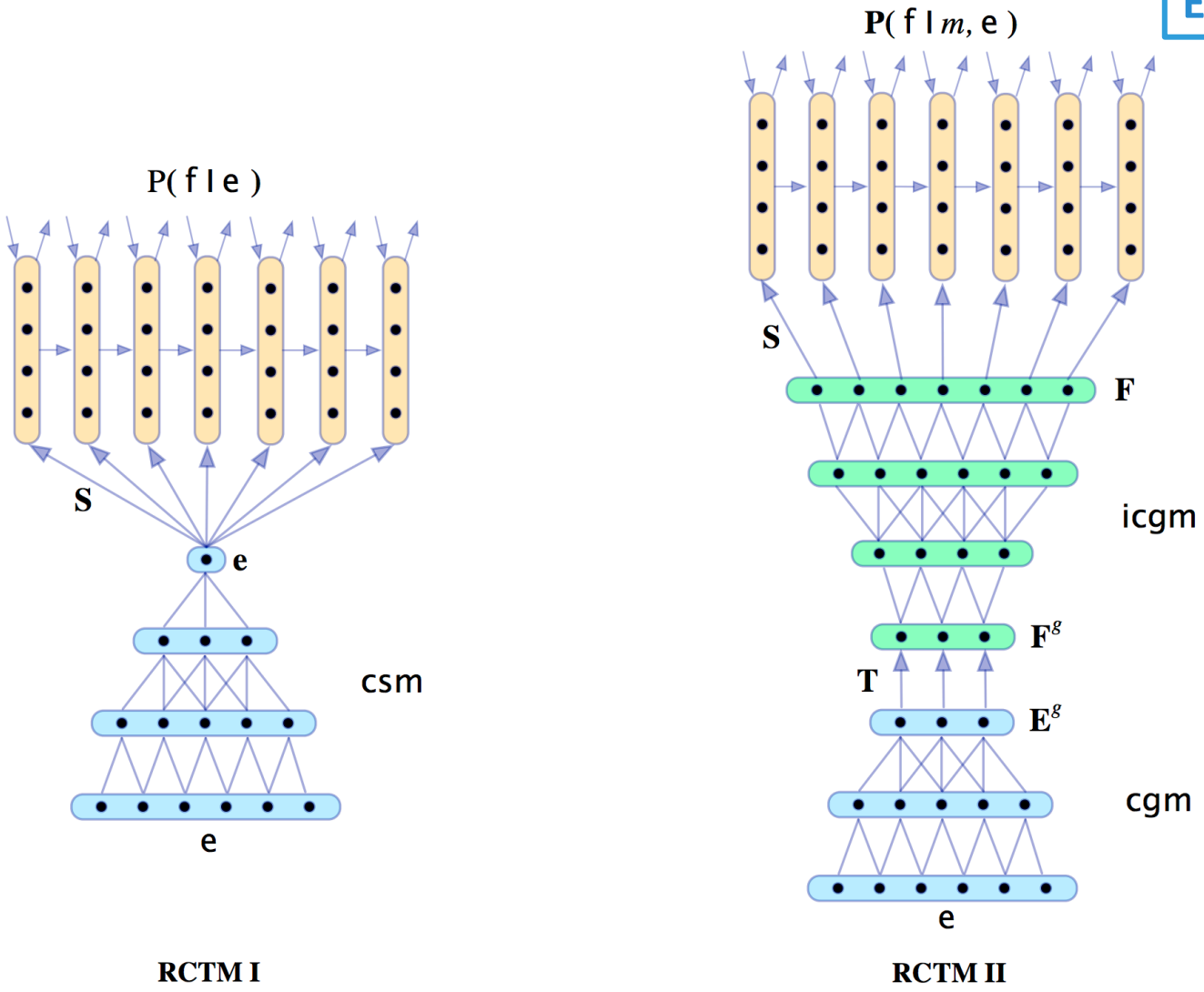


Figure 3: A graphical depiction of the two RCTMs. Arrows represent full matrix transformations while lines are vector transformations corresponding to columns of weight matrices.

Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation

EMNLP 2014

Kyunghyun Cho
Bart van Merriënboer **Caglar Gulcehre**
Université de Montréal
firstname.lastname@umontreal.ca

Dzmitry Bahdanau
Jacobs University, Germany
d.bahdanau@jacobs-university.de

Fethi Bougares **Holger Schwenk**
Université du Maine, France
firstname.lastname@lium.univ-lemans.fr

Yoshua Bengio
Université de Montréal, CIFAR Senior Fellow
find.me@on.the.web

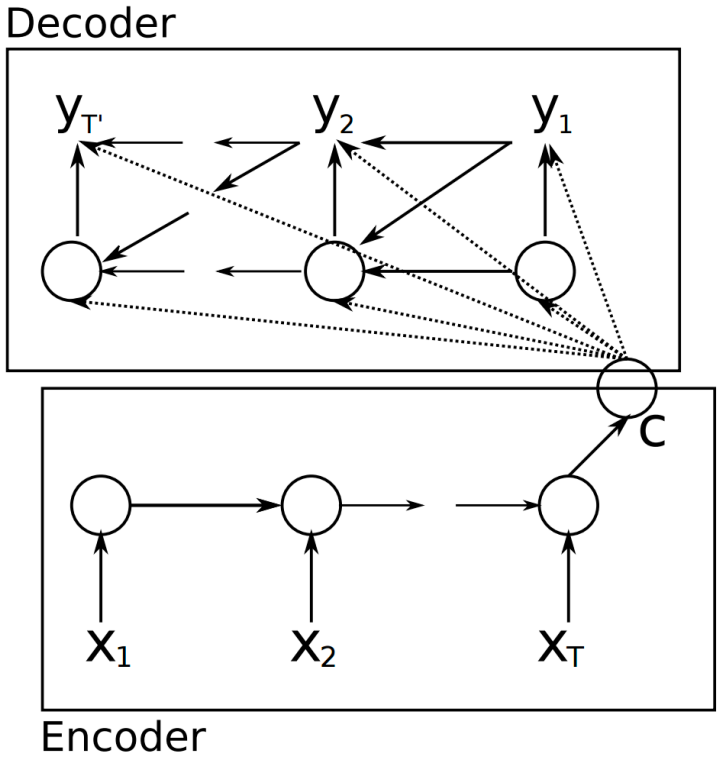
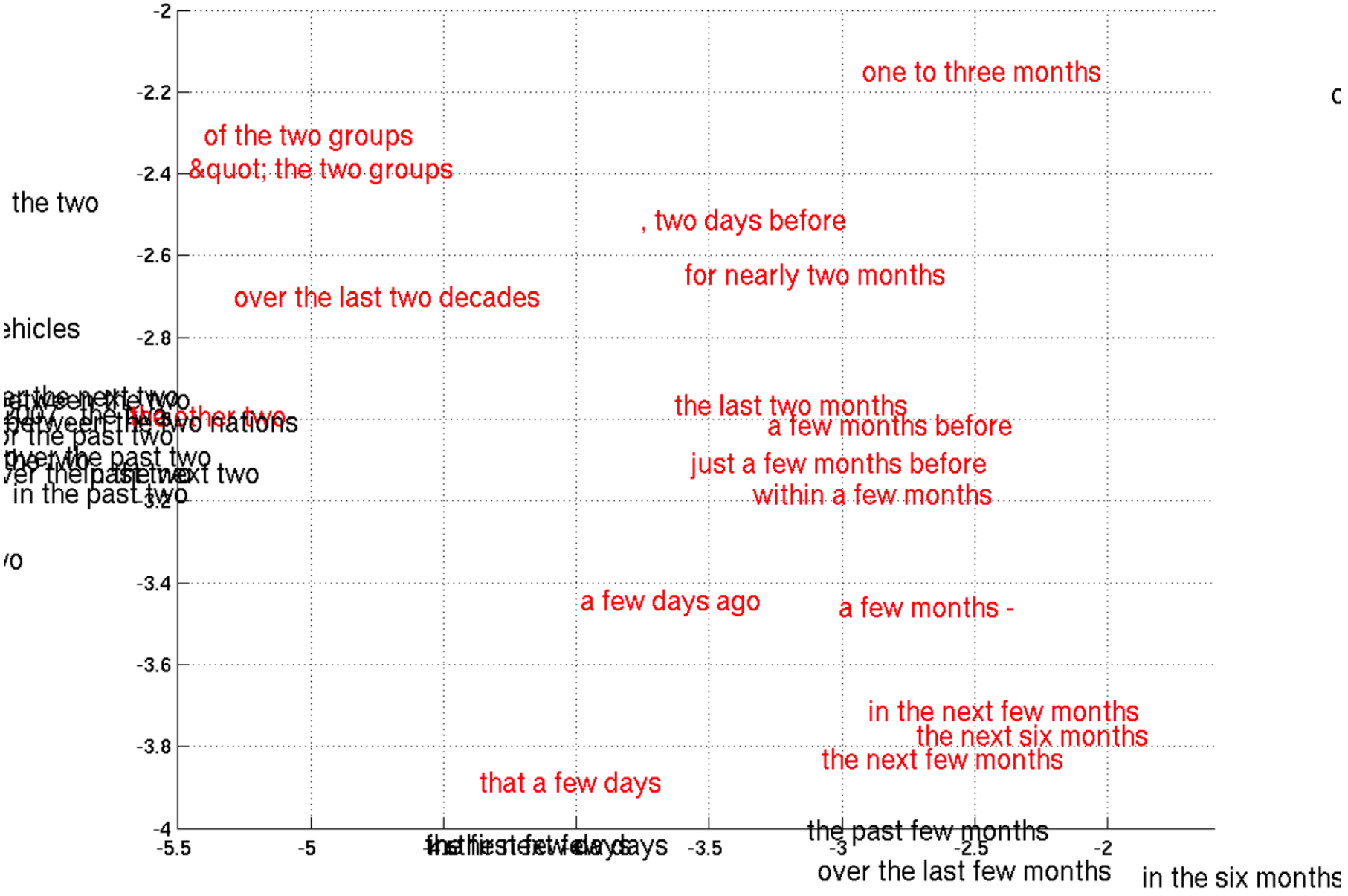


