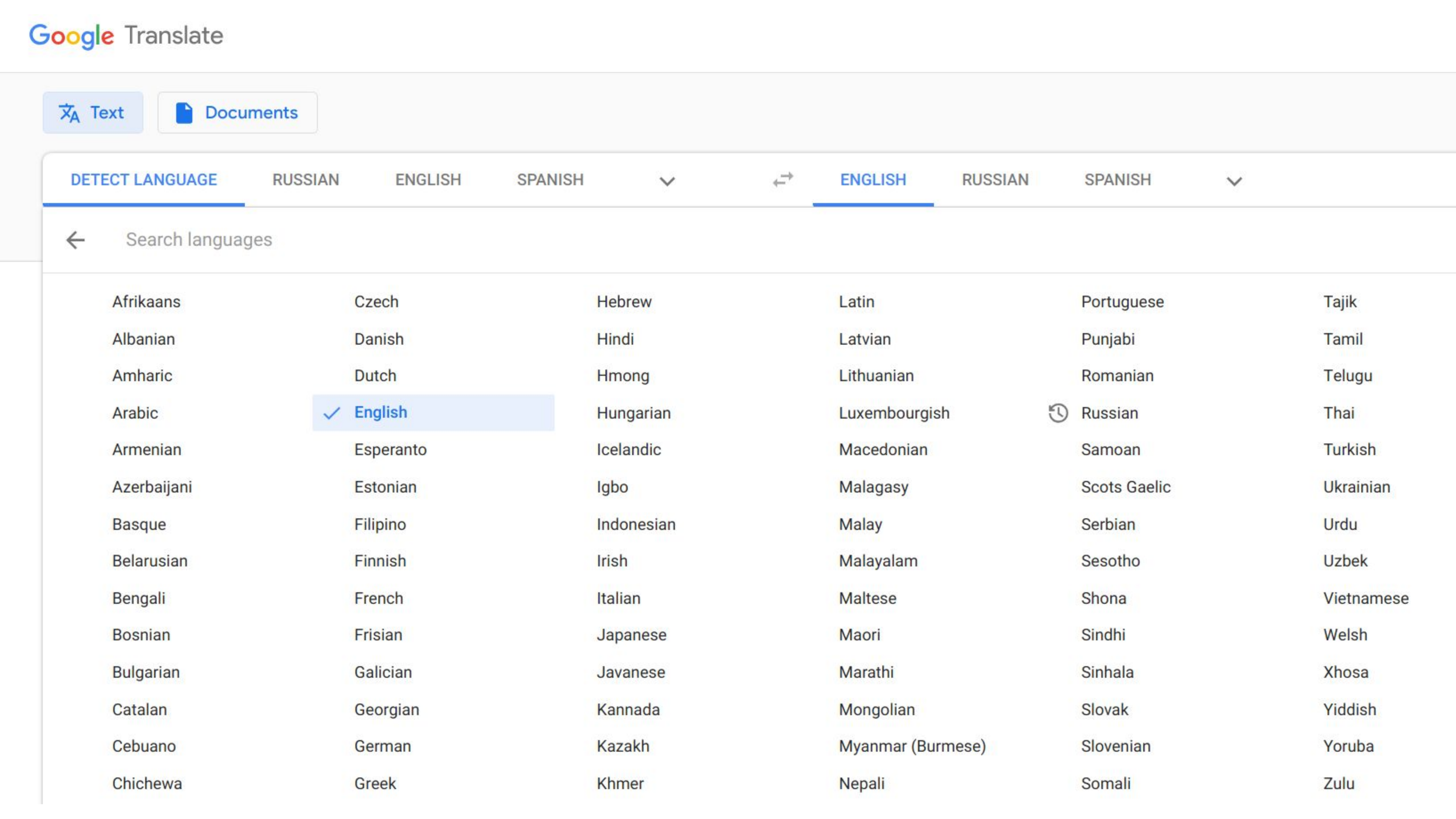
Algorithms for NLP

Neural Machine Translation

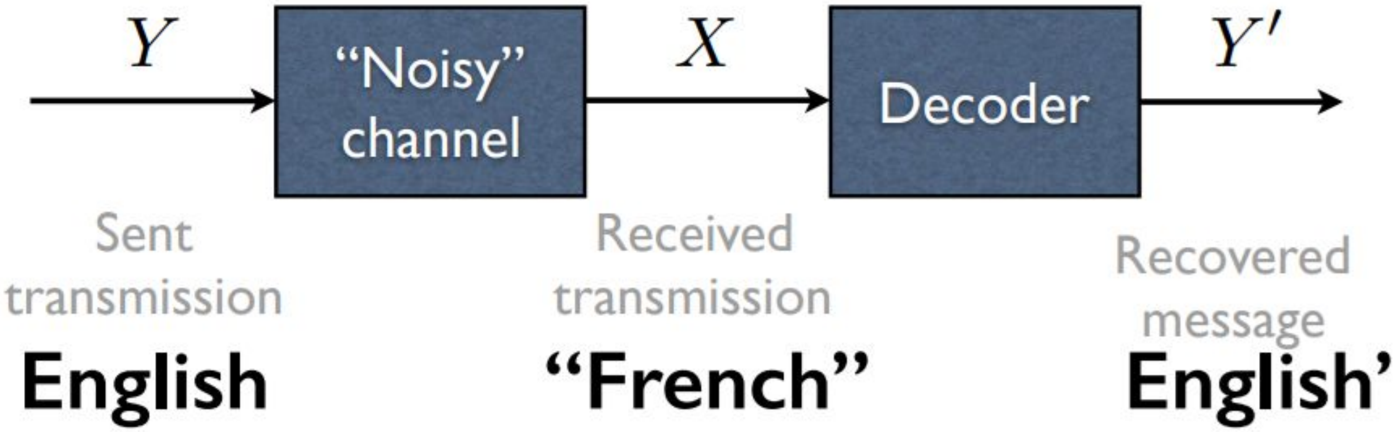
Yulia Tsvetkov – CMU

Slides: Chris Dyer – DeepMind


مرحبا
selamat
bonjour
hello
bon
dobry
dobar
goede
good
dobrý
Добрий
zdravo
bara
tere
dan
dia
hallo
chào
namaskaram
moien
morgen
gozaimasu
namaste
kazô
vin
arè
den
ani
dag
heil
večer
bonjou
pagi
hola
menyéga
warro
labas
boa
diarama
mat
azul
päivää
dober
chlomo
день
ora
hei
ina
kia


Google Translate
Text
Documents
DETECT LANGUAGE
RUSSIAN
ENGLISH
SPANISH
ENGLISH
RUSSIAN
SPANISH
Search languages
Afrikaans
Albanian
Amharic
Arabic
Armenian
Azerbaijani
Basque
Belarusian
Bengali
Bosnian
Bulgarian
Catalan
Cebuano
Chichewa
Czech
Danish
Dutch
English
Esperanto
Estonian
Filipino
Finnish
French
Frisian
Galician
Georgian
German
Greek
Hebrew
Hindi
Hmong
Hungarian
Icelandic
Igbo
Indonesian
Irish
Italian
Japanese
Javanese
Kannada
Kazakh
Khmer
Latin
Latvian
Lithuanian
Luxembourgish
Macedonian
Malagasy
Malay
Malayalam
Maltese
Maori
Marathi
Mongolian
Myanmar (Burmese)
Nepali
Portuguese
Punjabi
Romanian
Russian
Samoan
Scots Gaelic
Serbian
Sesotho
Shona
Sindhi
Sinhala
Slovak
Slovenian
Somali
Tajik
Tamil
Telugu
Thai
Turkish
Ukrainian
Urdu
Uzbek
Vietnamese
Welsh
Xhosa
Yiddish
Yoruba
Zulu

Noisy Channel Model

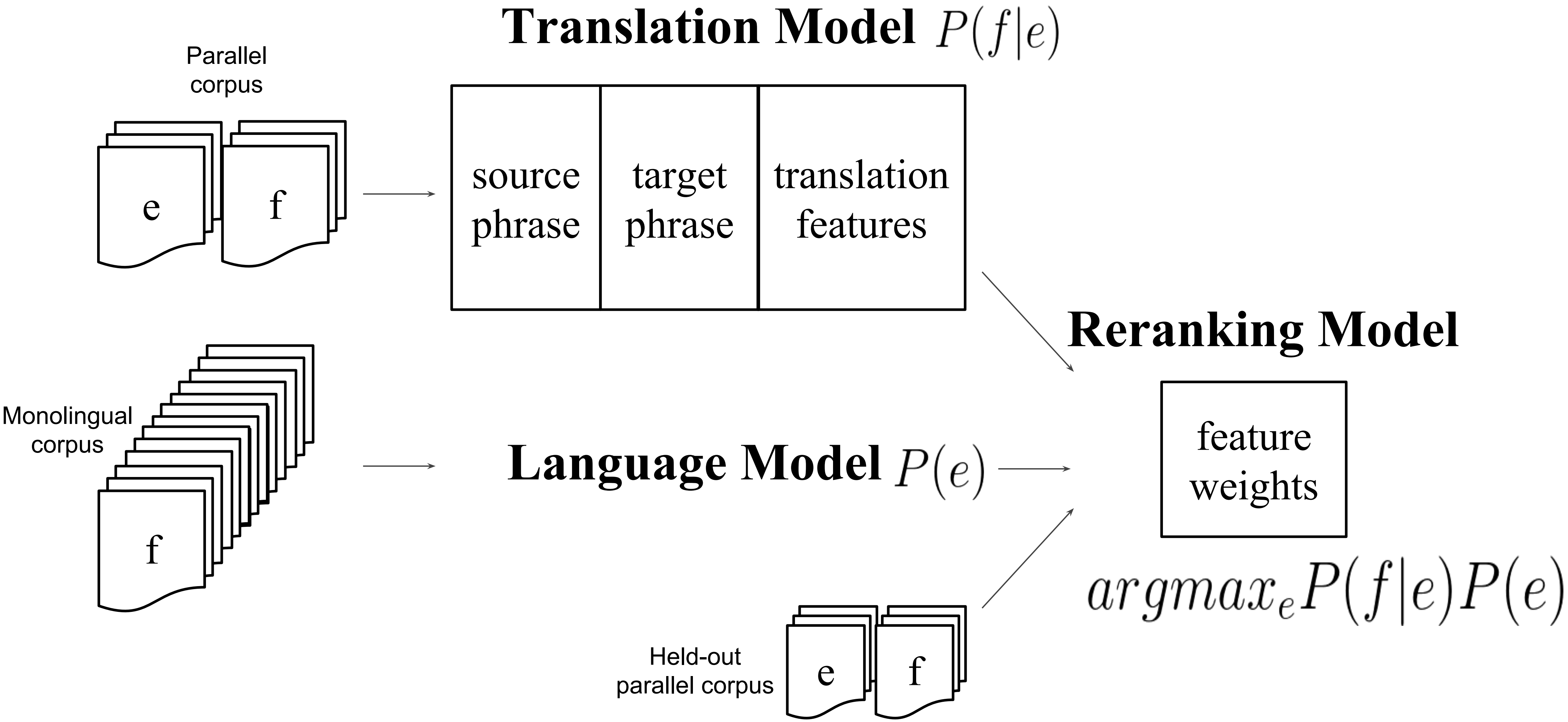


$$\hat{\mathbf{e}} = \arg\max_{\boldsymbol{e}} p_{\boldsymbol{\varphi}}(\boldsymbol{e}) \times p_{\boldsymbol{\theta}}(\boldsymbol{f} \mid \boldsymbol{e})$$

language model

translation model

Phrase-Based MT



Two Views of MT

- Code breaking (aka the noisy channel, Bayes rule)
  - I know the target language
  - I have example translations texts (example enciphered data)

$$\hat{e} = \arg\max_{e} p_{\varphi}(e) \times p_{\theta}(f \mid e)$$

- Direct modeling (aka pattern matching)
  - I have really good learning algorithms and a bunch of example inputs (source language sentences) and outputs (target language translations)

$$\hat{e} = \arg\max_{e} p_{\lambda}(e \mid f)$$

Two Views of MT

- Code breaking (aka the noisy channel, Bayes rule)
  - I know the target language
  - I have example translations texts (example enciphered data)

➡ Statistical Machine Translation (SMT)

- Direct modeling (aka pattern matching)
  - I have really good learning algorithms and a bunch of example inputs (source language sentences) and outputs (target language translations)

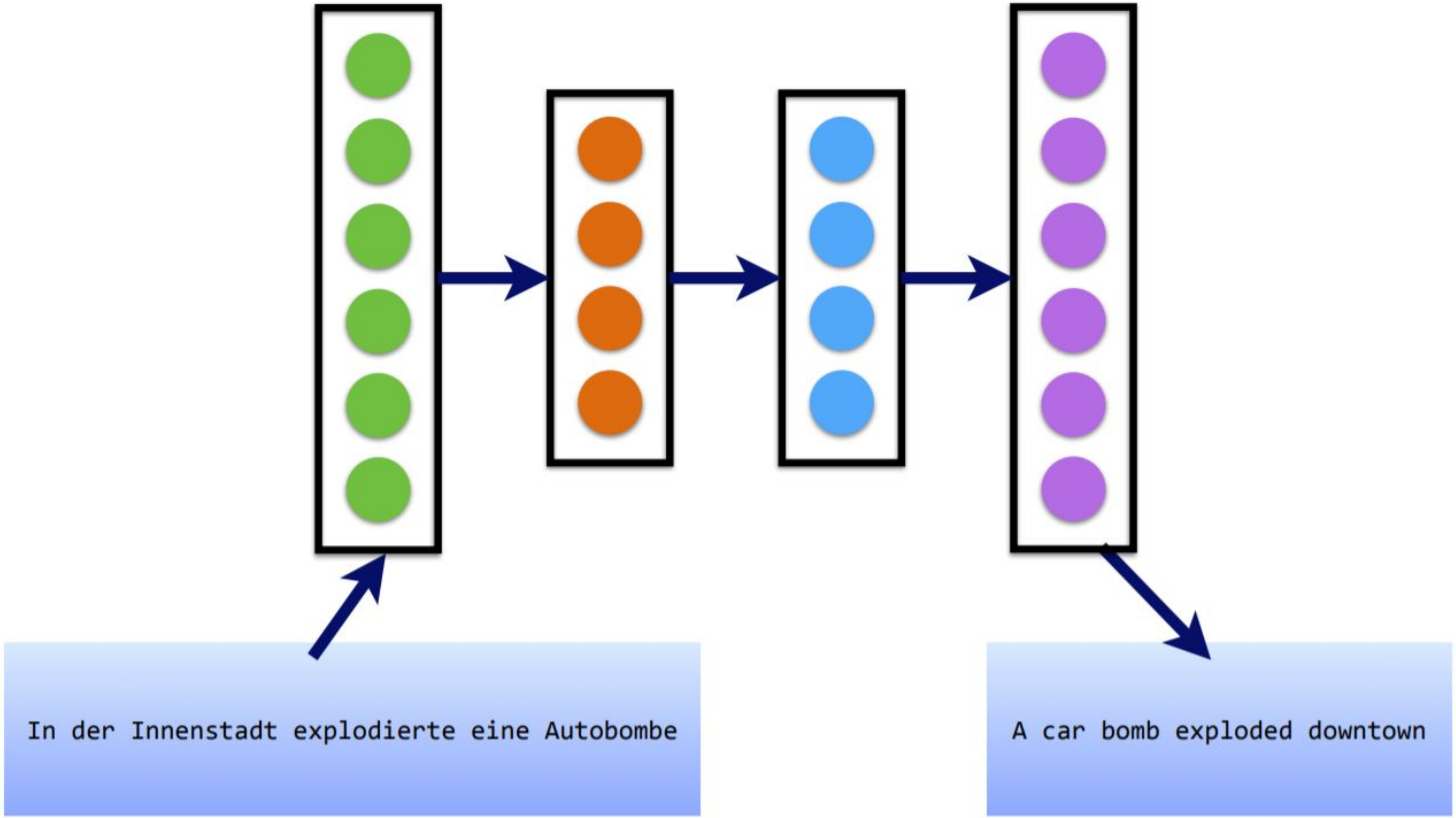
➡ Neural Machine Translation (NMT)

