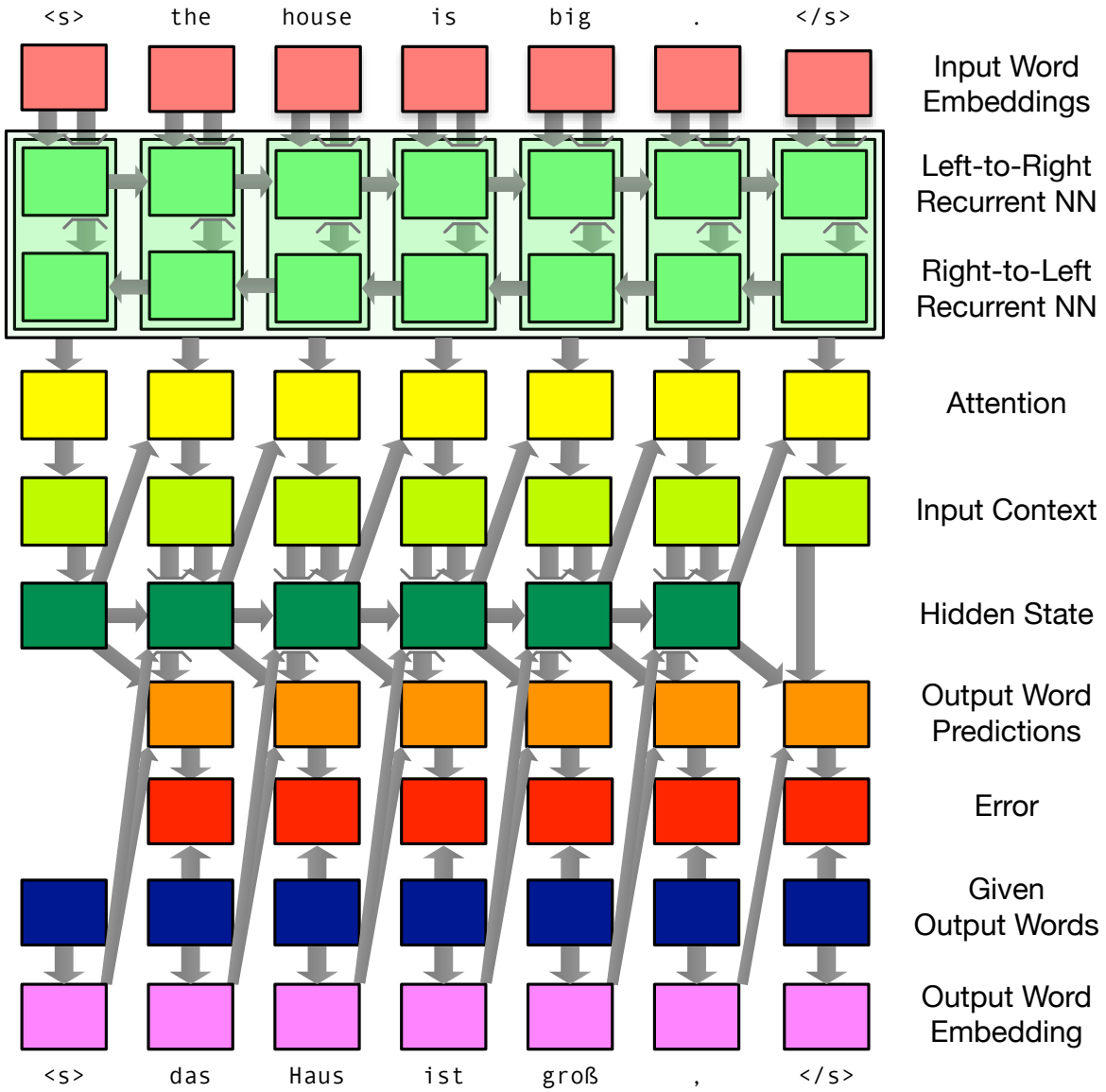
Neural Machine Translation III

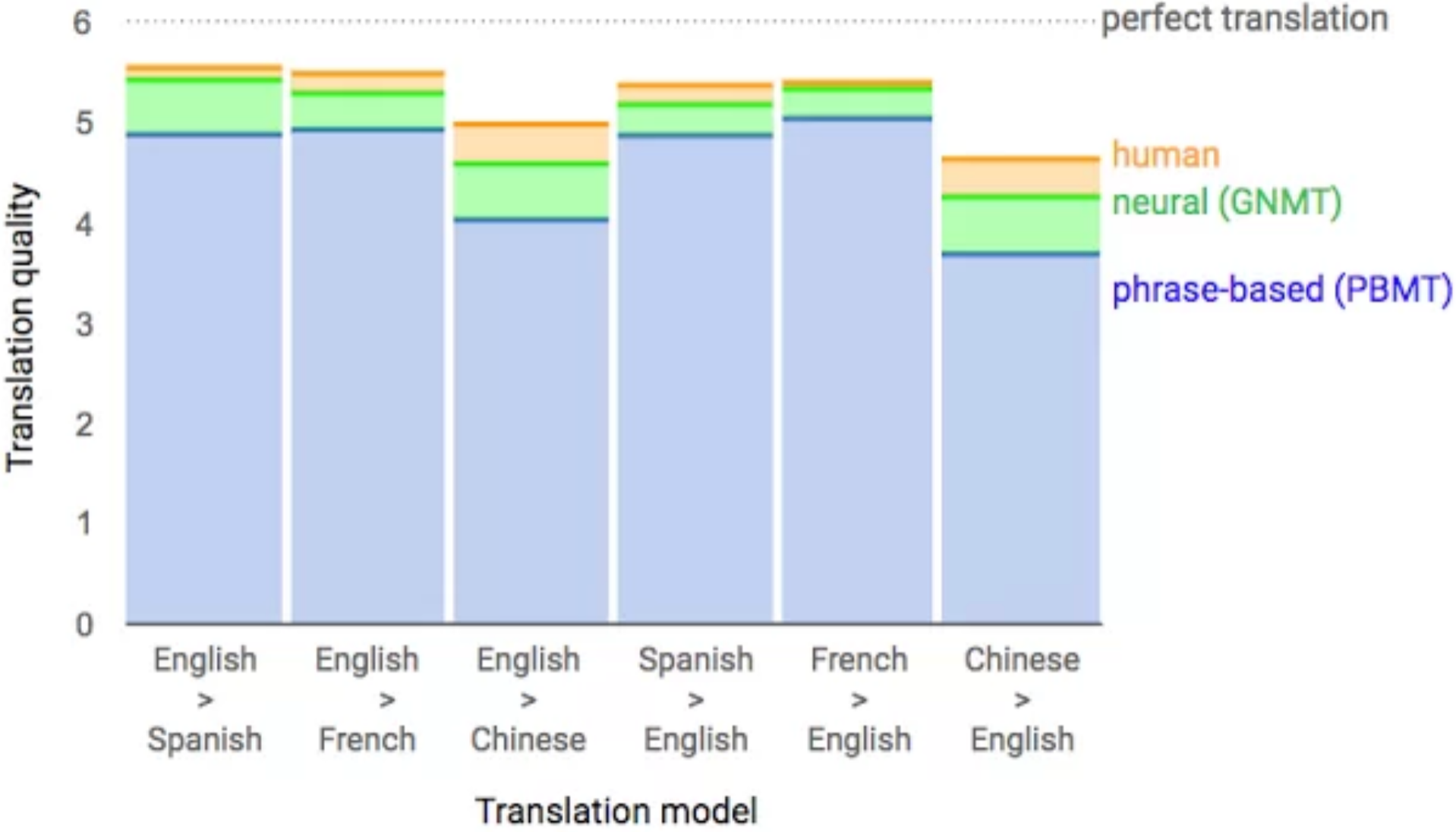
Philipp Koehn

24 October 2017

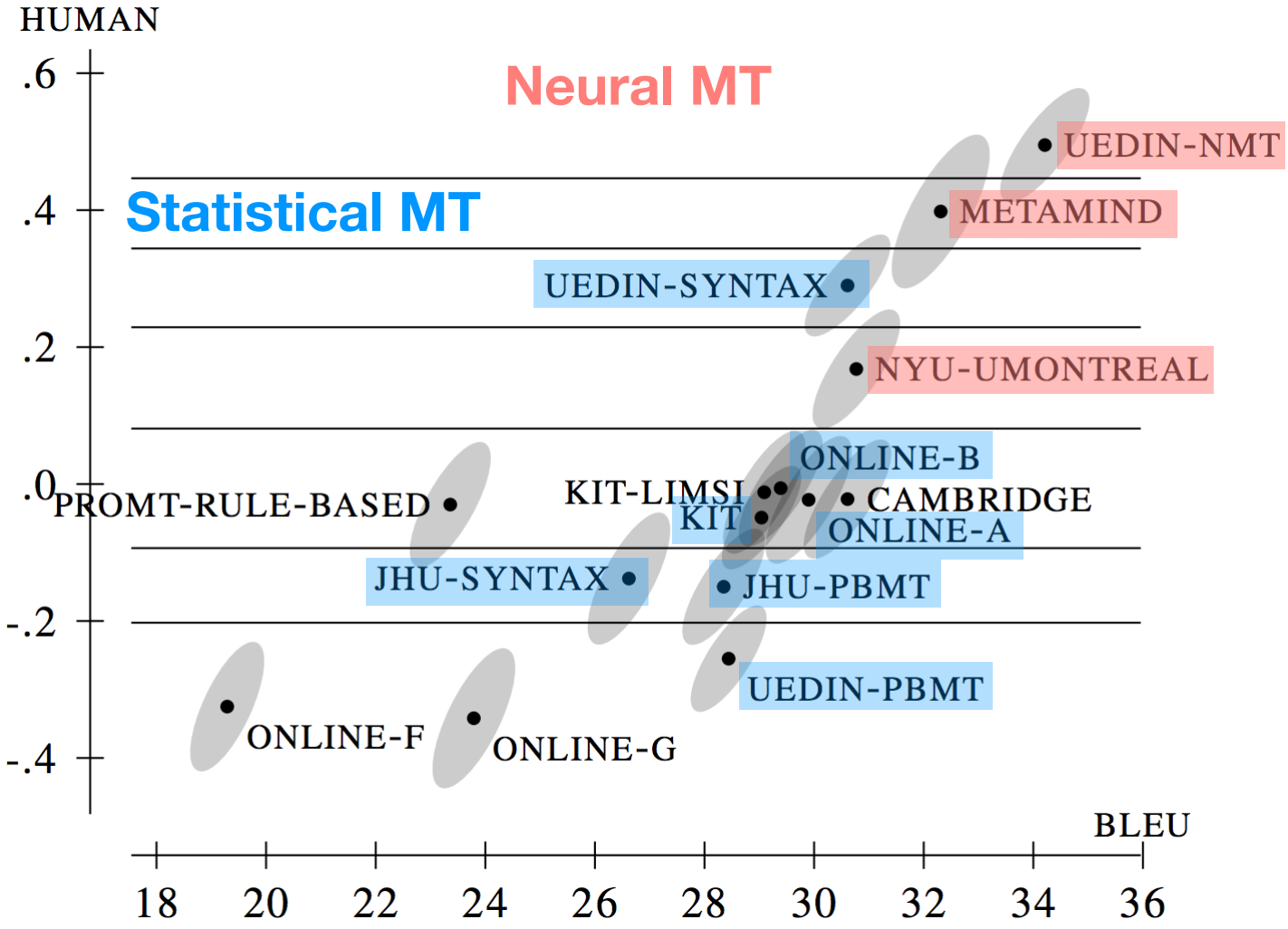
Neural Machine Translation



Google: Neural vs. Statistical MT



WMT 2016



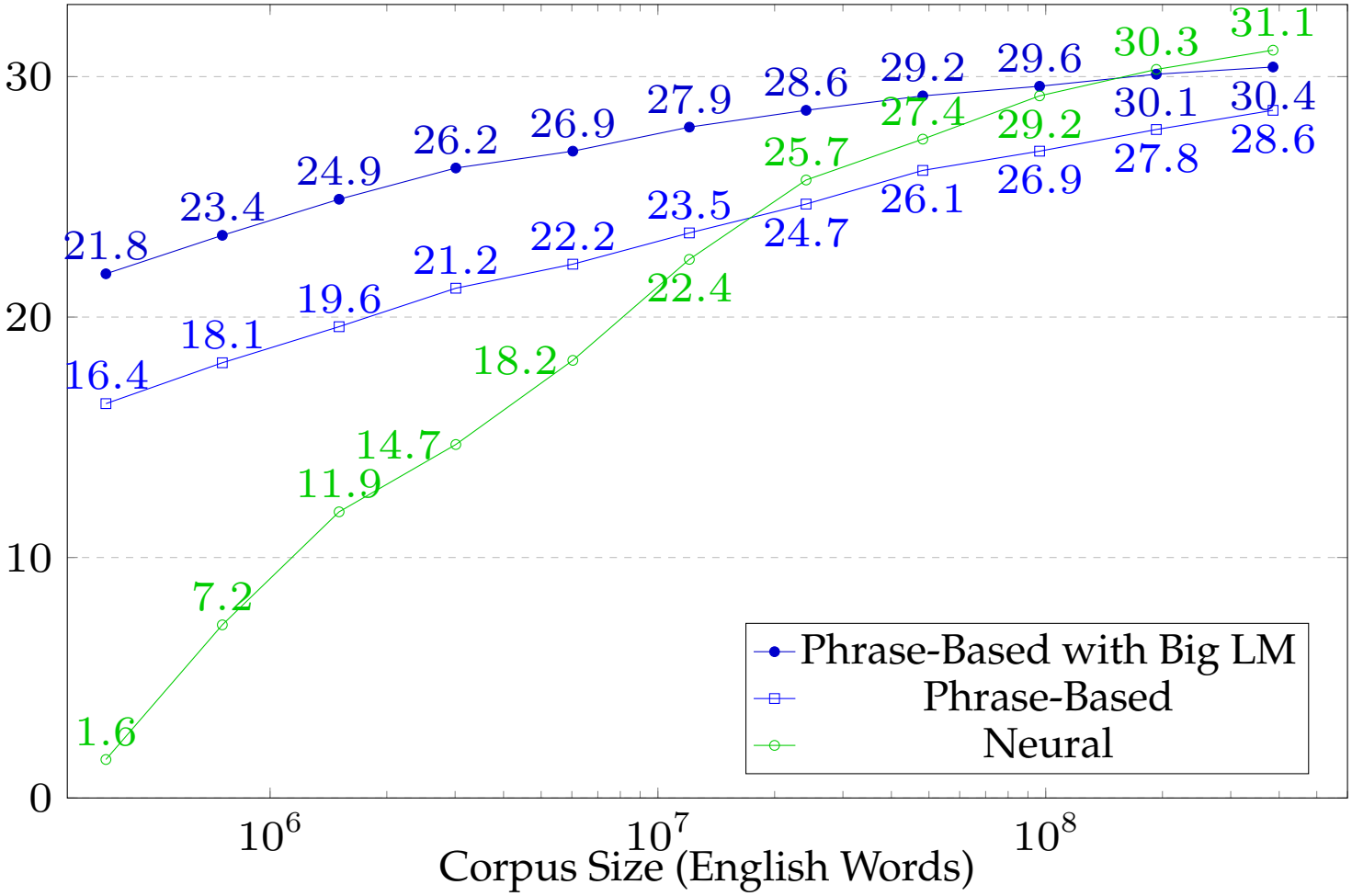
(in 2017 barely any statistical machine translation submissions)

Today's Agenda

- Challenges
  - lack of training data
  - domain mismatch
  - noisy data
  - sentence length
  - word alignment
  - beam search
- Alternative architectures
  - convolutional neural networks
  - self-attention

challenges

Amount of Training Data



English-Spanish systems trained on 0.4 million to 385.7 million words