Figure 1: An illustration of the proposed RNN Encoder–Decoder.

Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation

EMNLP 2014



Sequence to Sequence Learning with Neural Networks

NIPS 2014

Ilya Sutskever
Google
ilyasu@google.com

Oriol Vinyals
Google
vinyals@google.com

Quoc V. Le
Google
qvl@google.com

Abstract

Deep Neural Networks (DNNs) are powerful models that have achieved excellent performance on difficult learning tasks. Although DNNs work well whenever large labeled training sets are available, they cannot be used to map sequences to sequences. In this paper, we present a general end-to-end approach to sequence learning that makes minimal assumptions on the sequence structure. Our method uses a multilayered Long Short-Term Memory (LSTM) to map the input sequence to a vector of a fixed dimensionality, and then another deep LSTM to decode the target sequence from the vector. Our main result is that on an English to French translation task from the WMT-14 dataset, the translations produced by the LSTM achieve a BLEU score of 34.8 on the entire test set, where the LSTM's BLEU score was penalized on out-of-vocabulary words. Additionally, the LSTM did not have difficulty on long sentences. For comparison, a phrase-based SMT system achieves a BLEU score of 33.3 on the same dataset. When we used the LSTM to rerank the 1000 hypotheses produced by the aforementioned SMT system, its BLEU score increases to 36.5, which is close to the previous state of the art. The LSTM also learned sensible phrase and sentence representations that are sensitive to word order and are relatively invariant to the active and the passive voice. Finally, we found that reversing the order of the words in all source sentences (but not target sentences) improved the LSTM's performance markedly, because doing so introduced many short term dependencies between the source and the target sentence which made the optimization problem easier.