MT as Direct Modeling

$$\hat{\boldsymbol{e}} = \arg\max_{\boldsymbol{e}} p_{\boldsymbol{\lambda}}(\underset{\text{target}}{\boldsymbol{e}} \mid \underset{\text{source}}{\boldsymbol{f}})$$

- one model does everything
- trained to reproduce a corpus of translations

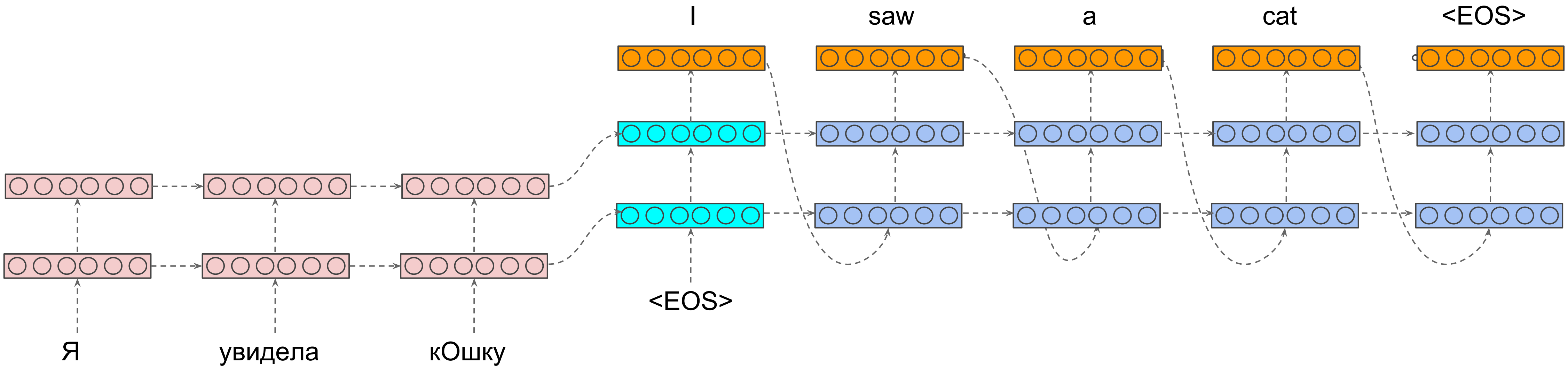
NMT



In der Innenstadt explodierte eine Autobombe
A car bomb exploded downtown

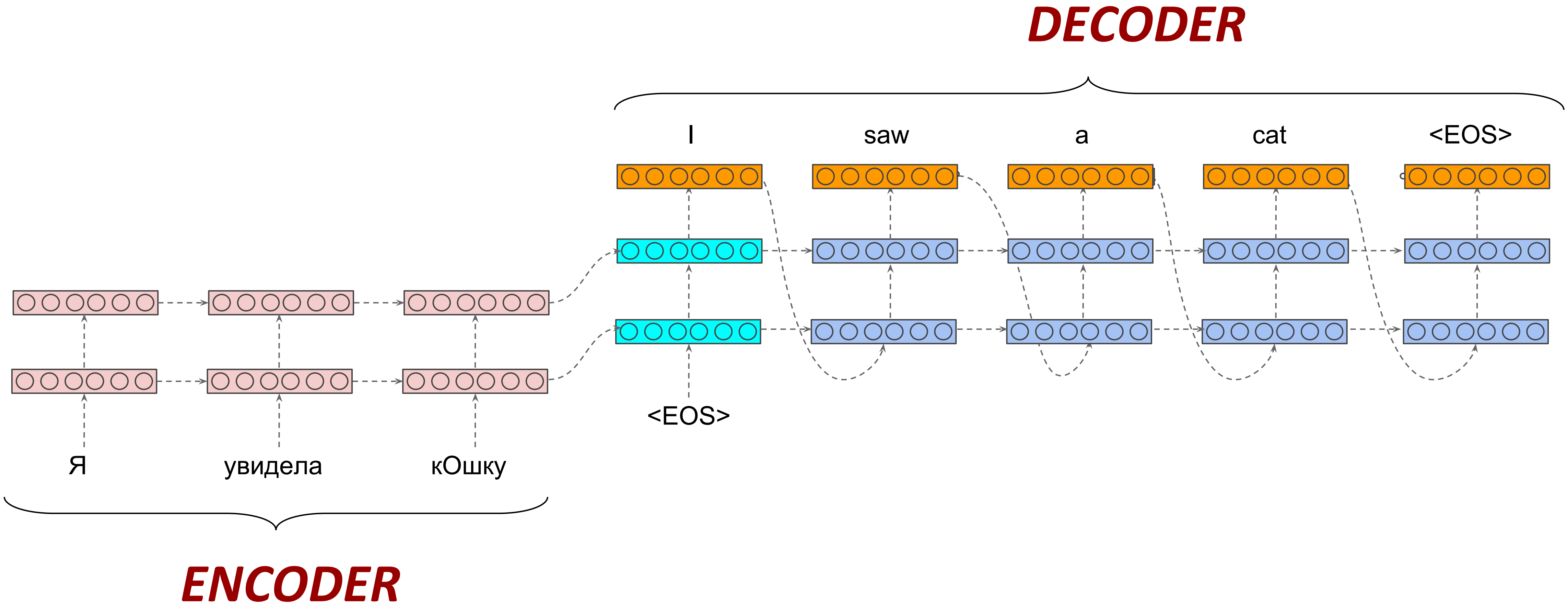
Sequence-to-Sequence Models for NMT

Ilya Sutskever, Oriol Vinyals, Quoc V. Le. 2014. Sequence to Sequence Learning with Neural Networks. *Proc. NIPS*



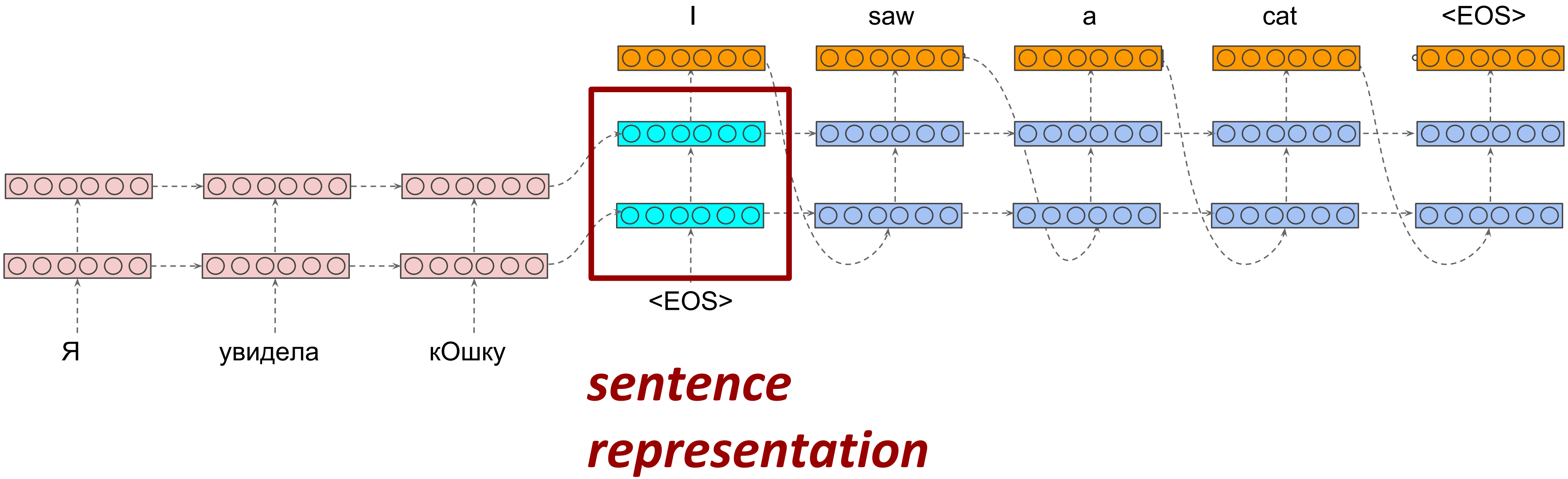
Sequence-to-Sequence Models for NMT

Ilya Sutskever, Oriol Vinyals, Quoc V. Le. 2014. Sequence to Sequence Learning with Neural Networks. *Proc. NIPS*
Kyunghyun Cho, Bart van Merrienboer, Dzmitry Bahdanau, Yoshua Bengio. 2014. On the Properties of Neural Machine Translation: Encoder-Decoder Approaches. *Proc. SSST*



Sequence-to-Sequence Models for NMT

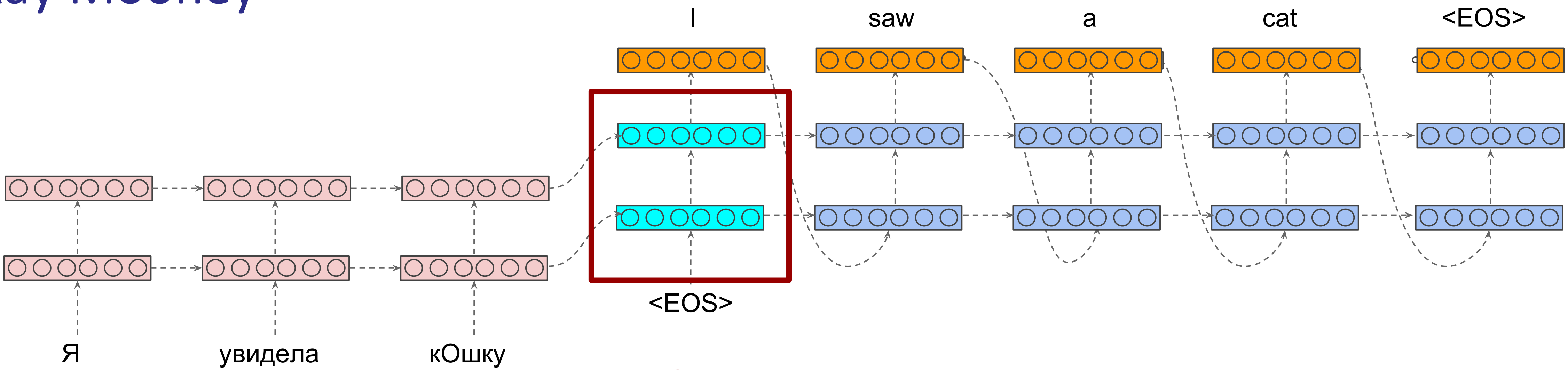
Ilya Sutskever, Oriol Vinyals, Quoc V. Le. 2014. Sequence to Sequence Learning with Neural Networks. *Proc. NIPS*



Problem With Vector Sentence Encoding

"You can't cram the meaning of a whole %&!$ing sentence into a single $&!*ing vector!"

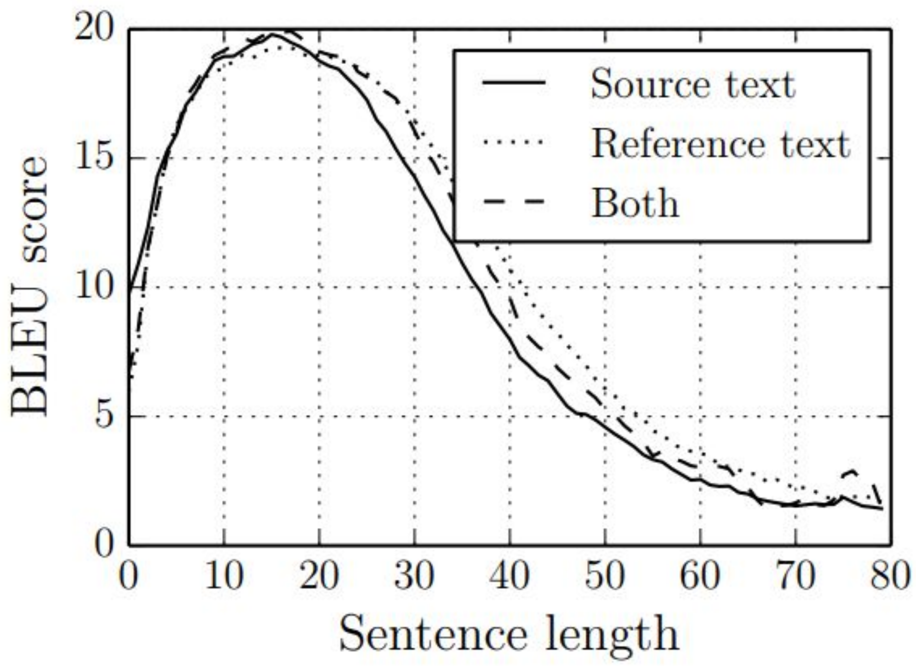
— Ray Mooney



sentence representation

Problem With Vector Sentence Encoding

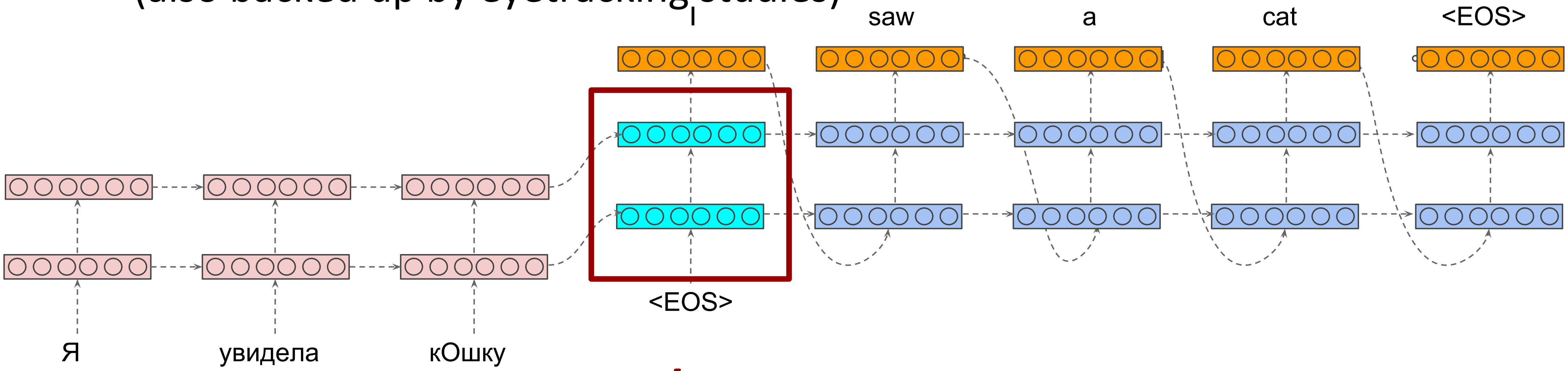
- Fixed sized representation degrades as sentence length increases
  - Reversing source brings some improvement (Sutskever et al., 2014)