Translation Examples

Source	A Republican strategy to counter the re-election of Obama
$\frac{1}{1024}$	Un órgano de coordinación para el anuncio de libre determinación
$\frac{1}{512}$	Lista de una estrategia para luchar contra la elección de hojas de Ohio
$\frac{1}{256}$	Explosión realiza una estrategia divisiva de luchar contra las elecciones de autor
$\frac{1}{128}$	Una estrategia republicana para la eliminación de la reelección de Obama
$\frac{1}{64}$	Estrategia siria para contrarrestar la reelección del Obama .
$\frac{1}{32}+$	Una estrategia republicana para contrarrestar la reelección de Obama

domain mismatch

Domain Mismatch

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

Translation Examples

Source	Schaue um dich herum.
Ref.	Look around you.
All	NMT: Look around you. SMT: Look around you.
Law	NMT: Sughum gravecorn. SMT: In order to implement dich Schaue .
Medical	NMT: EMEA / MB / 049 / 01-EN-Final Work progamme for 2002 SMT: Schaue by dich around .
IT	NMT: Switches to paused. SMT: To Schaue by itself . \t \t
Koran	NMT: Take heed of your own souls. SMT: And you see.
Subtitles	NMT: Look around you. SMT: Look around you .

noisy data

Noise in Training Data

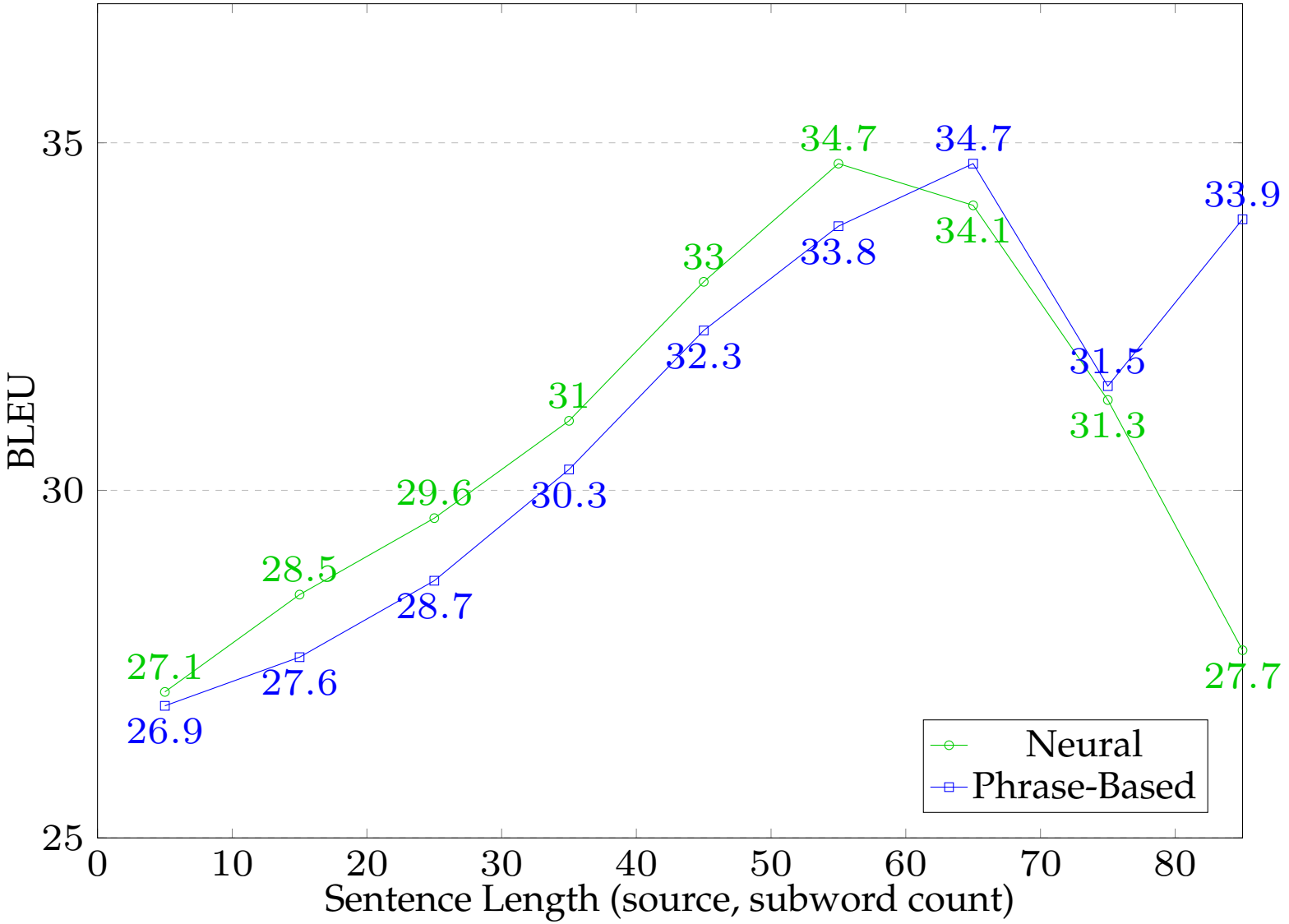
- Chen et al. [2016] add noise to WMT EN-FR training data
  - artificial noise: permute order of target sentences
  - conclusion: NMT is more sensitive to (some types of) noise than SMT