Sequence to Sequence Learning with Neural Networks

NIPS 2014

Ilya Sutskever
Google
ilyasu@google.com

Oriol Vinyals
Google
vinyals@google.com

Quoc V. Le
Google
qvl@google.com

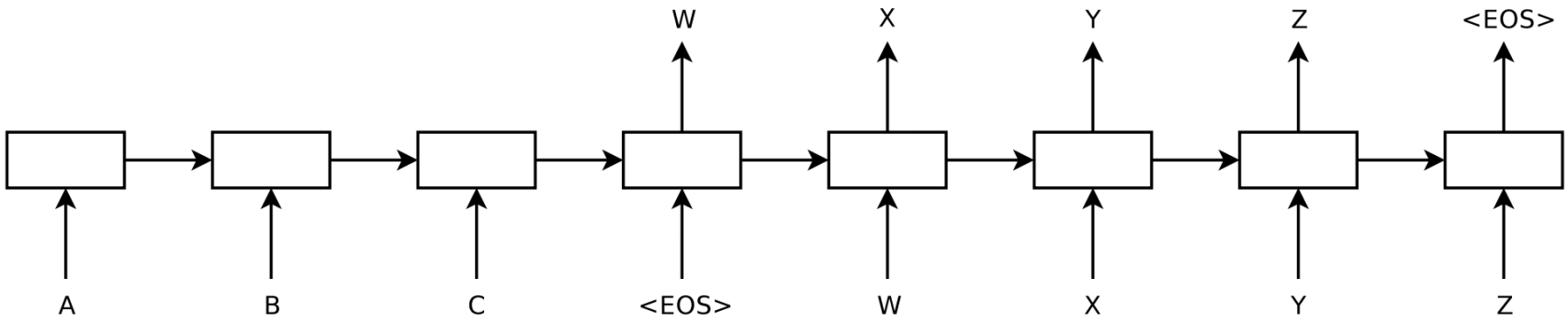


Figure 1: Our model reads an input sentence "ABC" and produces "WXYZ" as the output sentence. The model stops making predictions after outputting the end-of-sentence token. Note that the LSTM reads the input sentence in reverse, because doing so introduces many short term dependencies in the data that make the optimization problem much easier.