Kyunghyun Cho, Bart van Merrienboer, Dzmitry Bahdanau, Yoshua Bengio. 2014. On the Properties of Neural Machine Translation: Encoder-Decoder Approaches. *Proc. SSST*

Problem With Vector Sentence Encoding

- Cho's question: does a translator read and memorize the input sentence/document and then generate the output?
  - Compressing the entire input sentence into a vector basically says "memorize the sentence"
  - Common sense experience says translators refer back and forth to the input. (also backed up by eyetracking studies)

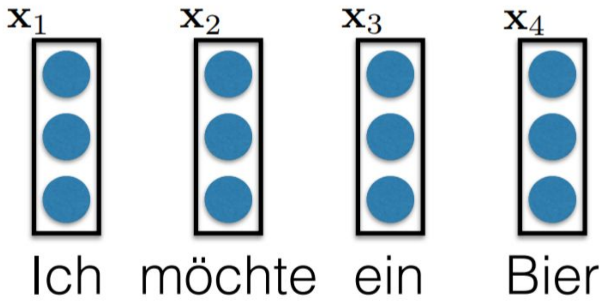


sentence representation

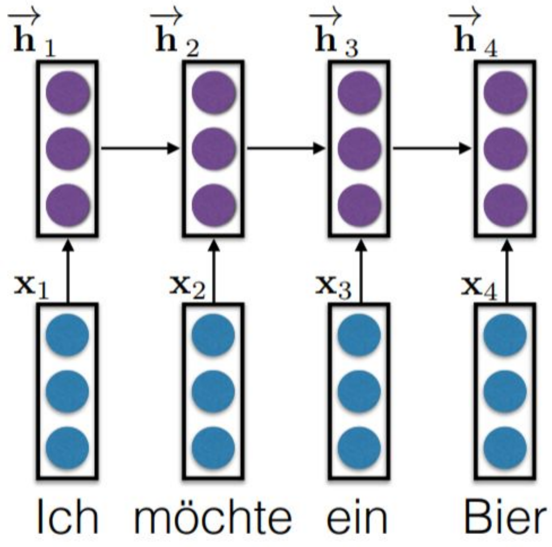
Sequence-to-Sequence Models for NMT

- By far the most widely used architecture is Bidirectional RNN with Attention due to Bahdanau et al (2015)
  - Dzmitry Bahdanau, Kyunghyun Cho, Yoshua Bengio. 2015. Neural Machine Translation by Jointly Learning to Align and Translate. Proc. ICLR
- One column per word
- Each column (word) has two halves concatenated together:
  - a “forward representation”, i.e., a word and its left context
  - a “reverse representation”, i.e., a word and its right context
- Implementation: bidirectional RNNs (GRUs or LSTMs) to read *f* from left to right and right to left, concatenate representations