Noise	0%	10%	20%	50%
SMT	32.7	32.7 ($\pm$0.0)	32.6 (–0.1)	32.0 (–0.7)
NMT	35.4 (–0.1)	34.8 (–0.6)	32.1 (–3.3)	30.1 (–5.3)

- Other kind of noise: non-text, text in wrong languages

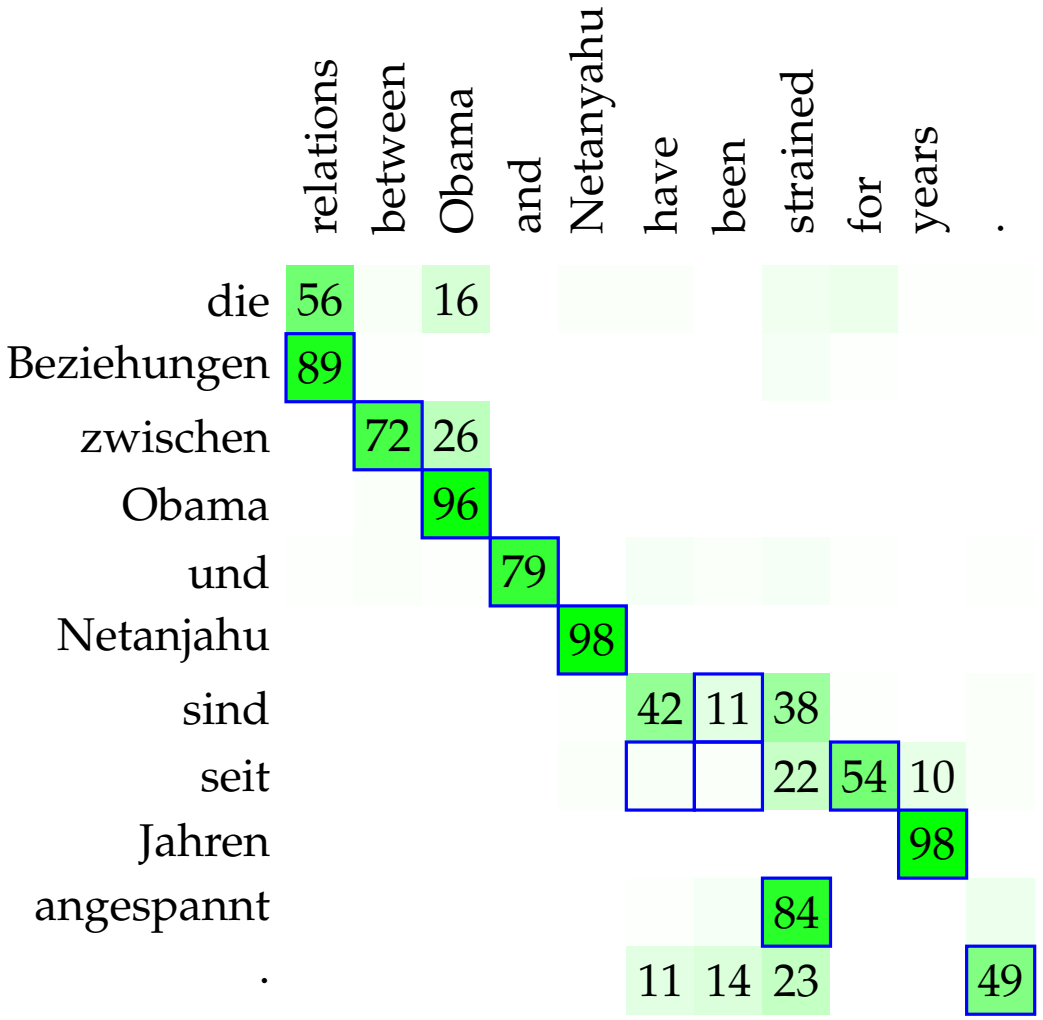
sentence length

Sentence Length

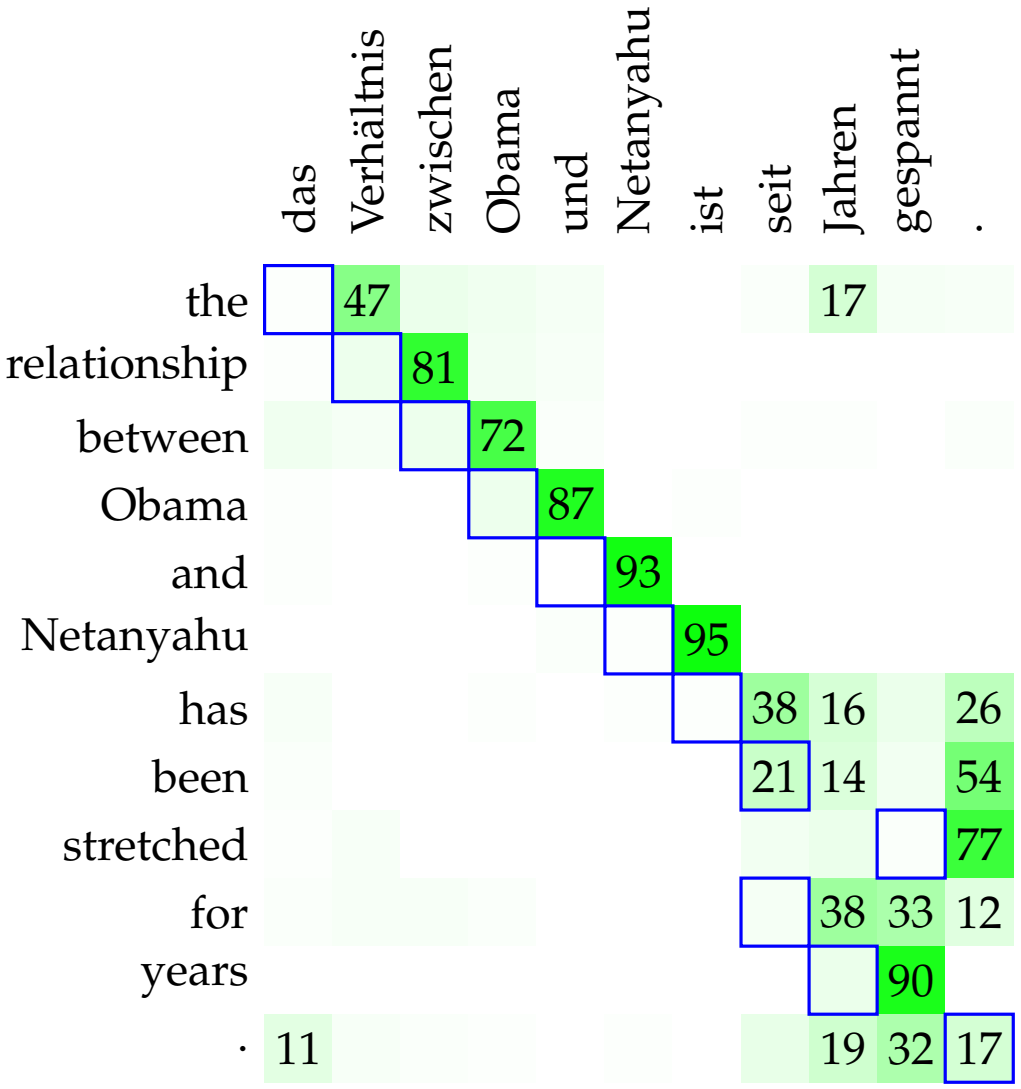


word alignment

Word Alignment

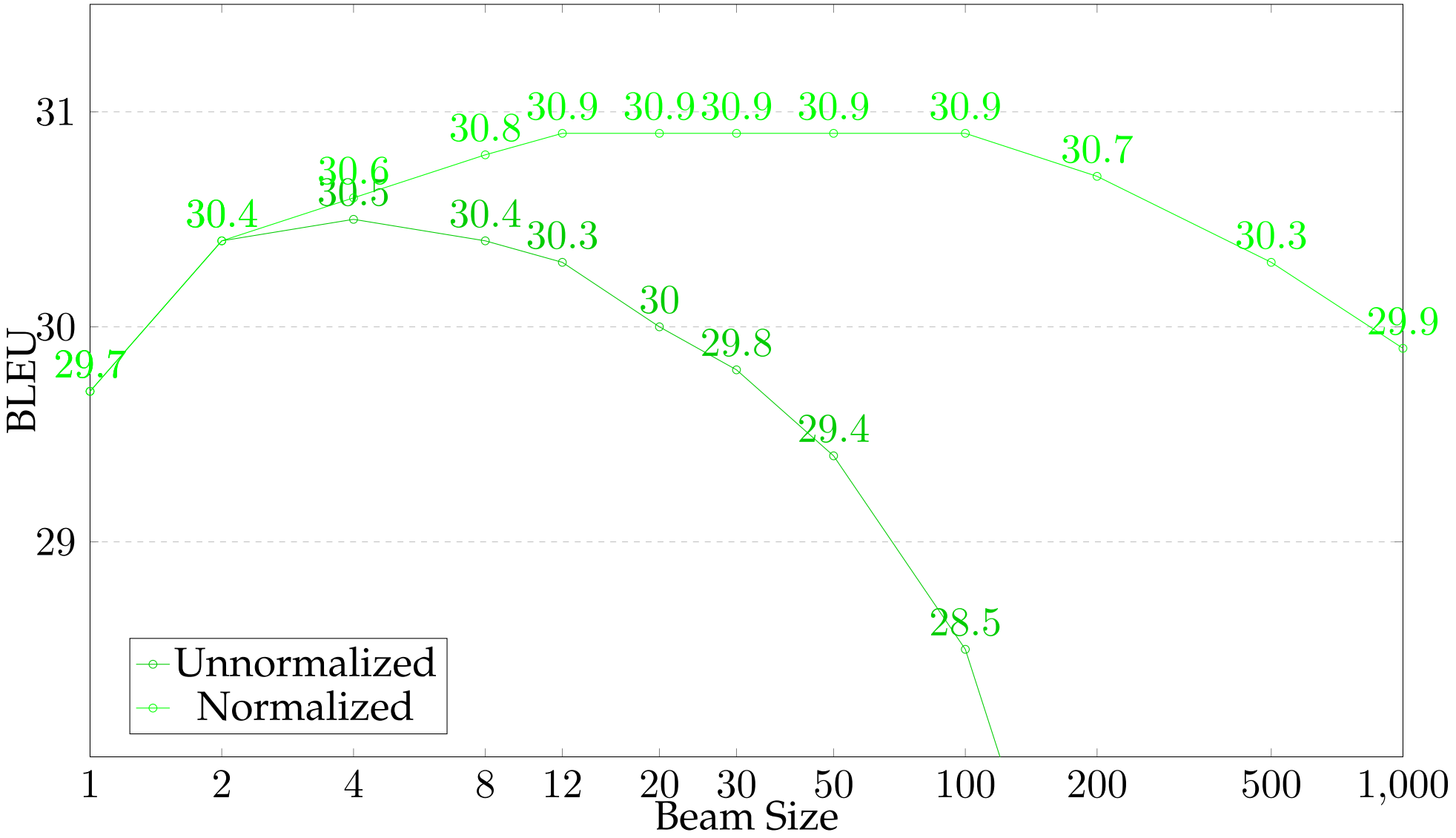


Word Alignment?