Sequence to Sequence Learning with Neural Networks

NIPS 2014

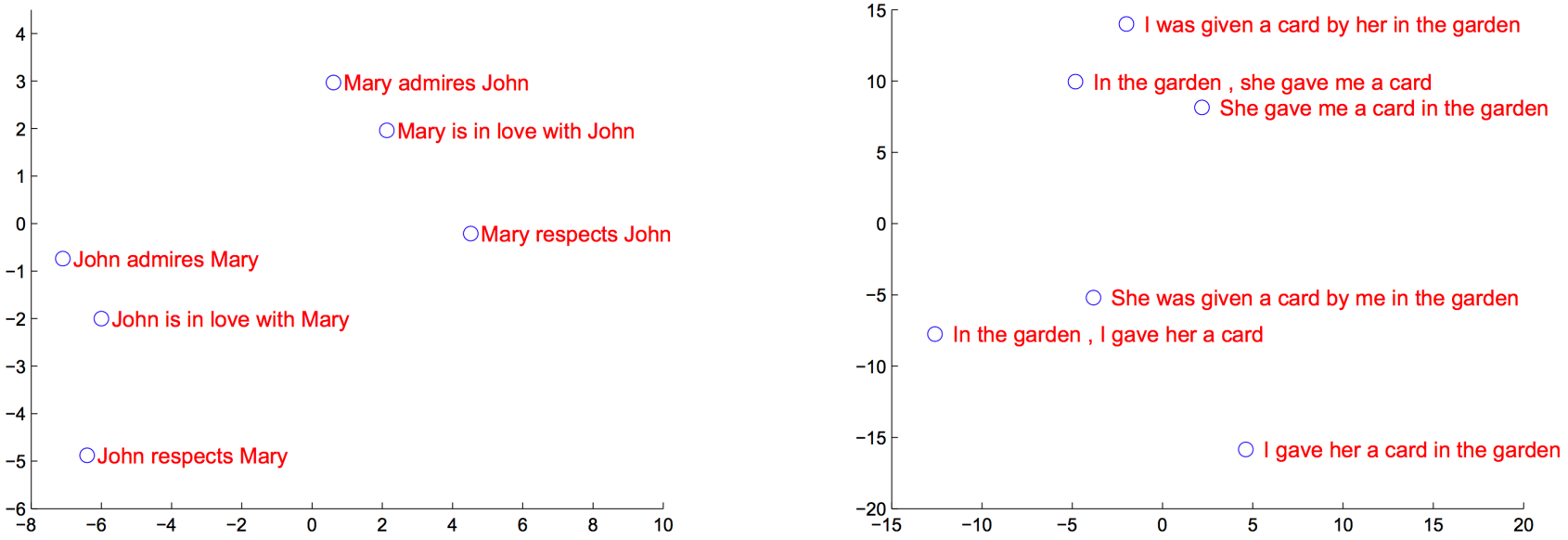


Figure 2: The figure shows a 2-dimensional PCA projection of the LSTM hidden states that are obtained after processing the phrases in the figures. The phrases are clustered by meaning, which in these examples is primarily a function of word order, which would be difficult to capture with a bag-of-words model. Notice that both clusters have similar internal structure.

NEURAL MACHINE TRANSLATION BY JOINTLY LEARNING TO ALIGN AND TRANSLATE

Dzmitry Bahdanau
Jacobs University Bremen, Germany

KyungHyun Cho **Yoshua Bengio***
Université de Montréal

ICLR 2015

ABSTRACT

Neural machine translation is a recently proposed approach to machine translation. Unlike the traditional statistical machine translation, the neural machine translation aims at building a single neural network that can be jointly tuned to maximize the translation performance. The models proposed recently for neural machine translation often belong to a family of encoder–decoders and encode a source sentence into a fixed-length vector from which a decoder generates a translation. In this paper, we conjecture that the use of a fixed-length vector is a bottleneck in improving the performance of this basic encoder–decoder architecture, and propose to extend this by allowing a model to automatically (soft-)search for parts of a source sentence that are relevant to predicting a target word, without having to form these parts as a hard segment explicitly. With this new approach, we achieve a translation performance comparable to the existing state-of-the-art phrase-based system on the task of English-to-French translation. Furthermore, qualitative analysis reveals that the (soft-)alignments found by the model agree well with our intuition.