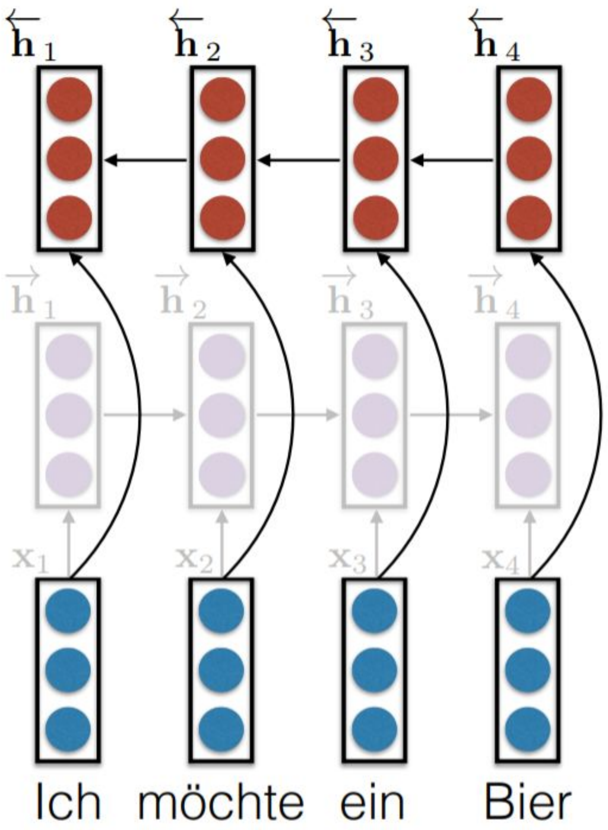
Encoder: Bidirectional RNN



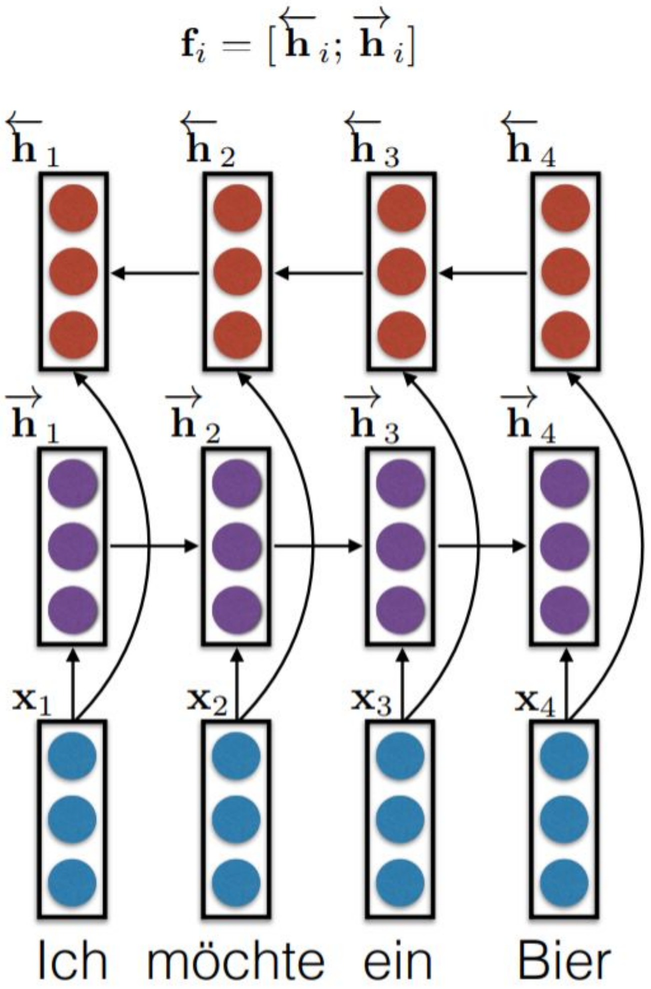
Encoder: Bidirectional RNN



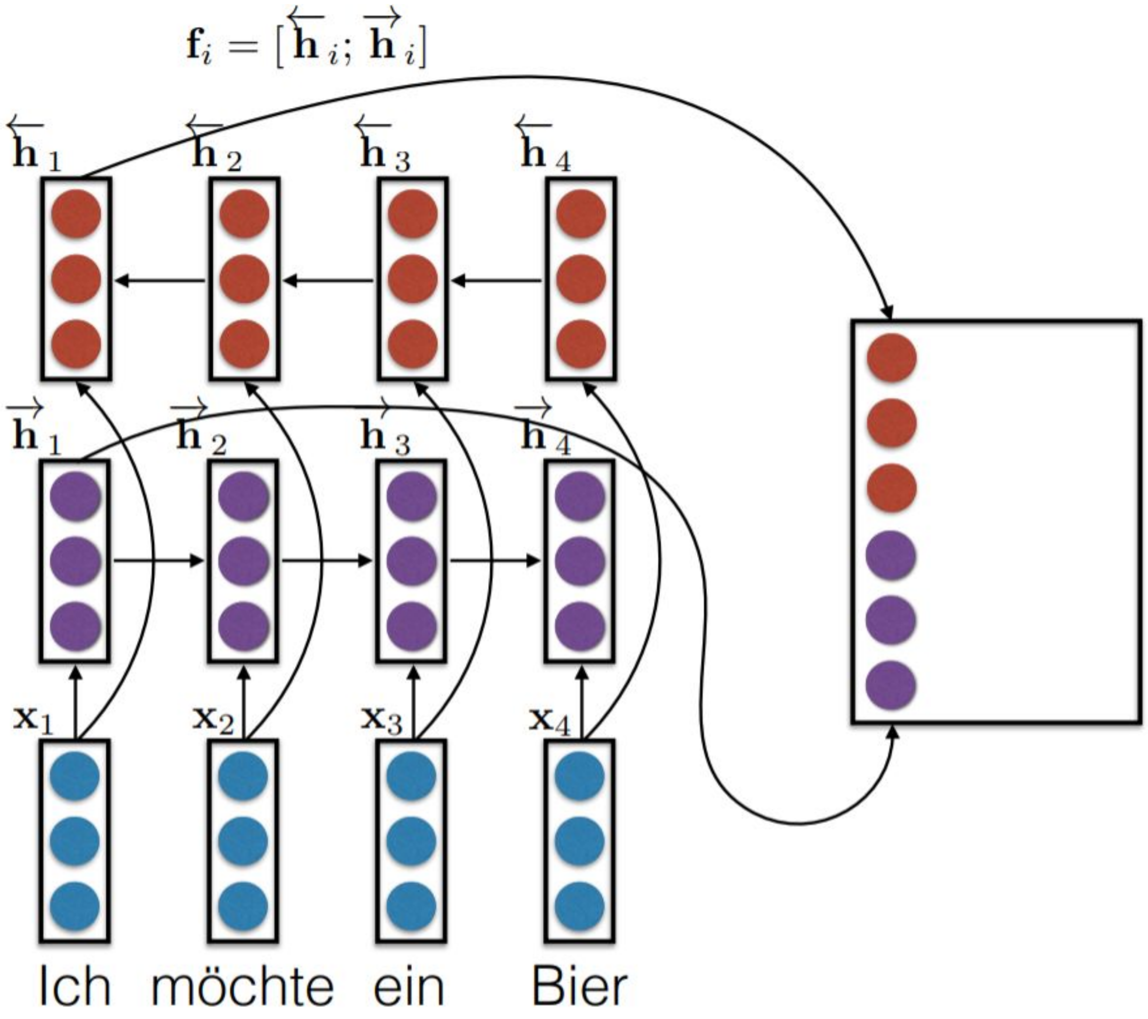
Encoder: Bidirectional RNN



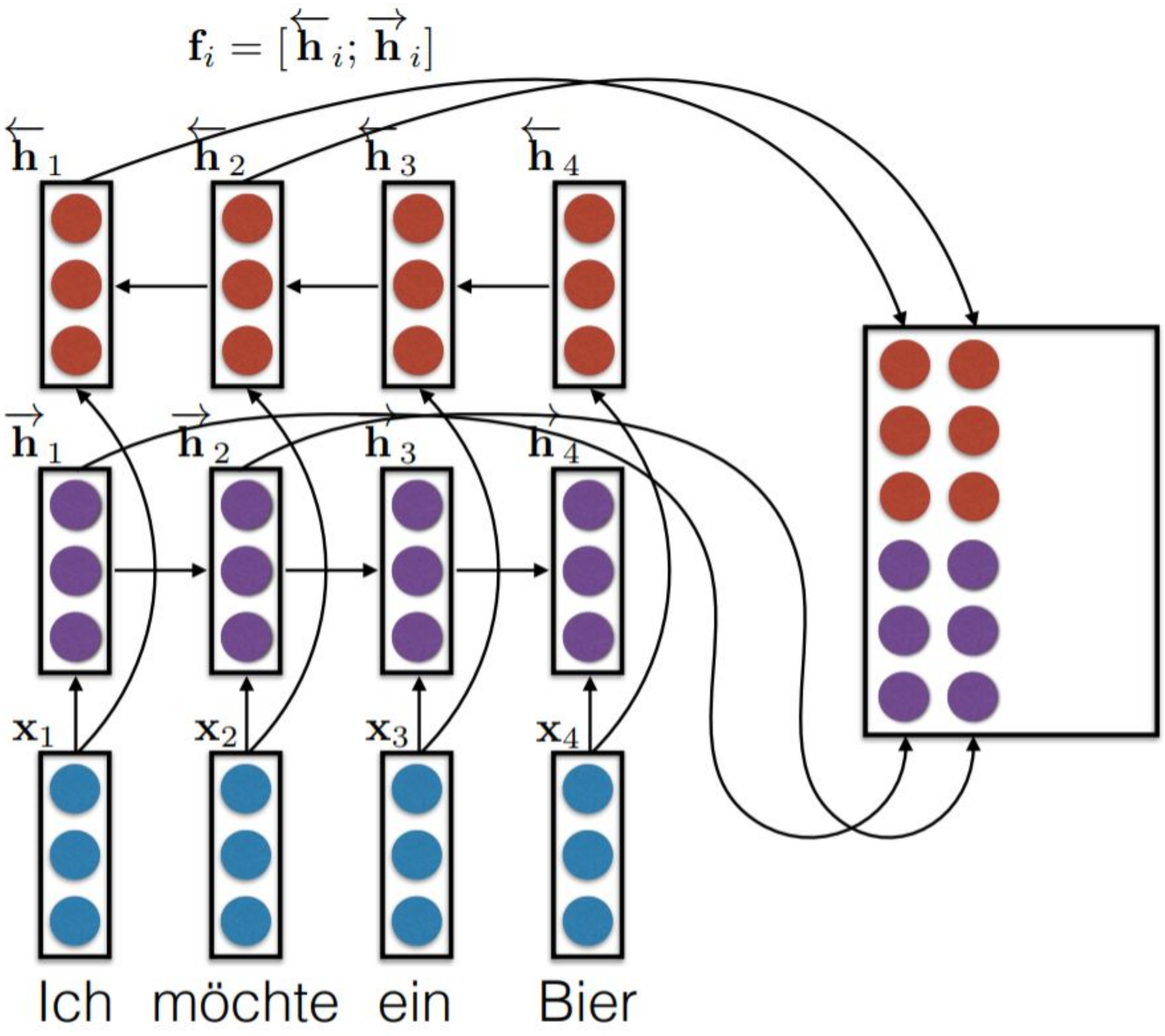
Encoder: Bidirectional RNN



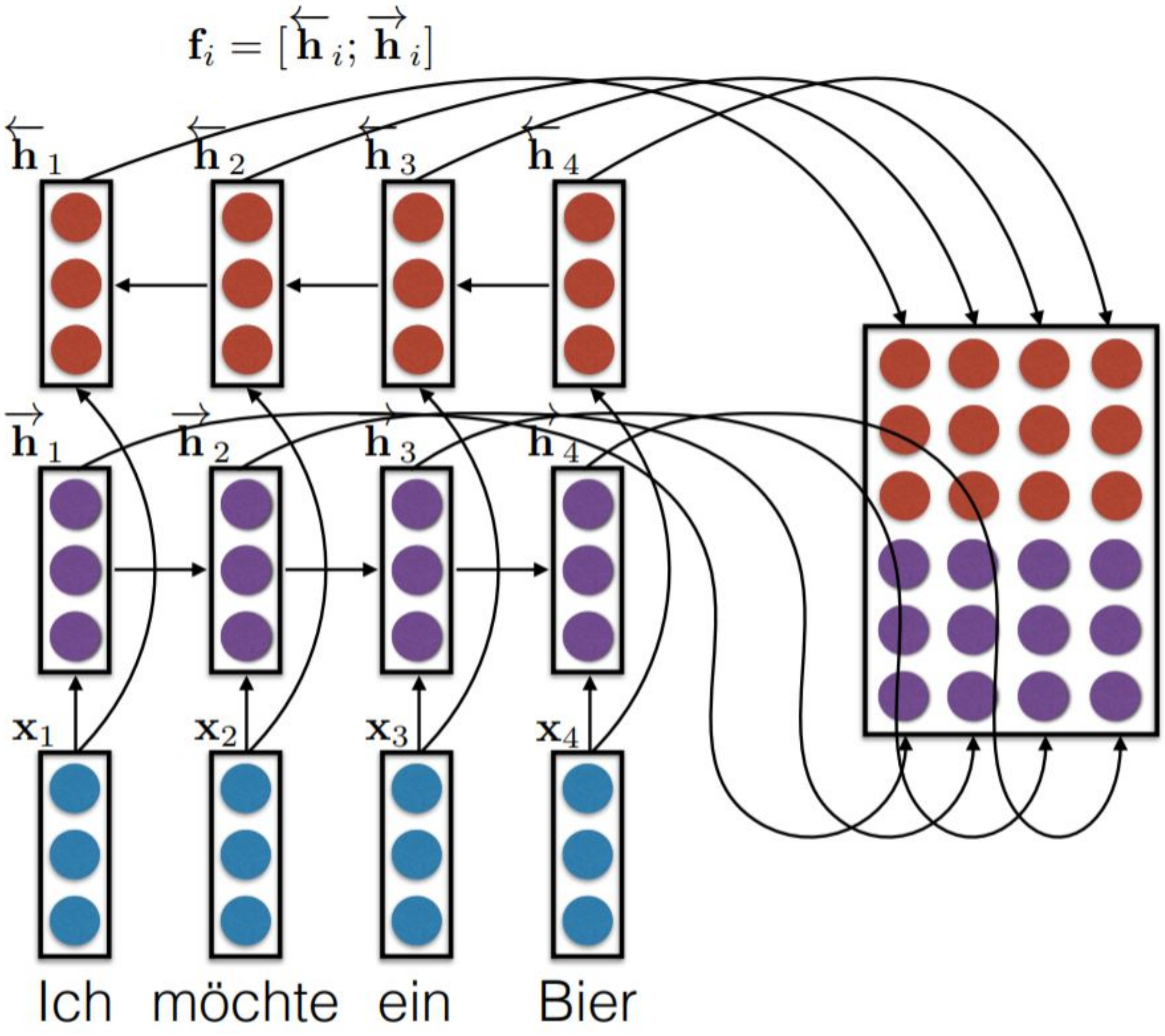
Encoder: Bidirectional RNN



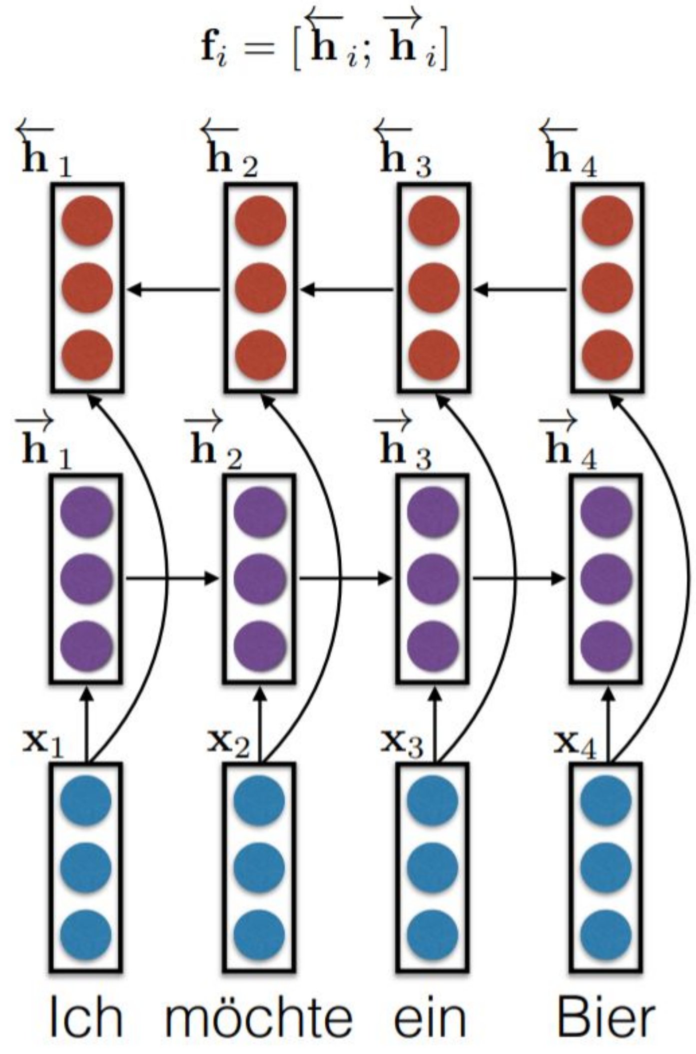
Encoder: Bidirectional RNN

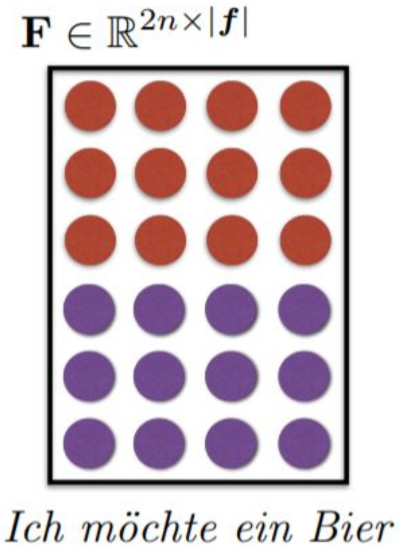


Encoder: Bidirectional RNN



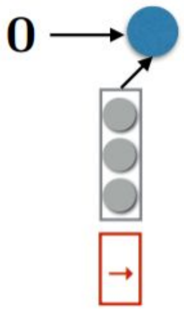
Matrix Sentence Encoding

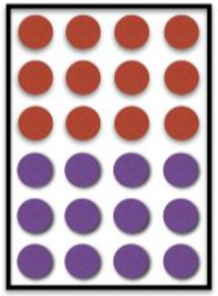




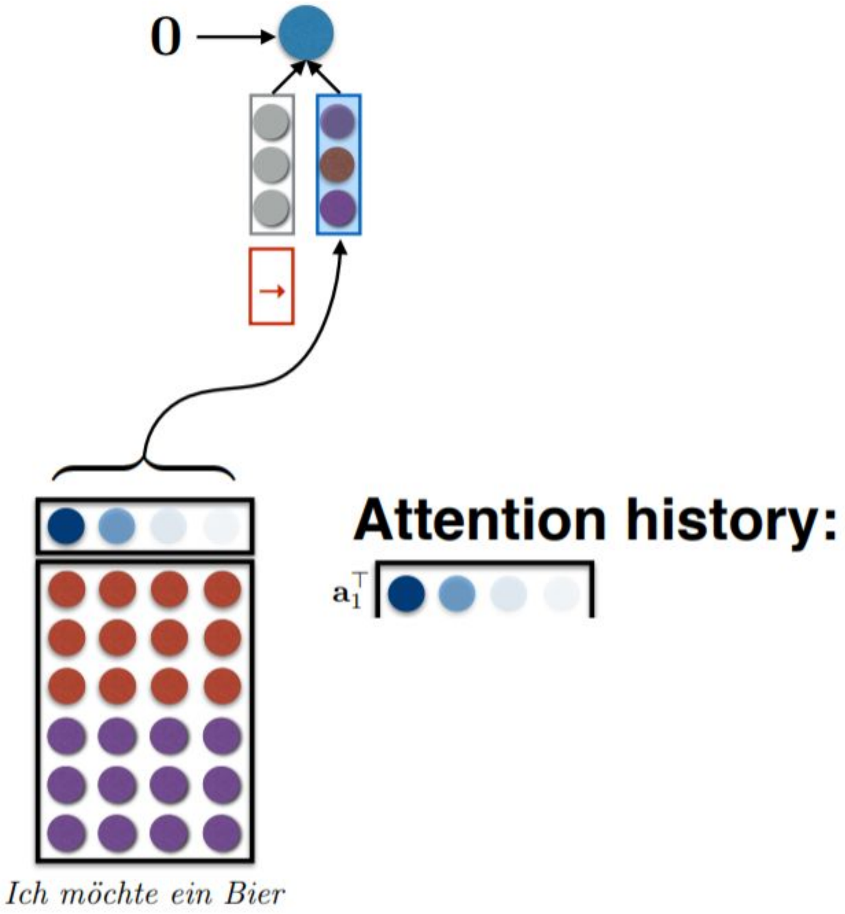
- matrix-encoded sentence

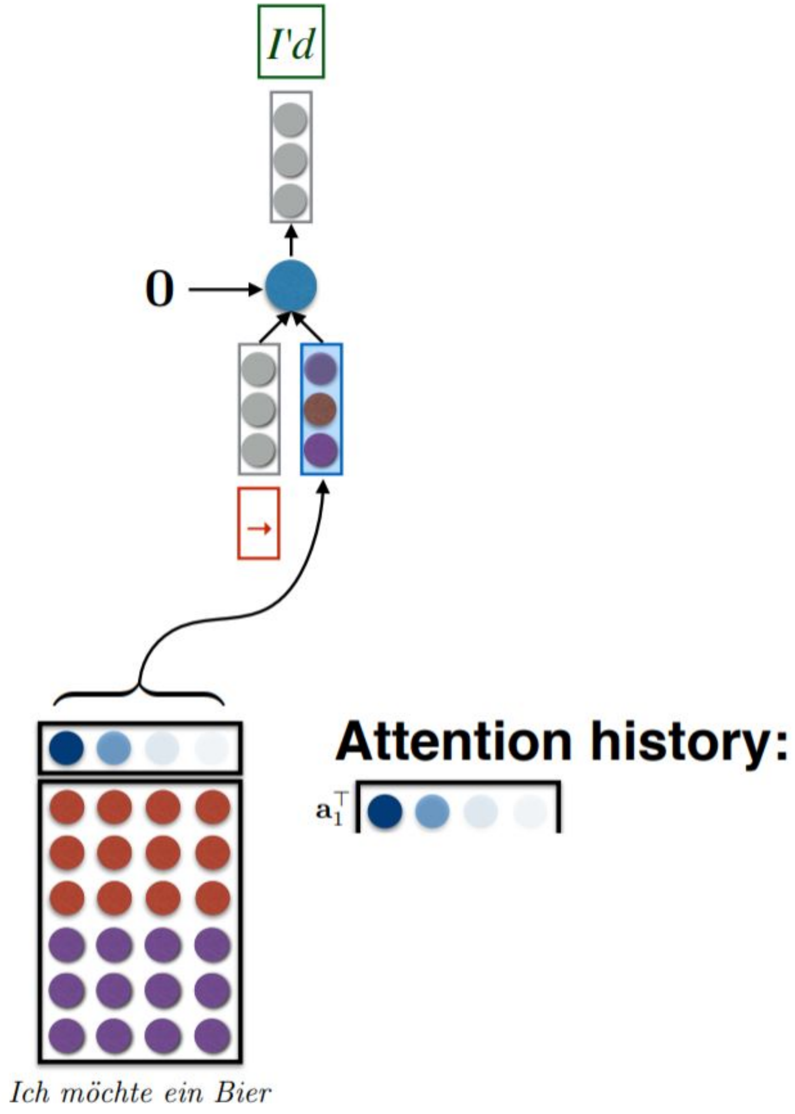
Decoder: RNN + Attention

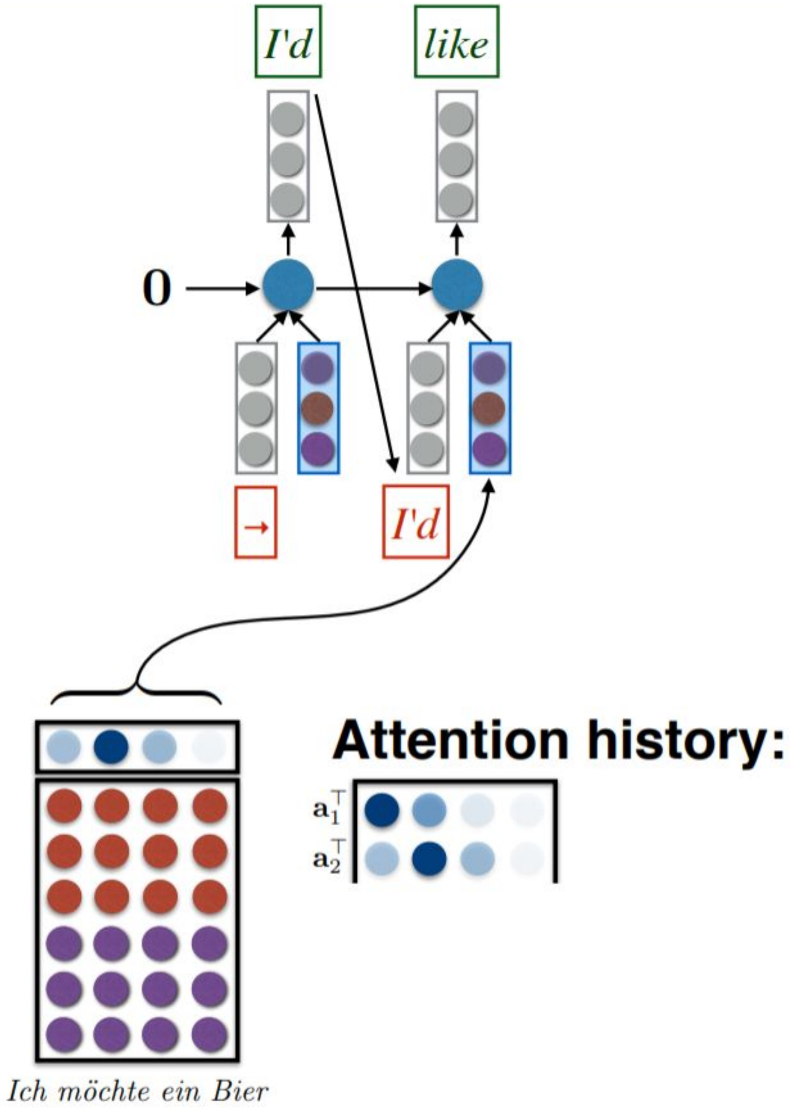


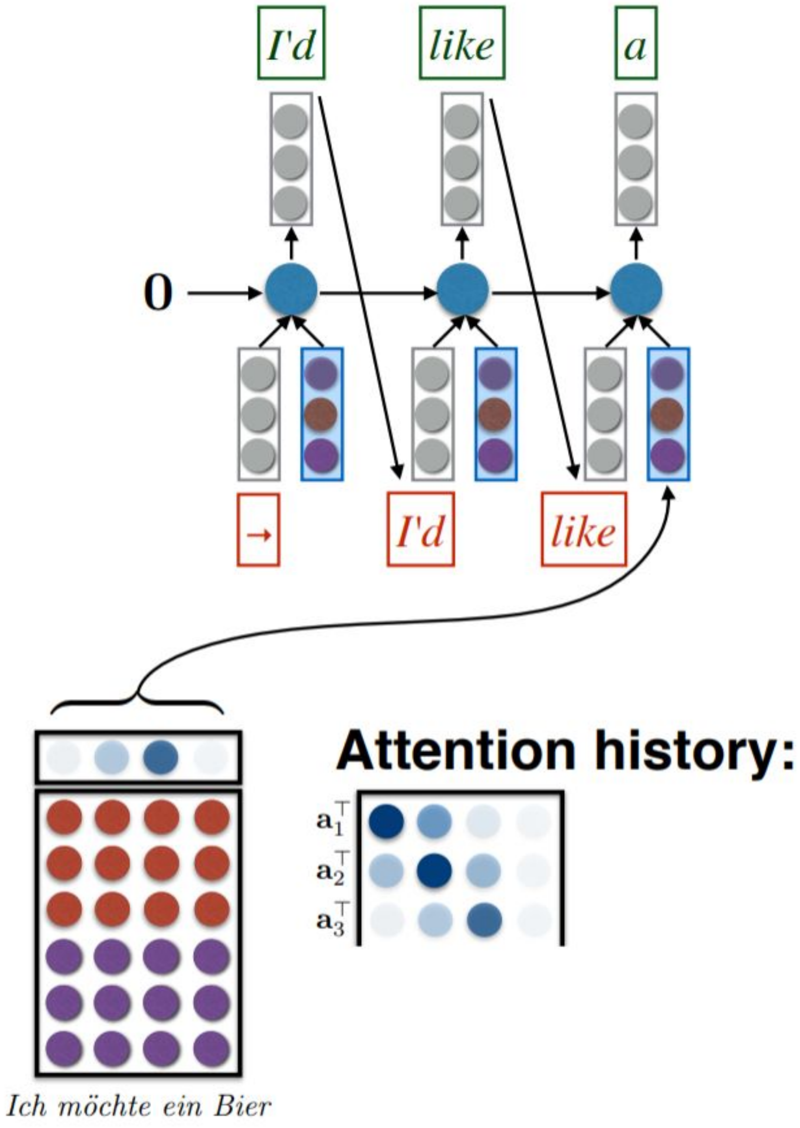


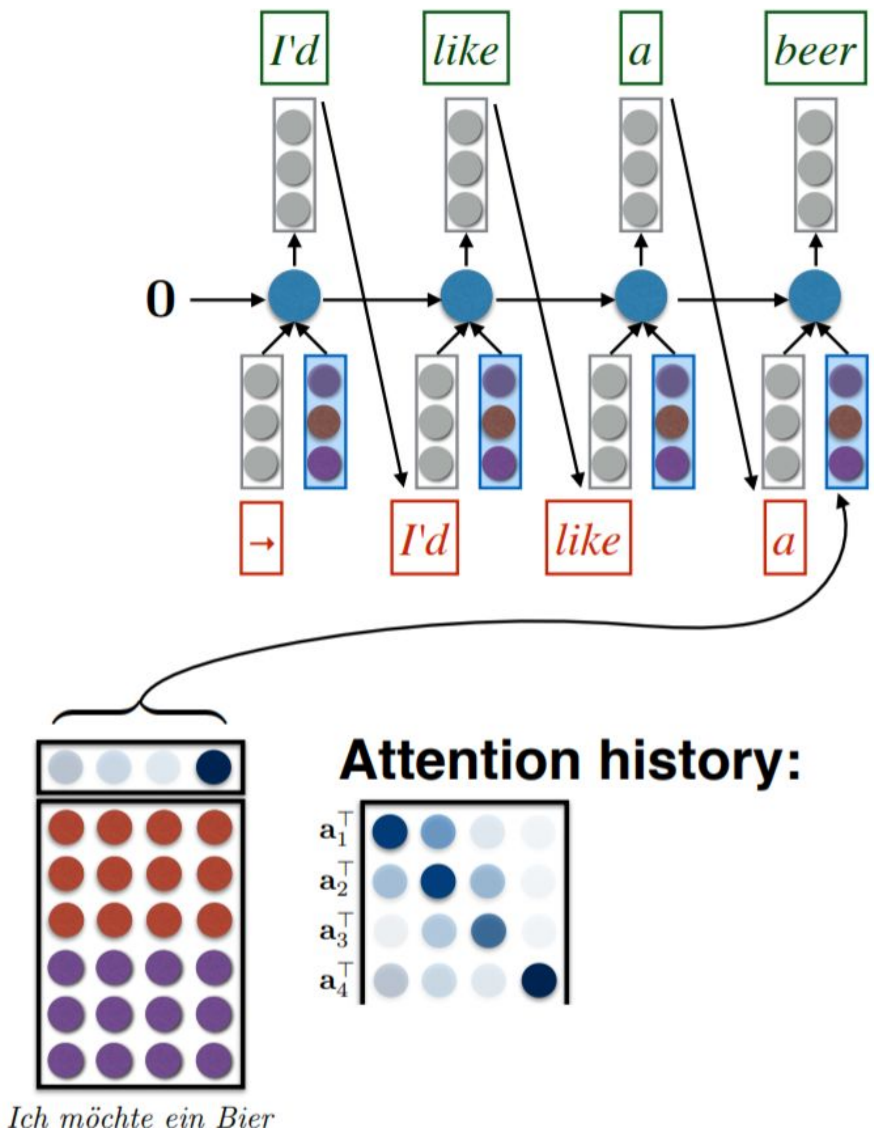
Ich möchte ein Bier


0
Ich möchte ein Bier
Attention history:
$\mathbf{a}_1^\top$

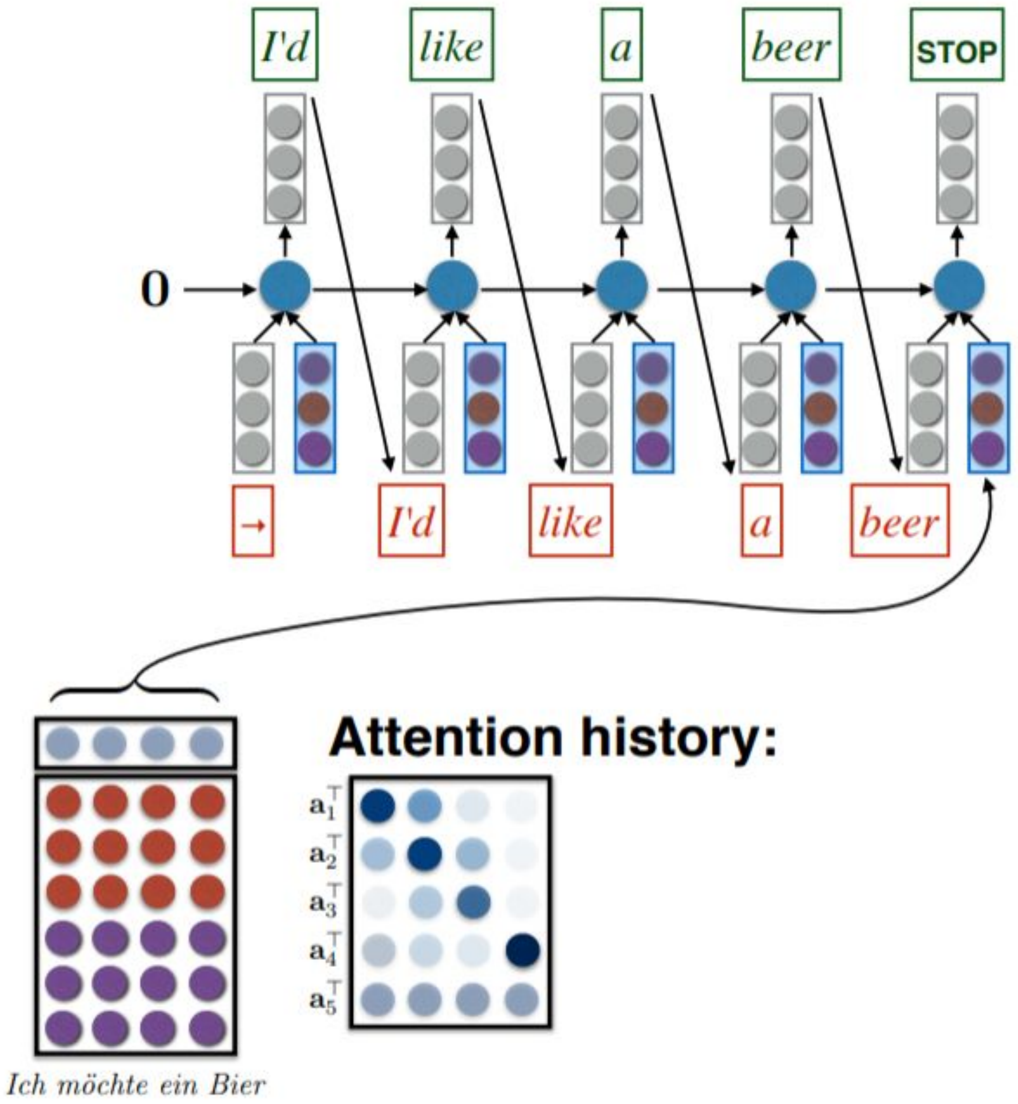

I'd
0
Attention history:
$\mathbf{a}_1^\top$
Ich möchte ein Bier


I'd
like
0
I'd
Ich möchte ein Bier
Attention history:
$\mathbf{a}_1^\top$
$\mathbf{a}_2^\top$


I'd
like
a
0
→
I'd
like
Attention history:
$\mathbf{a}_1^\top$
$\mathbf{a}_2^\top$
$\mathbf{a}_3^\top$
Ich möchte ein Bier



I'd
like
a
beer
0
→
I'd
like
a
Attention history:
$a_1^\top$
$a_2^\top$
$a_3^\top$
$a_4^\top$
Ich möchte ein Bier



I'd
like
a
beer
STOP
0
→
I'd
like