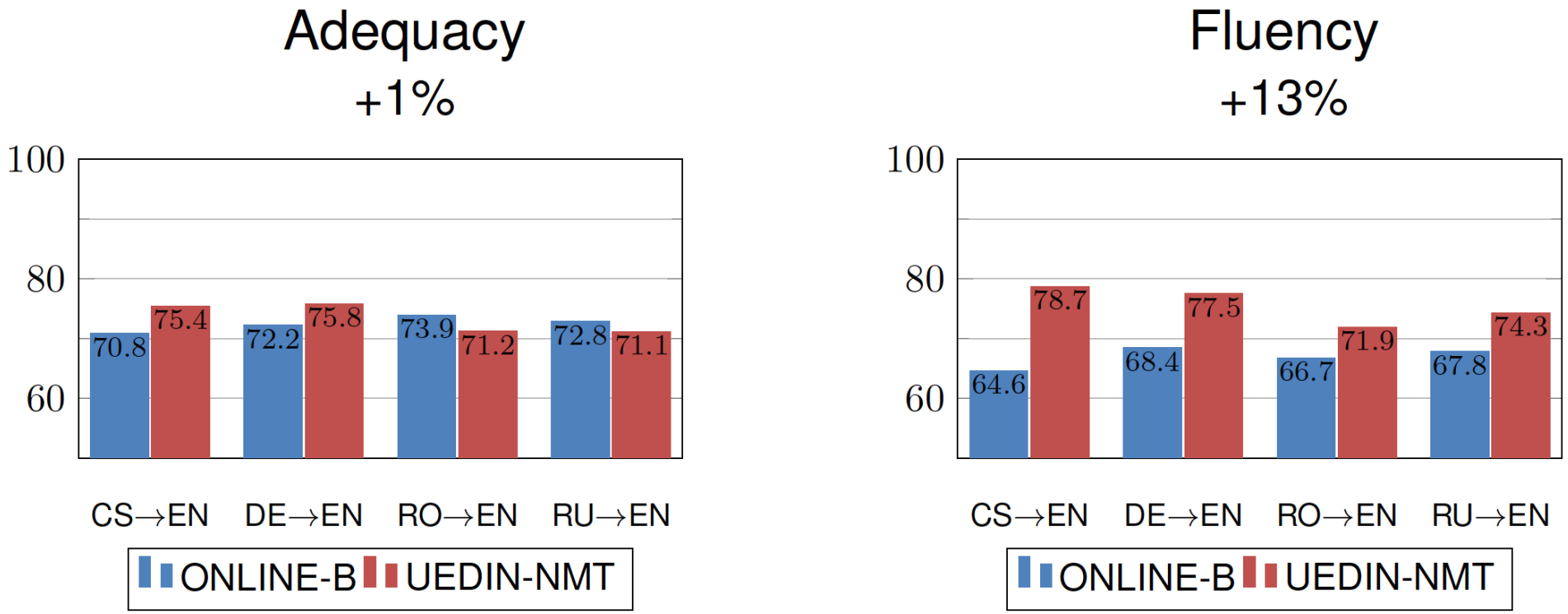


beam search

Beam Search


31
30
29
BLEU
30.4
30.6
30.8
30.9
30.9
30.9
30.9
30.9
30.7
30.3
29.9
29.7
30.5
30.4
30.3
30
29.8
29.4
28.5
Unnormalized
Normalized
1
2
4
8
12
20
30
50
100
200
500
1,000
Beam Size

Just Better Fluency?



(from: Sennrich and Haddow, 2017)

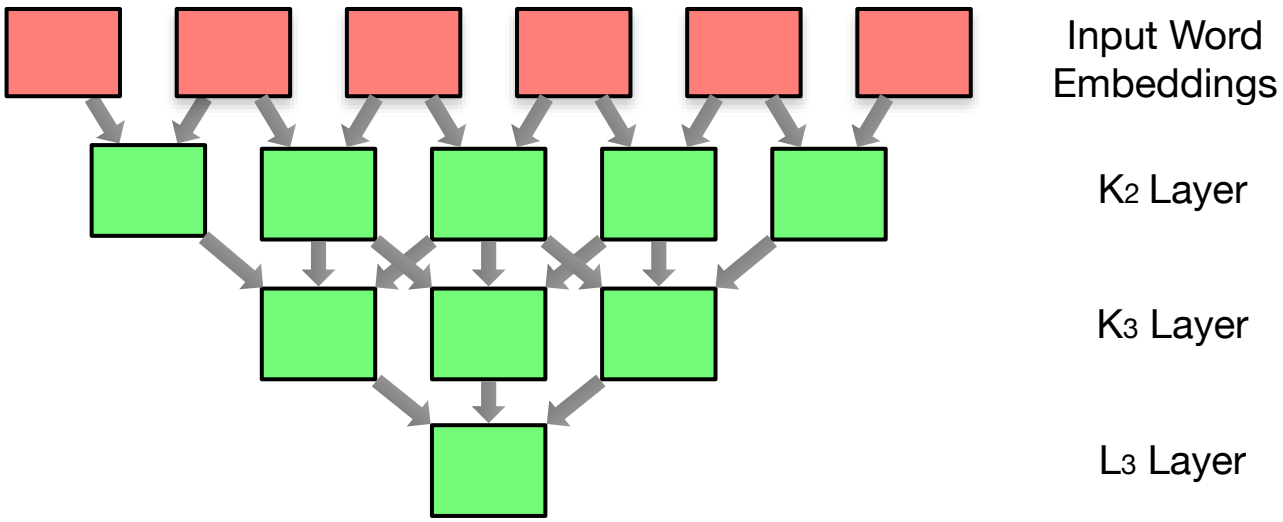
alternative architectures

Beyond Recurrent Neural Networks

- We presented the currently dominant model
  - recurrent neural networks for encoder and decoder
  - attention
- Convolutional neural networks
- Self attention