NEURAL MACHINE TRANSLATION BY JOINTLY LEARNING TO ALIGN AND TRANSLATE

Dzmitry Bahdanau
Jacobs University Bremen, Germany

KyungHyun Cho **Yoshua Bengio***
Université de Montréal

ICLR 2015

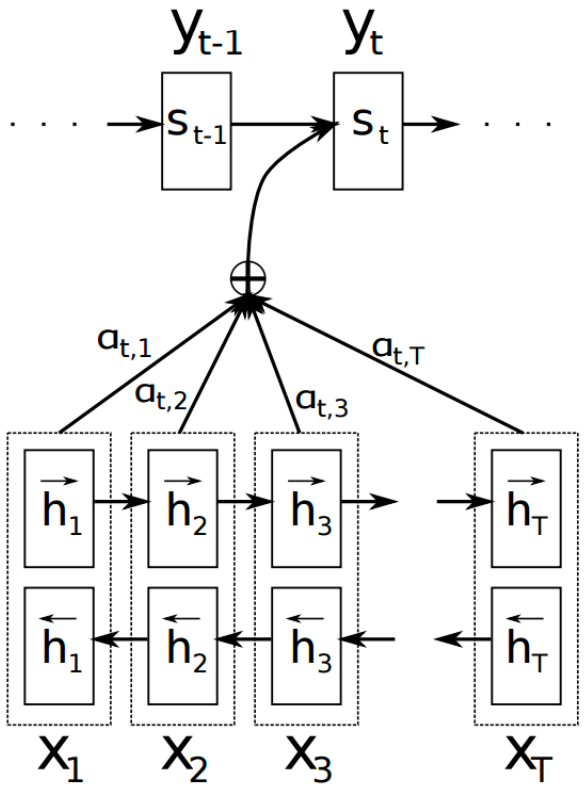
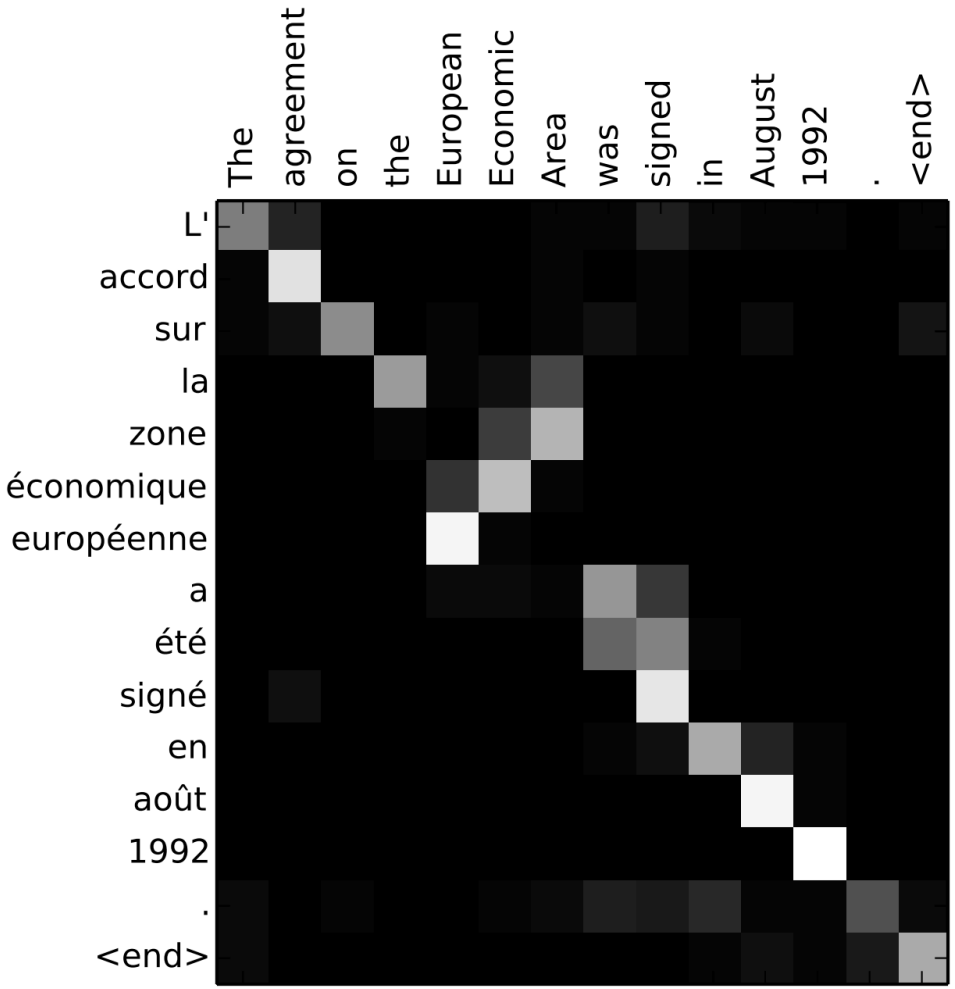


Figure 1: The graphical illustration of the proposed model trying to generate the t-th target word y_t given a source sentence $(x_1, x_2, \ldots, x_T)$.