a
beer
Attention history:
$a_1^\top$
$a_2^\top$
$a_3^\top$
$a_4^\top$
$a_5^\top$
Ich möchte ein Bier

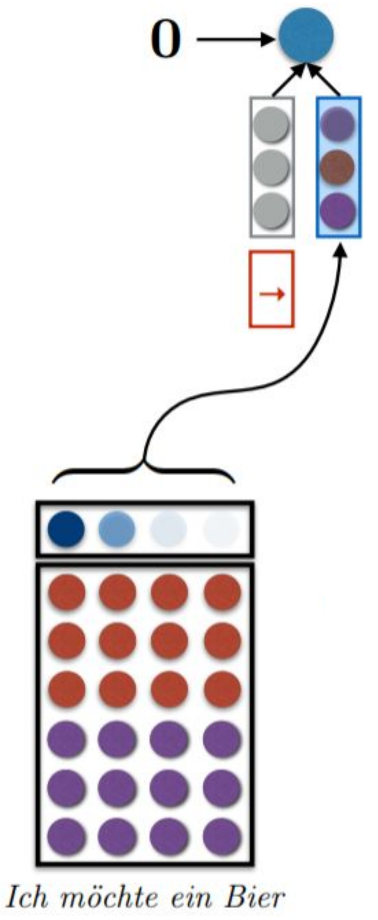
- Bahdanau et al. (2015) were the first to propose using **attention** for translating from matrix-encoded sentences
- High-level idea
  - Generate the output sentence word by word using an RNN
  - At each output position *t*, the RNN receives two inputs (in addition to any recurrent inputs)
    - a fixed-size vector embedding of the previously generated output symbol e_{t-1}
    - a fixed-size vector encoding a "view" of the input matrix
  - How do we get a fixed-size vector from a matrix that changes over time?
    - Bahdanau et al: do a weighted sum of the columns of input words based on how important they are at the *current time step*.
    - The weighting of the input columns at each time-step is called attention

Attention

- How do we know what to attend to at each timestep?

Attention

Minh-Thang Luong, Hieu Pham, Christopher D. Manning. 2015. Effective Approaches to Attention-based Neural Machine Translation. *Proc. EMNLP*

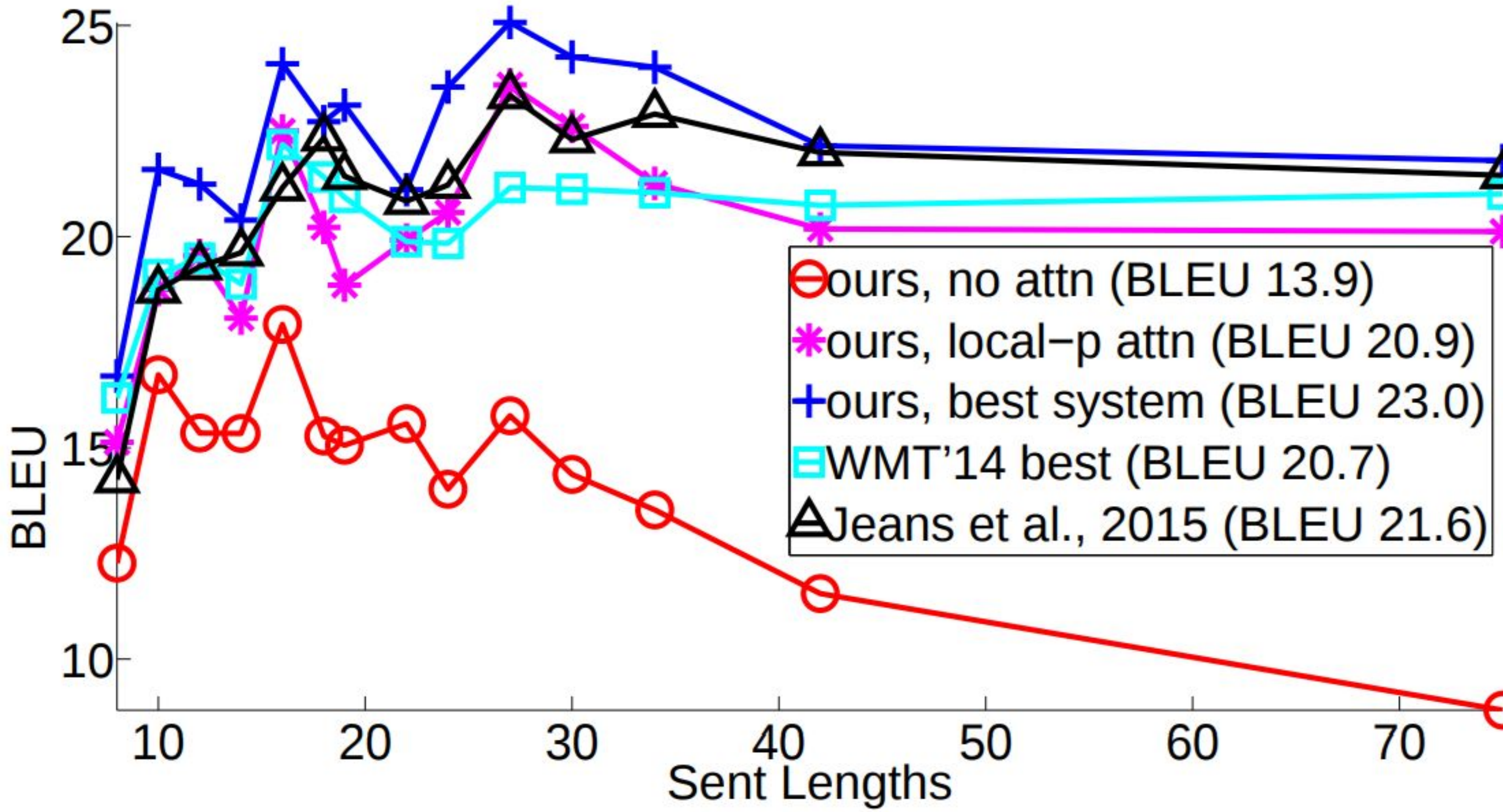


$$\mathrm{score}(\boldsymbol{h}_t, \bar{\boldsymbol{h}}_s) = \begin{cases} \boldsymbol{h}_t^\top \bar{\boldsymbol{h}}_s \\ \boldsymbol{h}_t^\top \boldsymbol{W}_a \bar{\boldsymbol{h}}_s \\ \boldsymbol{v}_a^\top \tanh\left(\boldsymbol{W}_a [\boldsymbol{h}_t; \bar{\boldsymbol{h}}_s]\right) \end{cases}$$