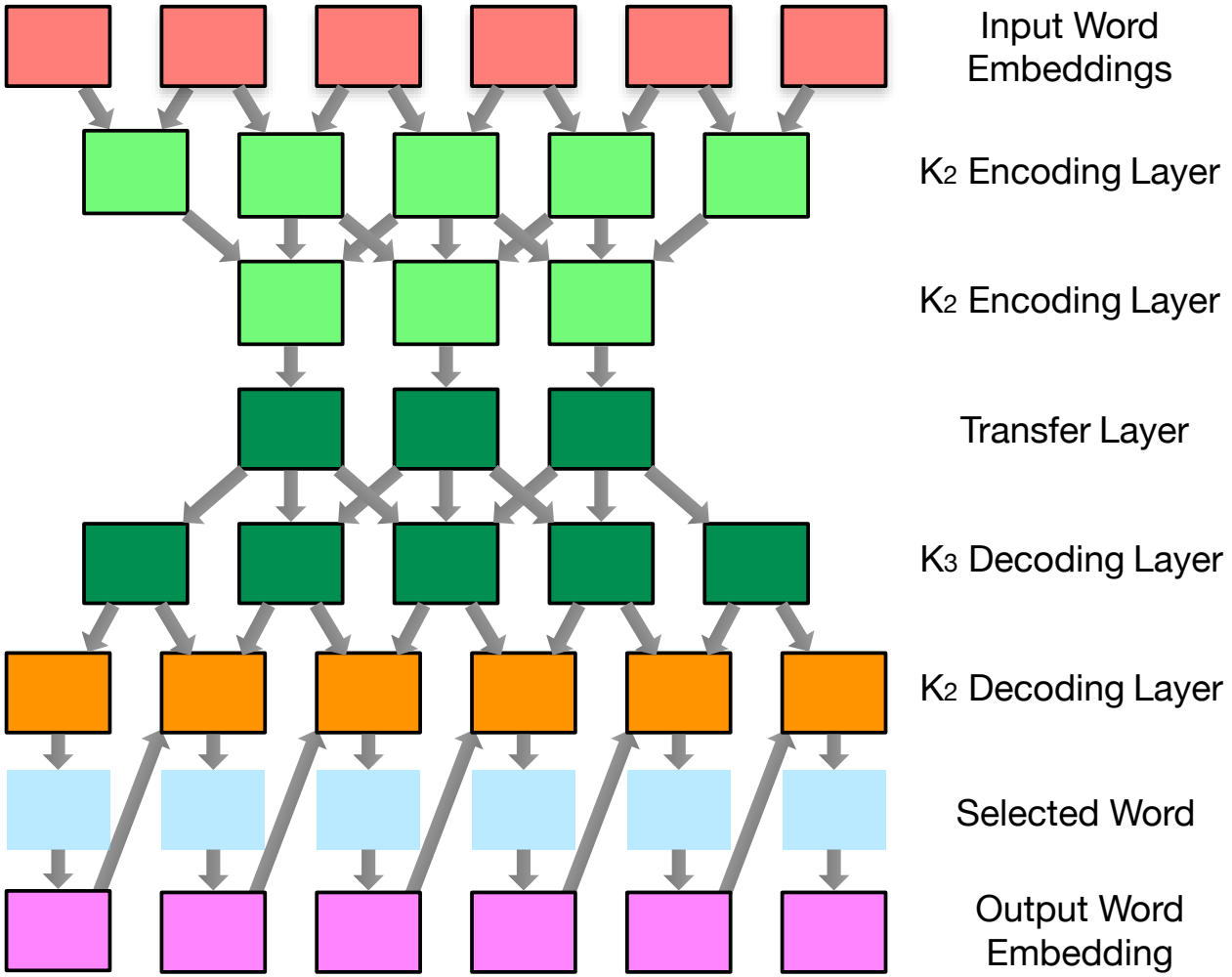
convolutional neural networks

Convolutional Neural Networks



- Build sentence representation bottom-up
  - merge any n neighboring nodes
  - n may be 2, 3, ...

Generation


Input Word Embeddings
K2 Encoding Layer
K2 Encoding Layer
Transfer Layer
K3 Decoding Layer
K2 Decoding Layer
Selected Word
Output Word Embedding

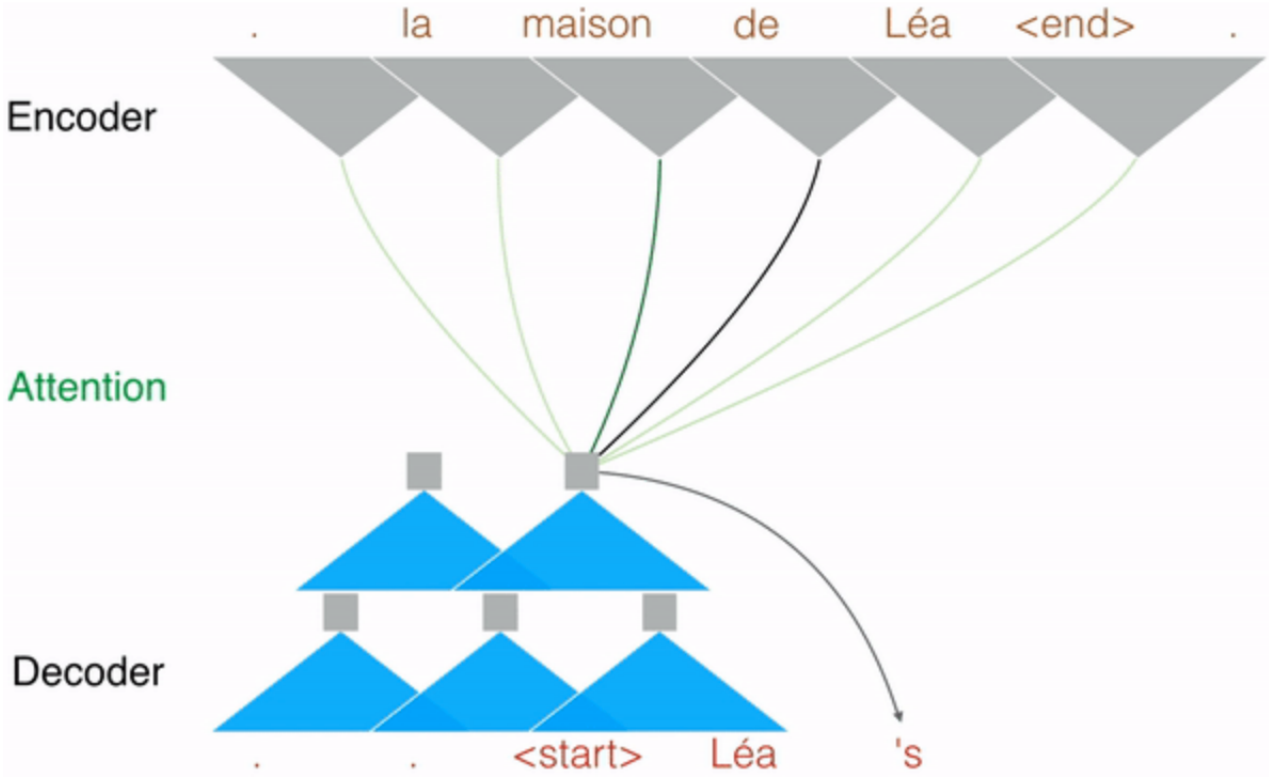
Generation

- Encode with convolutional neural network
- Decode with convolutional neural network
- Also include a linear recurrent neural network