Neural Machine Translation by Jointly Learning to Align and Translate

ICLR 2015

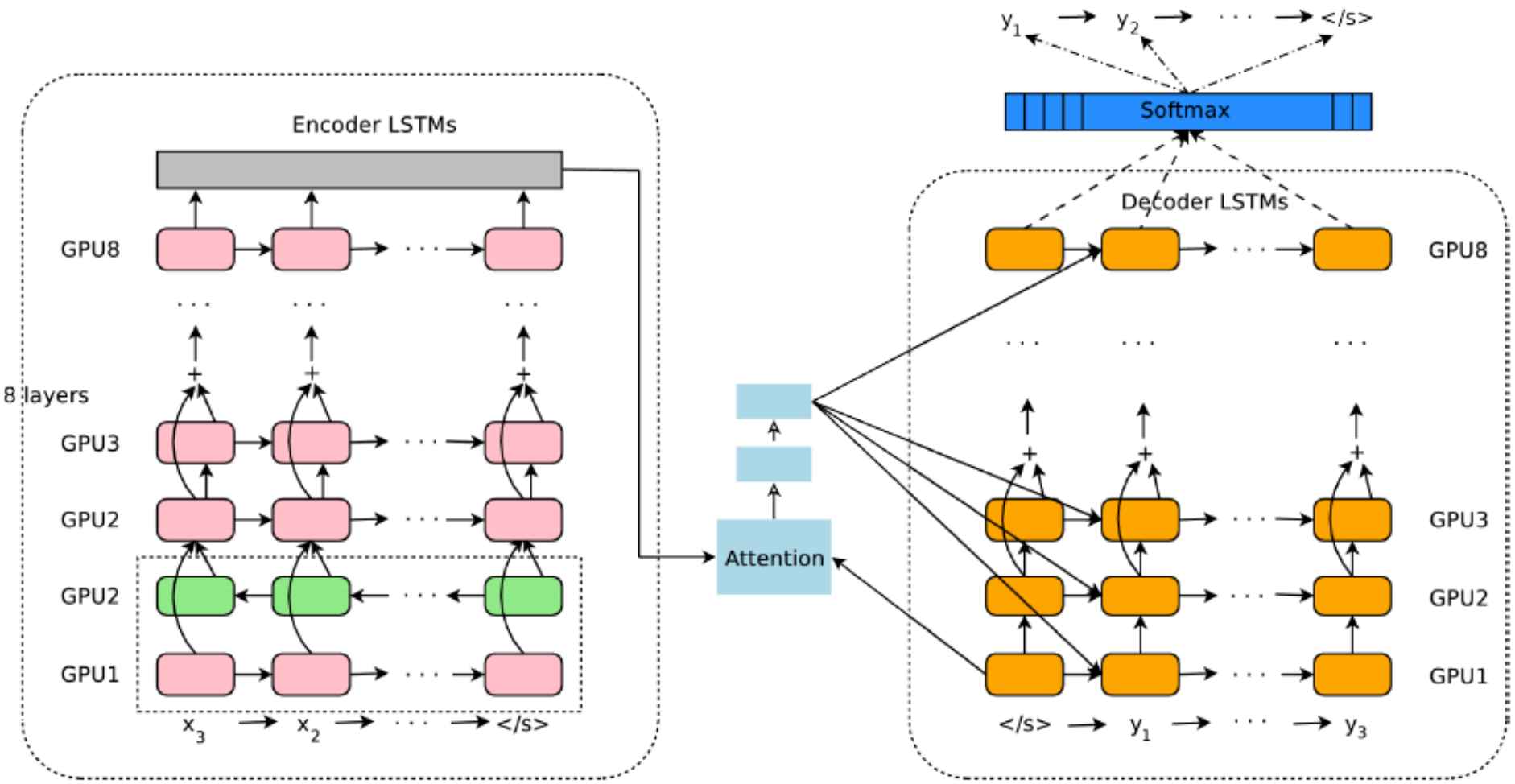


Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi
yonghui,schuster,zhifengc,qvl,mnorouzi@google.com

Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, Jeffrey Dean

Google's NMT System



Google's NMT System

In our experience with large-scale translation tasks, simple stacked LSTM layers work well up to 4 layers, barely with 6 layers, and very poorly beyond 8 layers.