(Luong et al. ‘15)

(Bahdanau et al’15)

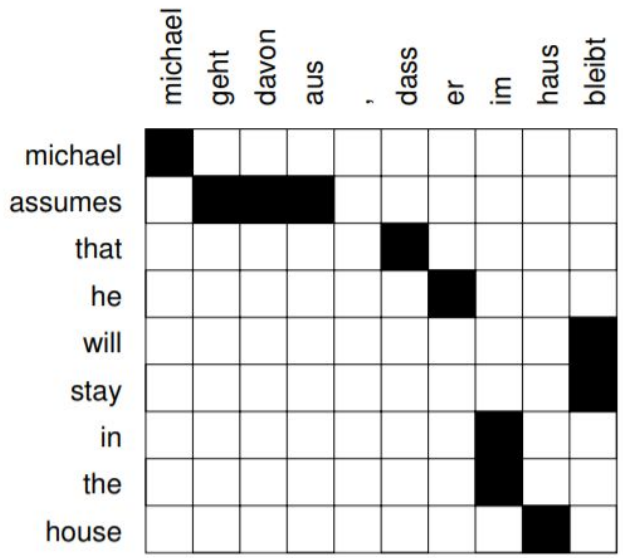
Attention

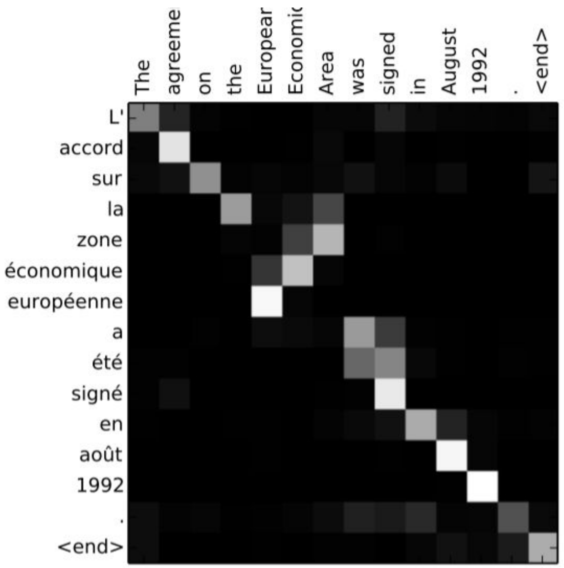


Minh-Thang Luong, Hieu Pham, Christopher D. Manning. 2015. Effective Approaches to Attention-based Neural Machine Translation. *Proc. EMNLP*

Attention vs Alignment

- Attention is similar to alignment, but there are important differences

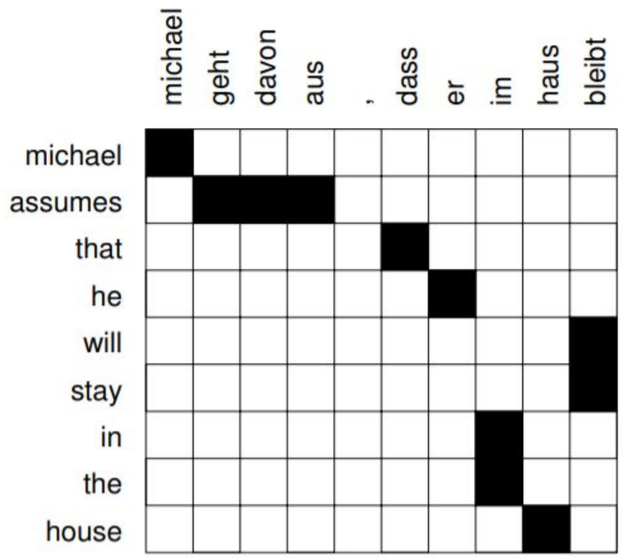


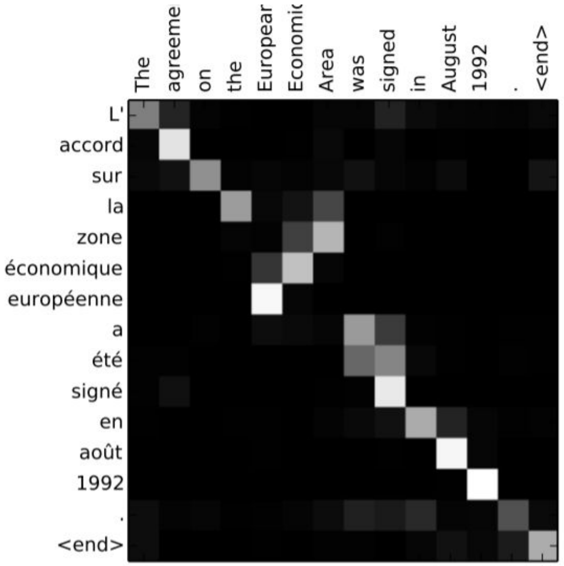


(Cho et al. 2015)

Attention vs Alignment

- Attention is similar to alignment, but there are important differences
  - alignment makes stochastic but hard decisions: the model picks one word or phrase at a time
  - attention is "soft" (you add together all the words)
  - there is no guarantee that attention corresponds to alignment since information can also flow along recurrent connections

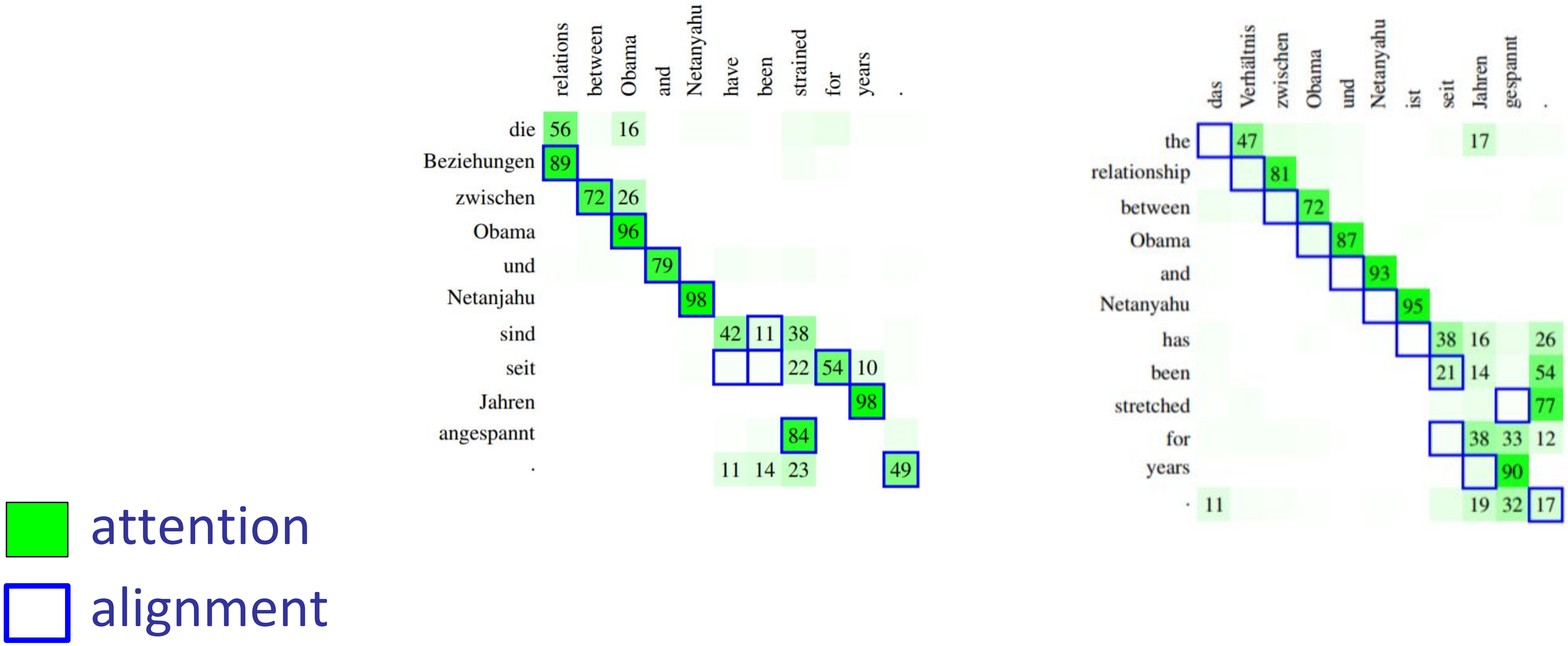




(Cho et al. 2015)

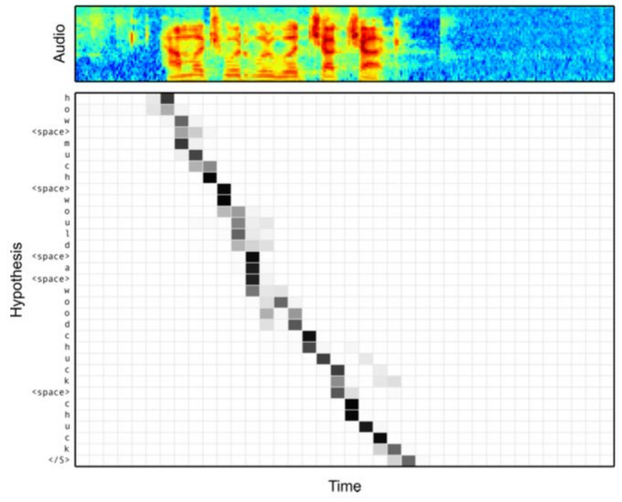
Attention is not Alignment!

Philipp Koehn, Rebecca Knowles. 2017. Six Challenges for Neural Machine Translation. *Proc. WNMT*



Conditional Language Models

- Speech recognition
- Vision
  - Image captioning
- NLP
  - NMT
  - Summarization
  - QA
  - Dialogue



(Chan et al. 2015)

. Examples of attending to the correct object (*white* indicates the attended regions, *underlines* indicated the corresponding word)

A woman is throwing a frisbee in a park.
A dog is standing on a hardwood floor.
A stop sign is on a road with a mountain in the background.

Show, Attend and Tell: Neural Image Caption Generation with Visual Attention (Xu et al. 2016)

Decoding

- Exact search
  - generate every possible sentence T in target language
  - compute score p(T|S) for each
  - pick best one

Decoding

- Exact search
  - generate every possible sentence T in target language
  - compute score p(T|S) for each
  - pick best one

  $\longrightarrow$ intractable: $|\text{vocab}|^N$ translations for output length N

Decoding

- Exact search
  - generate every possible sentence T in target language
  - compute score p(T|S) for each
  - pick best one

  $\longrightarrow$ intractable: $|\text{vocab}|^N$ translations for output length N
- Greedy search
  - at each time stamp pick the most likely word
    argmax log $p(y_i|S, y<i)$
  - until <EOS>

Decoding

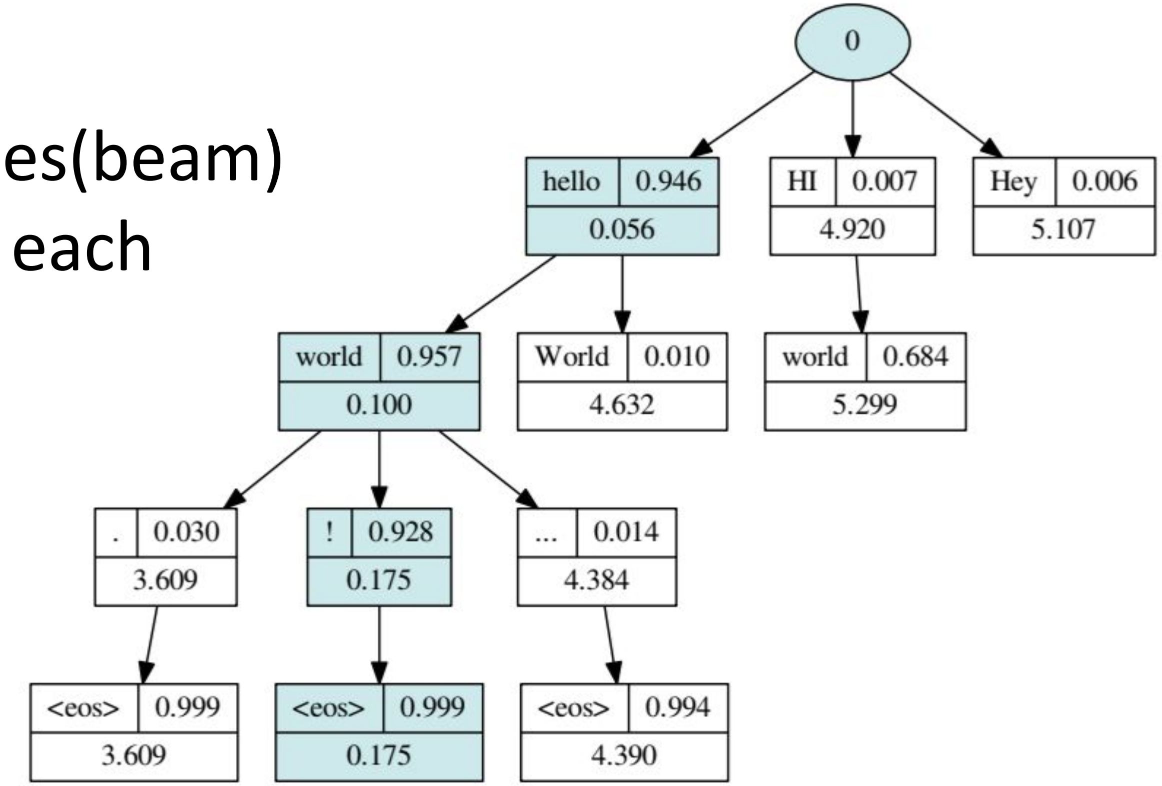
- Exact search
  - generate every possible sentence T in target language
  - compute score p(T|S) for each
  - pick best one

  $\longrightarrow$ intractable: $|\text{vocab}|^N$ translations for output length N
- Greedy search
  - at each time stamp pick the most likely word
    argmax log p(yi|S, y<i)
  - until <EOS>

  $\longrightarrow$ efficient, but heavily suboptimal

Decoding

- Beam search
  - maintain list of K hypotheses(beam)
  - at each time step, expand each hypothesis
  - select hypotheses with highest total probability



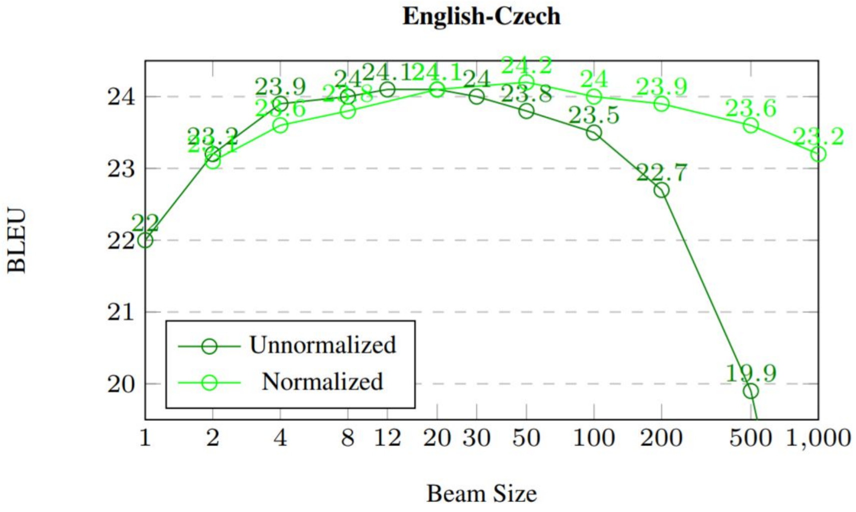
$$K = 3$$

Image thanks to Rico Sennrich

Decoding with Beam Search

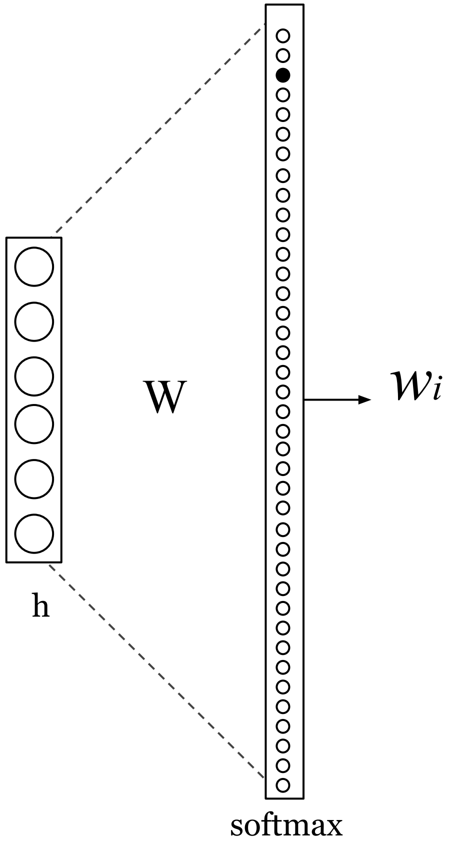
Strategy	# Chains	Valid Set		Test Set	
		NLL	BLEU	NLL	BLEU
Ancestral Sampling	50	22.98	15.64	26.25	16.76
Greedy Decoding	-	27.88	15.50	26.49	16.66
Beamsearch	5	20.18	17.03	22.81	18.56
Beamsearch	10	19.92	17.13	22.44	18.59

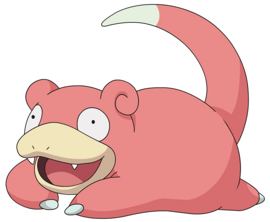
(Cho 2016)



(Koehn & Knowles 2017)

Dealing with Very Large Vocabulary





$$P(w_i|h) = \frac{e^{s_i}}{\sum_j e^{s_j}}$$

This is a <unk> sentence with very <unk> <unk> and <unk>.

Dealing with Very Large Vocabulary

- Sampling-based approximations
  - Importance Sampling: evaluate the denominator over a subset
  - Noise Contrastive Estimation: convert to a proxy binary classification problem
- Structure-based approximations
  - Class-based Softmax: divide the vocabulary to multiple classes; first predict a class, then predict a word of the class
  - Hierarchical Softmax: binary tree with words as leaves
- Self normalization (Devlin et al. '2014, Andreas et al. 2015)
- Subword Units
  - Byte Pair Encoding (BPE) (Sennrich et al. '2016)

    $\longrightarrow$ current standard

Byte pair encoding for word segmentation

- Repeatedly replace most frequent symbol pair (’A’,’B’) with ’AB’
-

<table>
<tr><th>system</th><th>sentence</th></tr>
<tr><td>source</td><td>health research institutes</td></tr>
<tr><td>reference</td><td>Gesundheitsforschungsinstitute</td></tr>
<tr><td>word-level (with back-off)</td><td>Forschungsinstitute</td></tr>
<tr><td>character bigrams</td><td>Fo|rs|ch|un|gs|in|st|it|ut|io|ne|n</td></tr>
<tr><td>BPE</td><td>Gesundheits|forsch|ungsin|stitute</td></tr>
<tr><td>source</td><td>rakfisk</td></tr>
<tr><td>reference</td><td>ракфиска (rakfiska)</td></tr>
<tr><td>word-level (with back-off)</td><td>rakfisk → UNK → rakfisk</td></tr>
<tr><td>character bigrams</td><td>ra|kf|is|k → ра|кф|ис|к (ra|kf|is|k)</td></tr>
<tr><td>BPE</td><td>rak|f|isk → рак|ф|иска (rak|f|iska)</td></tr>
</table>

Image thanks to Rico Sennrich

Alternative to Softmax

	Sampling Based	Structure Based	Subword Units
Training Time	☺	☺	☺
Test Time	😐	😐	☺
Accuracy	☹	☹	☺
Memory	😐	☹	☺
Handle Very Large Vocab	😐	😐	☺

■ Similar ■ Worse ■ Better ■ Much Better (>2X)

SMT vs NMT

SMT vs NMT

Pros of NMT

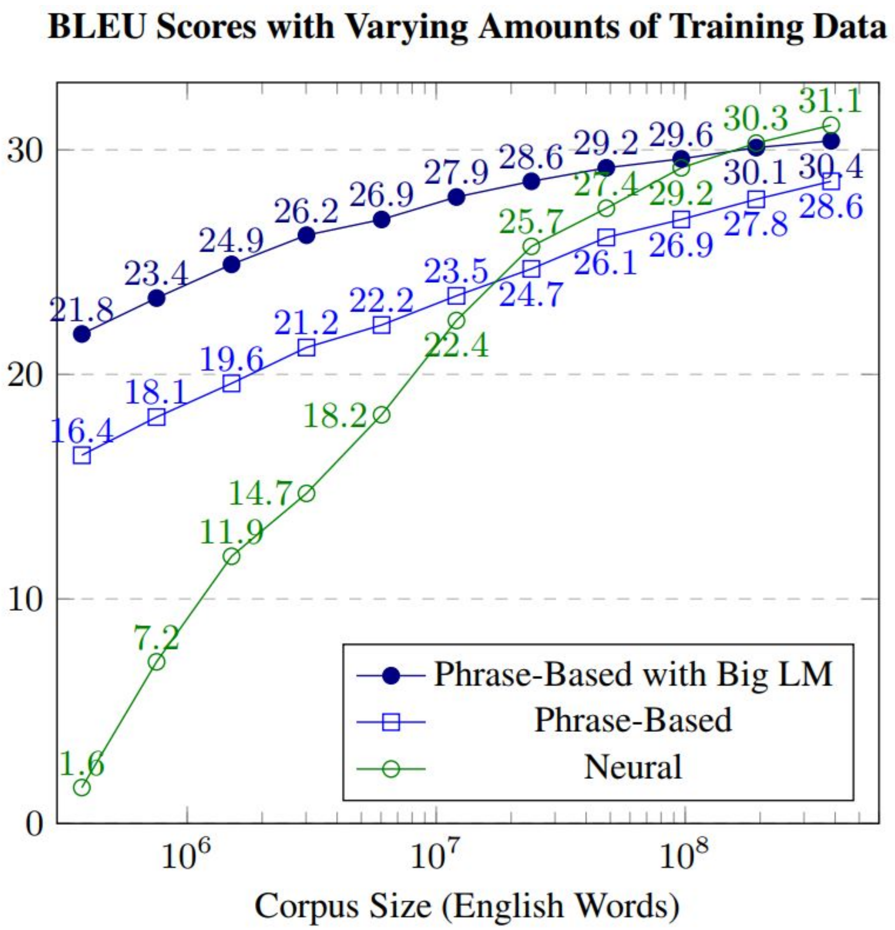
- simpler end-to-end pipeline
- output conditioned on full source text and target history
- continuous word representations better exploit similarities
- smaller model
- more fluent outputs

Cons of NMT

- poor interpretability
- hard to integrate knowledge
- data hungry, underperform in low-resource settings

PBMT vs NMT

Philipp Koehn, Rebecca Knowles. 2017. Six Challenges for Neural Machine Translation. *Proc. WNMT*



PBMT vs NMT

Philipp Koehn, Rebecca Knowles. 2017. Six Challenges for Neural Machine Translation. *Proc. WNMT*

System ↓	Law	Medical	IT	Koran	Subtitles
All Data	30.5 32.8	45.1 42.2	35.3 44.7	17.9 17.9	26.4 20.8
Law	31.1 34.4	12.1 18.2	3.5 6.9	1.3 2.2	2.8 6.0
Medical	3.9 10.2	39.4 43.5	2.0 8.5	0.6 2.0	1.4 5.8
IT	1.9 3.7	6.5 5.3	42.1 39.8	1.8 1.6	3.9 4.7
Koran	0.4 1.8	0.0 2.1	0.0 2.3	15.9 18.8	1.0 5.5
Subtitles	7.0 9.9	9.3 17.8	9.2 13.6	9.0 8.4	25.9 22.1

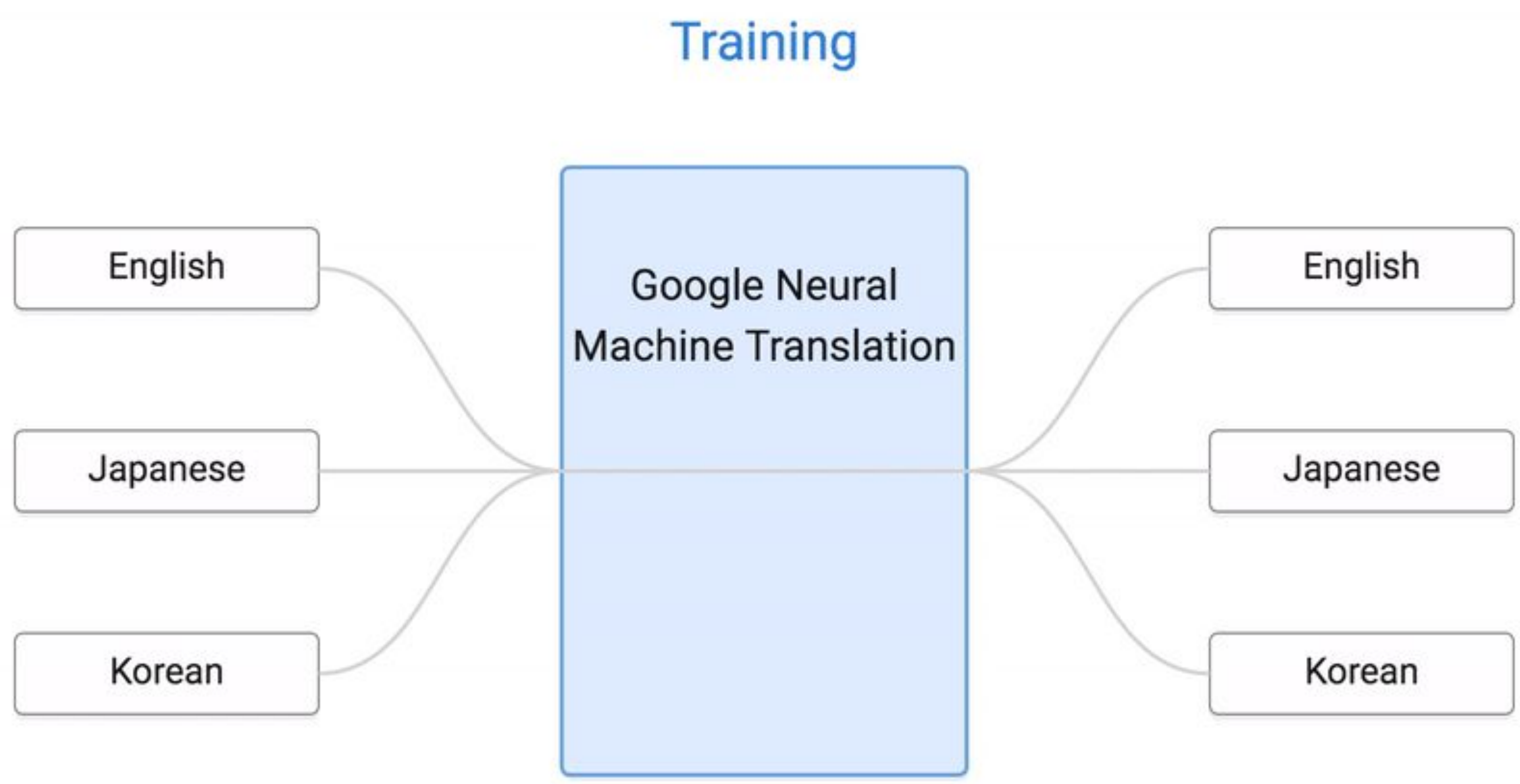
Figure 1: Quality of systems (BLEU), when trained on one domain (rows) and tested on another domain (columns). Comparably, NMT systems (left bars) show more degraded performance out of domain.

Case Studies

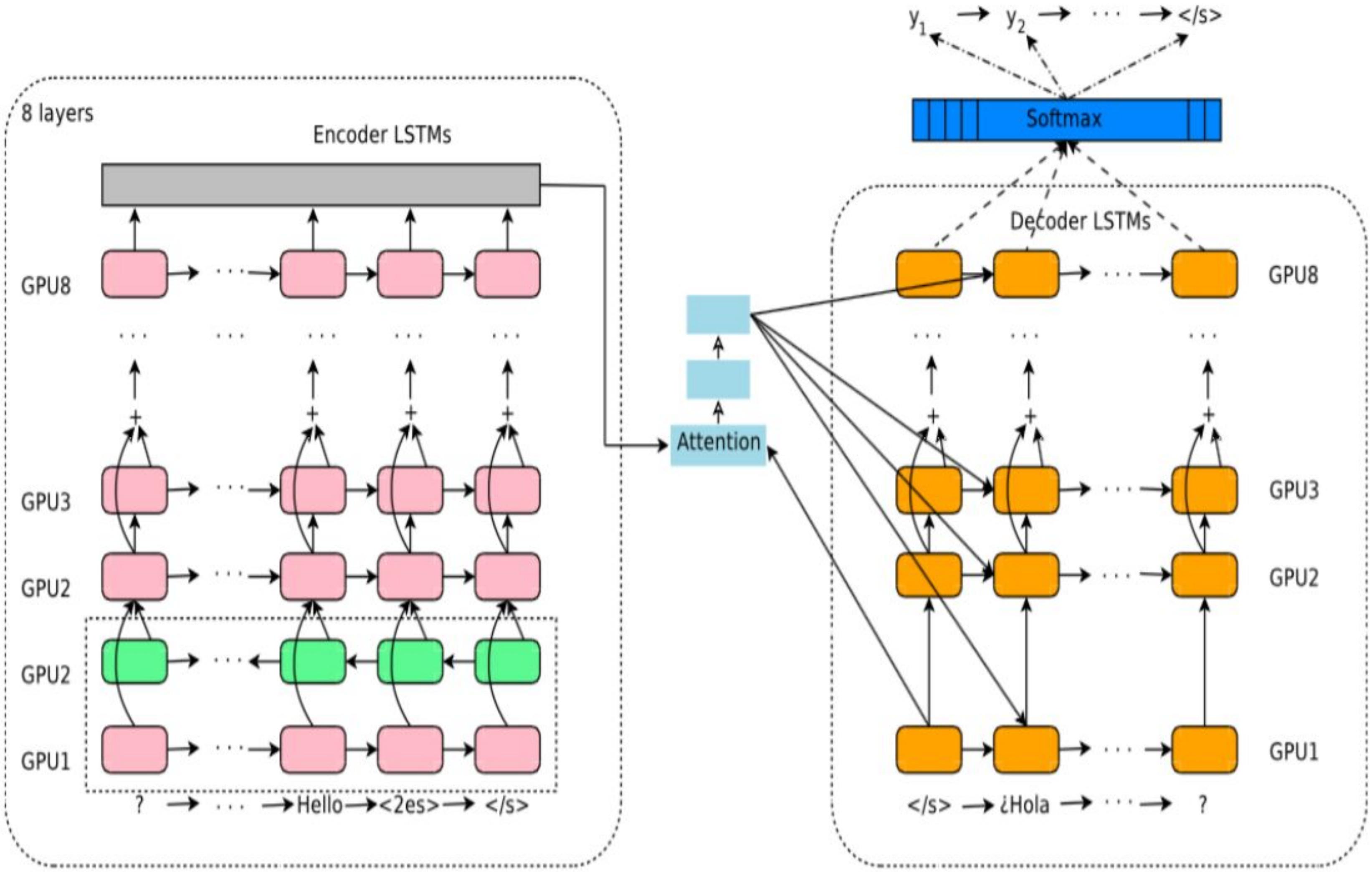
Multilingual NMT

Google's Multilingual Neural Machine Translation System: Enabling Zero-Shot Translation

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, Jeffrey Dean



Multilingual NMT



Multilingual NMT

Artificial token in the beginning of the input sentence to indicate the target language:

<2es> Hello, how are you? -> ¿Hola como estás?