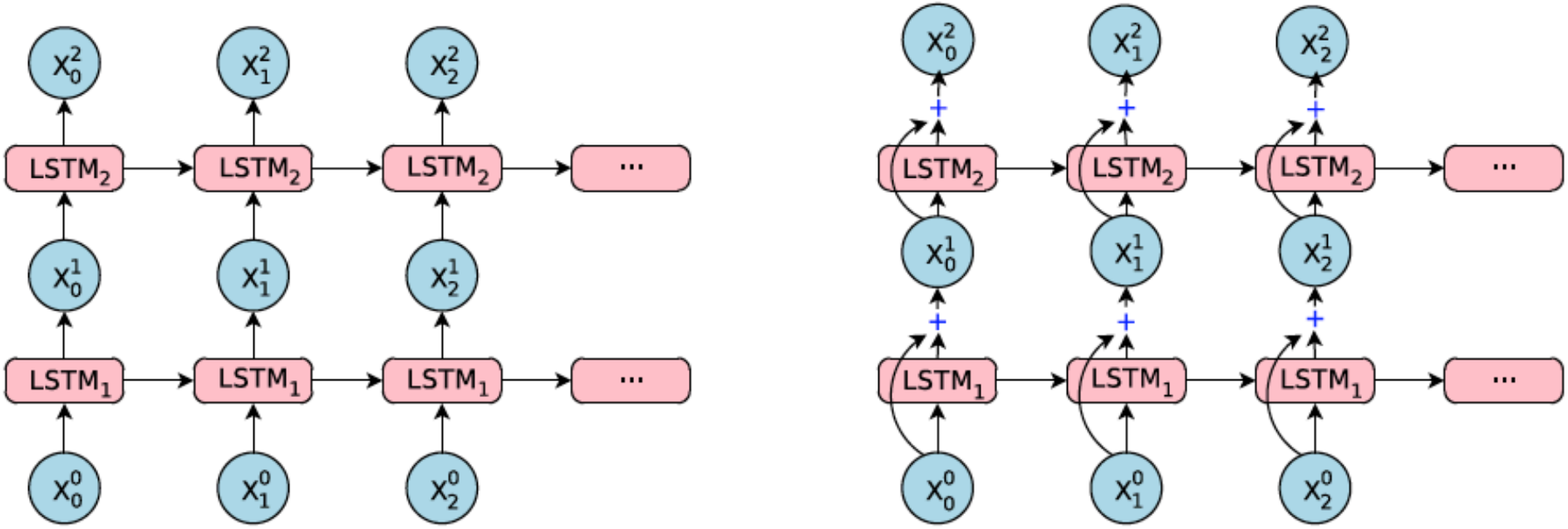


Figure 2: The difference between normal stacked LSTM and our stacked LSTM with residual connections. On the left: simple stacked LSTM layers [41]. On the right: our implementation of stacked LSTM layers with residual connections. With residual connections, input to the bottom LSTM layer ($\mathbf{x_i^0}$'s to LSTM$_1$) is element-wise added to the output from the bottom layer ($\mathbf{x_i^1}$'s). This sum is then fed to the top LSTM layer (LSTM$_2$) as the new input.

Google's NMT System

Here is an example of a word sequence and the corresponding wordpiece sequence:

- **Word**: Jet makers feud over seat width with big orders at stake
- **wordpieces**: _J et _makers _fe ud _over _seat _width _with _big _orders _at _stake

In the above example, the word "Jet" is broken into two wordpieces "_J" and "et", and the word "feud" is broken into two wordpieces "_fe" and "ud". The other words remain as single wordpieces. "_" is a special character added to mark the beginning of a word.

they use a procedure that deterministically segments any character sequence into wordpieces

vocab: 8k-32k wordpieces

they first learn a "wordpiece model"

Google's NMT System

Table 5: Single model results on WMT En→De (newstest2014)

Model	BLEU	CPU decoding time per sentence (s)
Word	23.12	0.2972
Character (512 nodes)	22.62	0.8011
WPM-8K	23.50	0.2079
WPM-16K	24.36	0.1931
WPM-32K	24.61	0.1882
Mixed Word/Character	24.17	0.3268
PBMT [6]	20.7	
RNNSearch [37]	16.5	
RNNSearch-LV [37]	16.9	
RNNSearch-LV [37]	16.9	
Deep-Att [45]	20.6	