Multilingual NMT

Table 5: Portuguese→Spanish BLEU scores using various models.

	Model	Zero-shot	BLEU
(a)	PBMT bridged	no	28.99
(b)	NMT bridged	no	30.91
(c)	NMT Pt→Es	no	31.50
(d)	Model 1 (Pt→En, En→Es)	yes	21.62
(e)	Model 2 (En↔{Es, Pt})	yes	24.75
(f)	Model 2 + incremental training	no	31.77

Transformers

Attention Is All You Need

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin

- SOTA results on WMT datasets
- Fast: only matrix multiplications

- stack of N self-attention layers
- self-attention in decoder is masked
- decoder also attends to encoder states
- RNN can learn to count raw text
- Transformer needs positional encoding

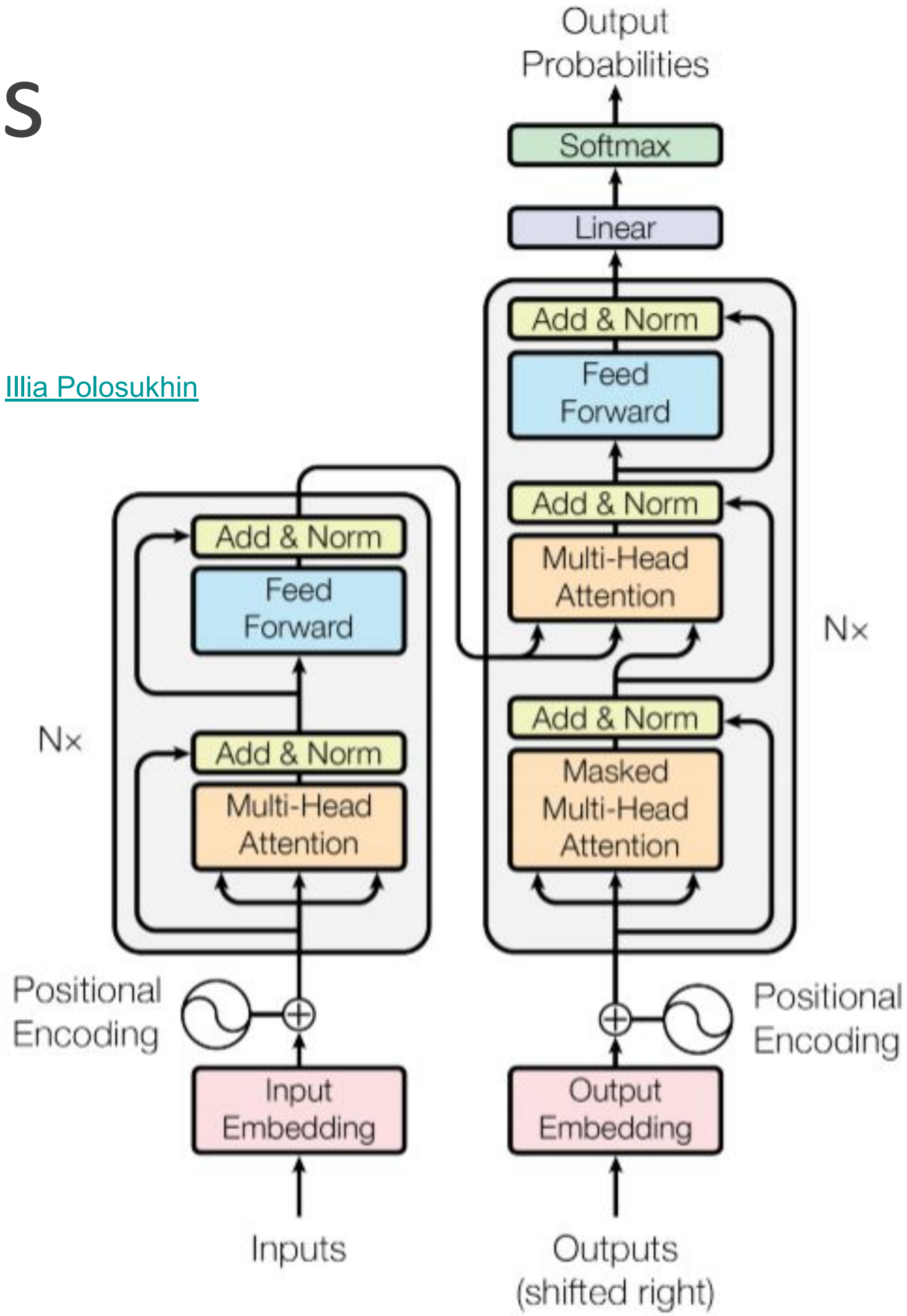
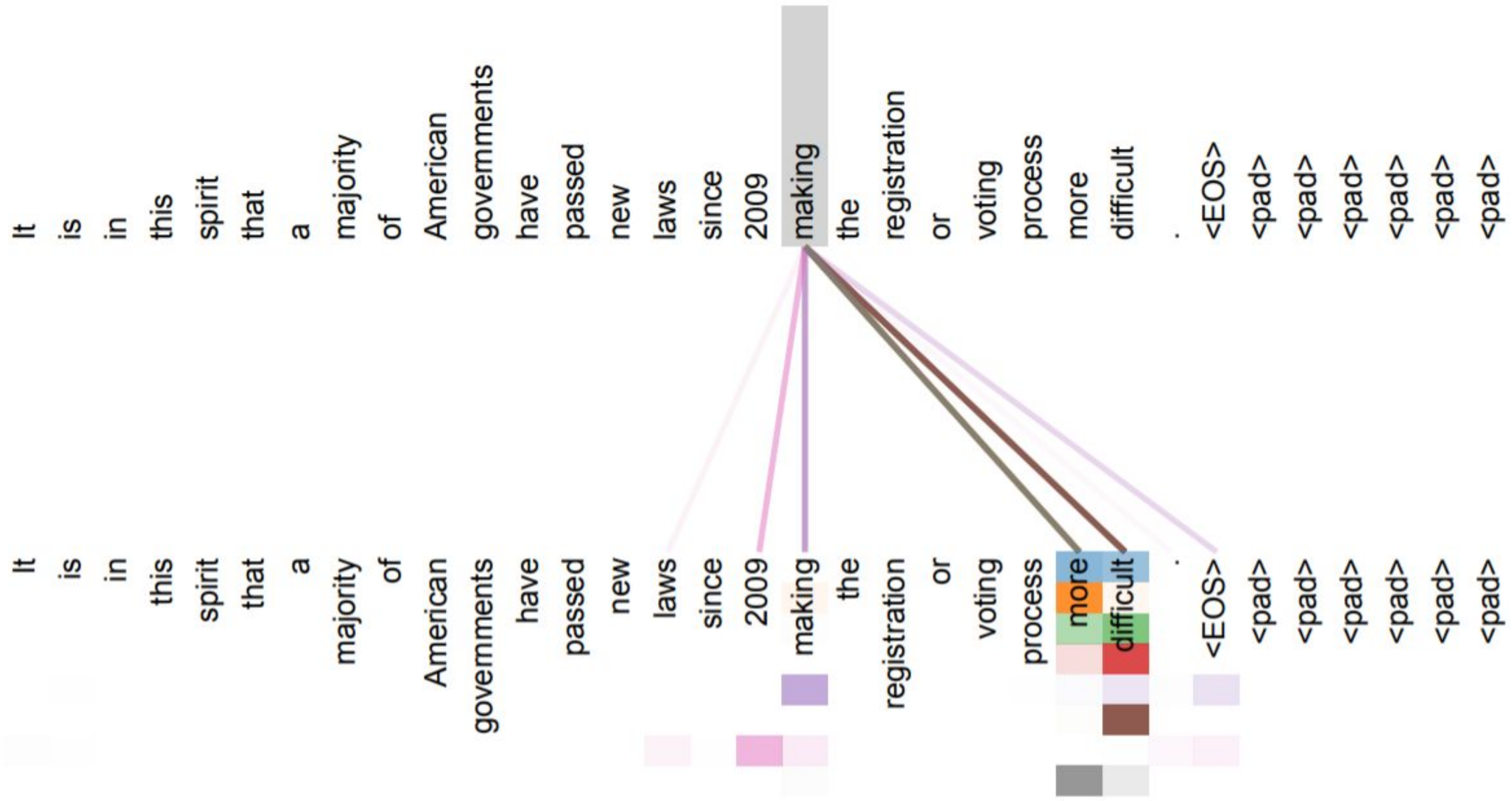


Figure 1: The Transformer - model architecture.

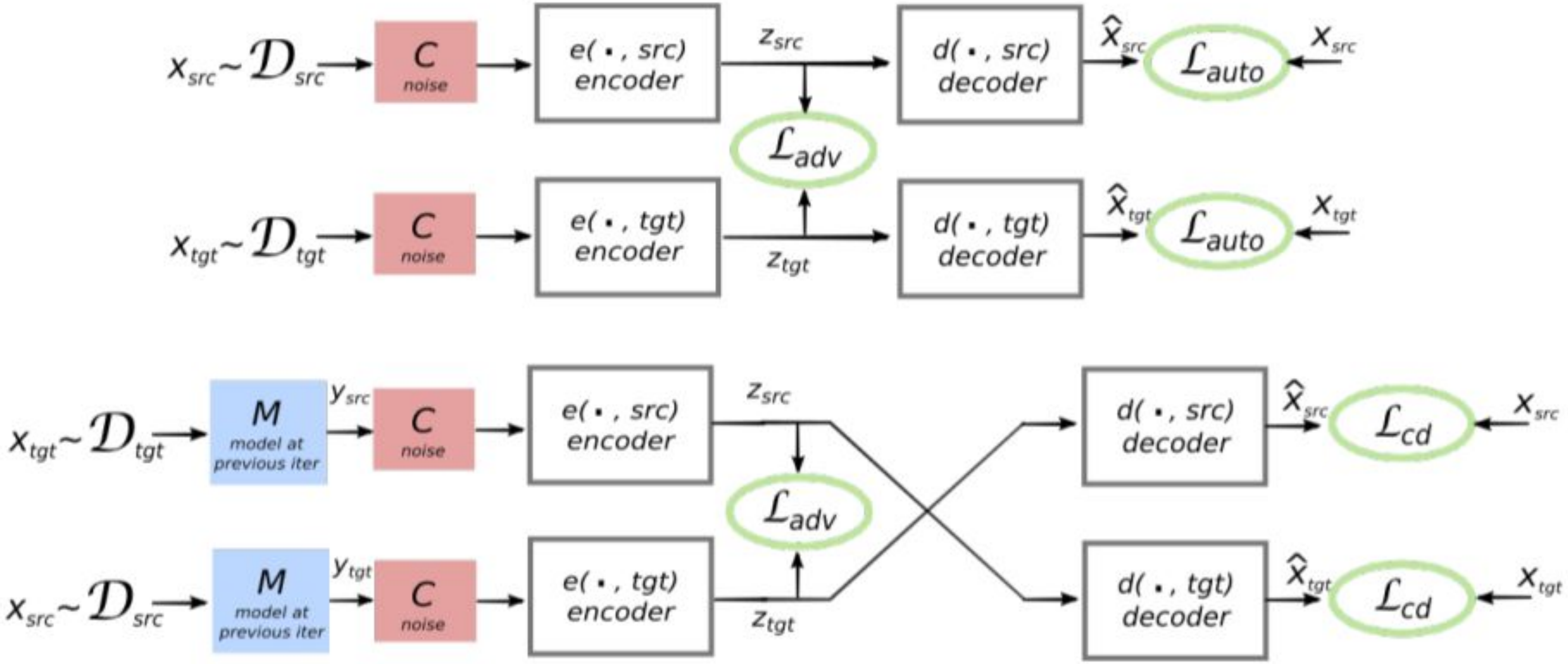
Multi-Head Attention



Unsupervised NMT

Unsupervised Machine Translation Using Monolingual Corpora Only

Guillaume Lample, Alexis Conneau, Ludovic Denoyer, Marc'Aurelio Ranzato



WMT competitions

http://www.statmt.org/wmt18/

THIRD CONFERENCE ON MACHINE TRANSLATION (WMT18)

October 31 — November 1, 2018
Brussels, Belgium

Home

[HOME] [SCHEDULE] [PAPERS]
TRANSLATION TASKS: [NEWS] [BIOMEDICAL] [MULTIMODAL]
EVALUATION TASKS: [METRICS] [QUALITY ESTIMATION]
OTHER TASKS: [AUTOMATIC POST-EDITING] [PARALLEL CORPUS FILTERING]

his conference builds on a series of annual workshops and conferences on statistical machine translation, going back to 2006:

- the NAACL-2006 Workshop on Statistical Machine Translation,
- the ACL-2007 Workshop on Statistical Machine Translation,
- the ACL-2008 Workshop on Statistical Machine Translation,
- the EACL-2009 Workshop on Statistical Machine Translation,
- the ACL-2010 Workshop on Statistical Machine Translation
- the EMNLP-2011 Workshop on Statistical Machine Translation,
- the NAACL-2012 Workshop on Statistical Machine Translation,
- the ACL-2013 Workshop on Statistical Machine Translation,
- the ACL-2014 Workshop on Statistical Machine Translation,
- the EMNLP-2015 Workshop on Statistical Machine Translation,
- the First Conference on Machine Translation (at ACL-2016).
- the Second Conference on Machine Translation (at EMNLP-2017).

WMT competitions

http://matrix.statmt.org/