- Important: predict length of output sentence

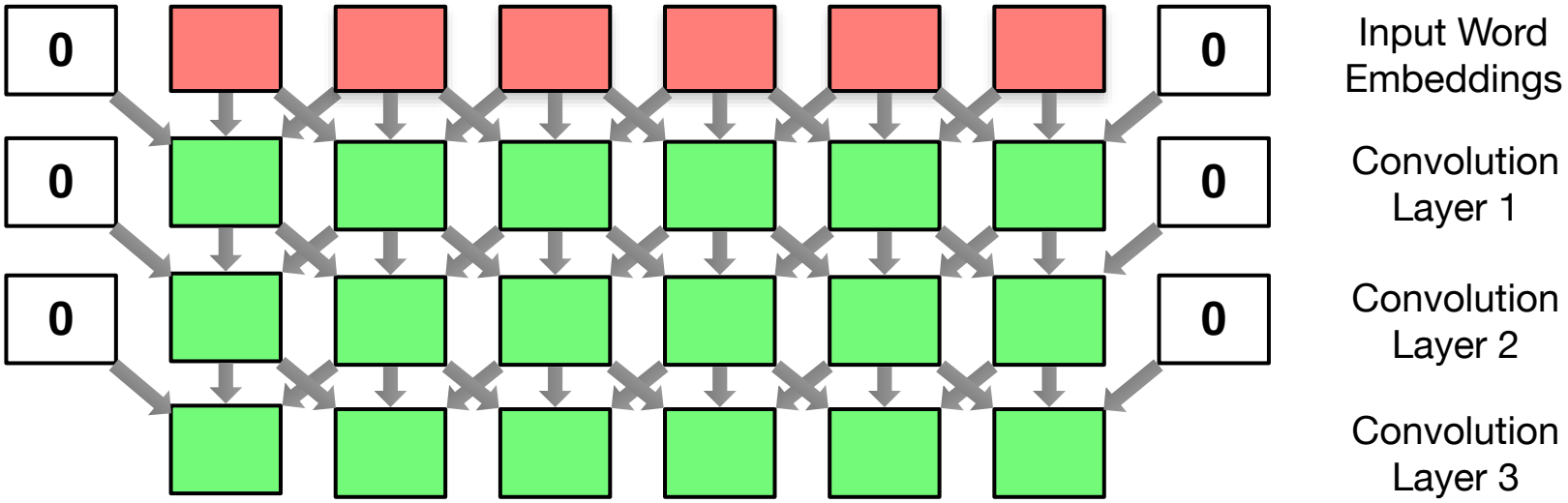
- Does it work?
  used successfully in re-ranking (Cho et al., 2014)

Convolutional Network with Attention



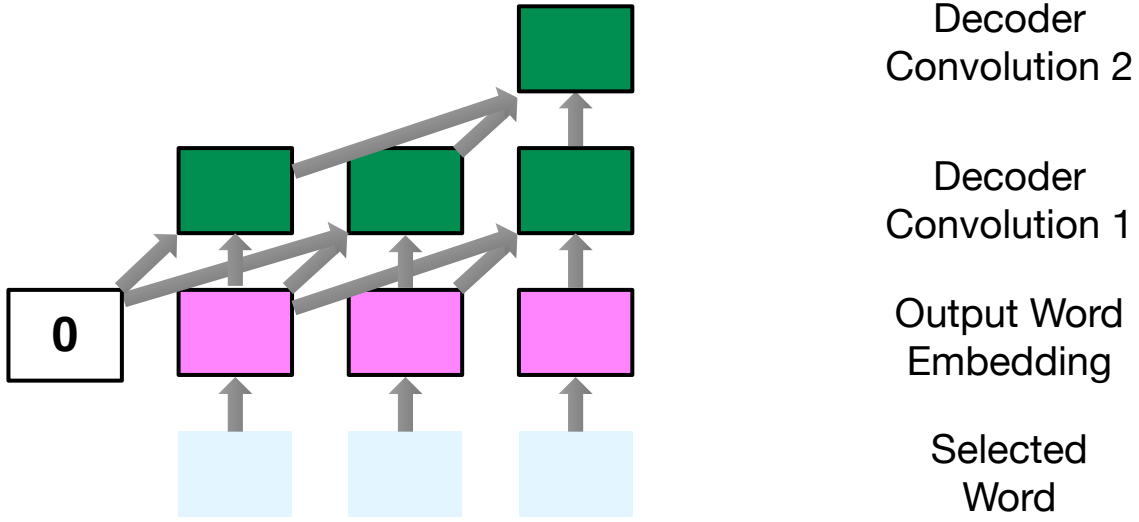
(Facebook, 2017)

Convolutional Encoder



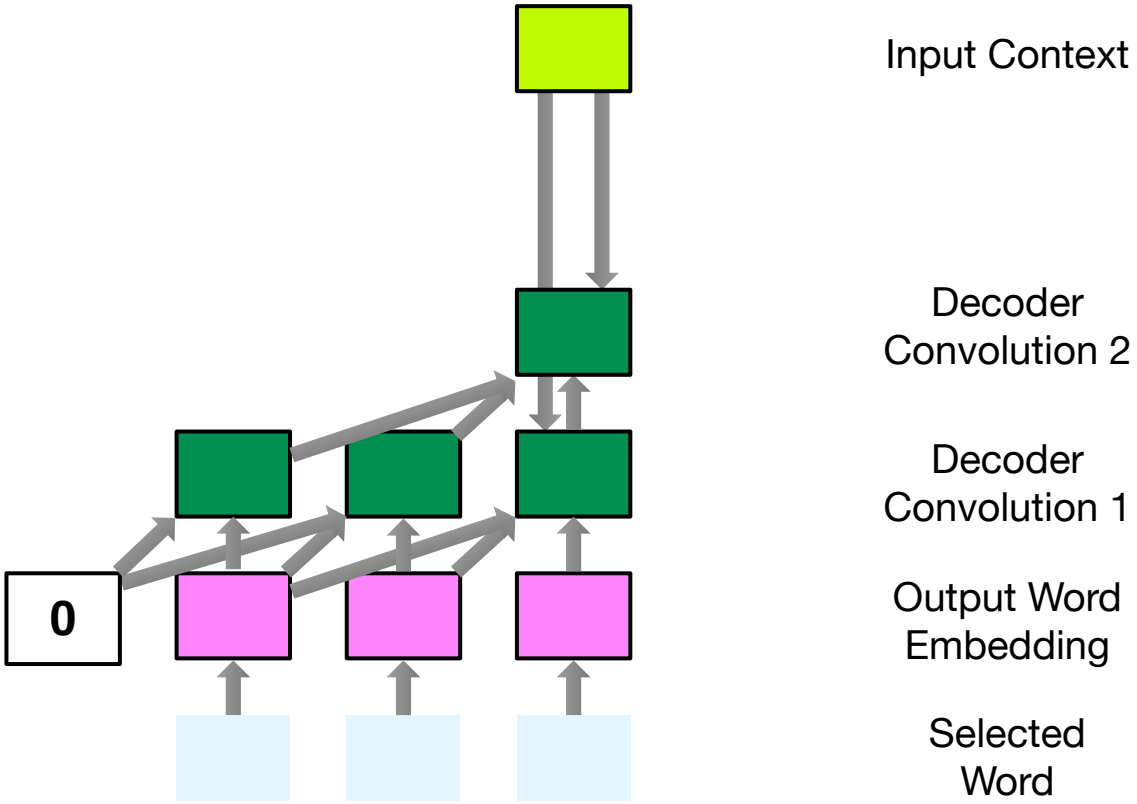
- Similar idea as deep recurrent neural networks
- Good: more parallelizable
- Bad: less context when refining representation of a word

Convolutional Decoder



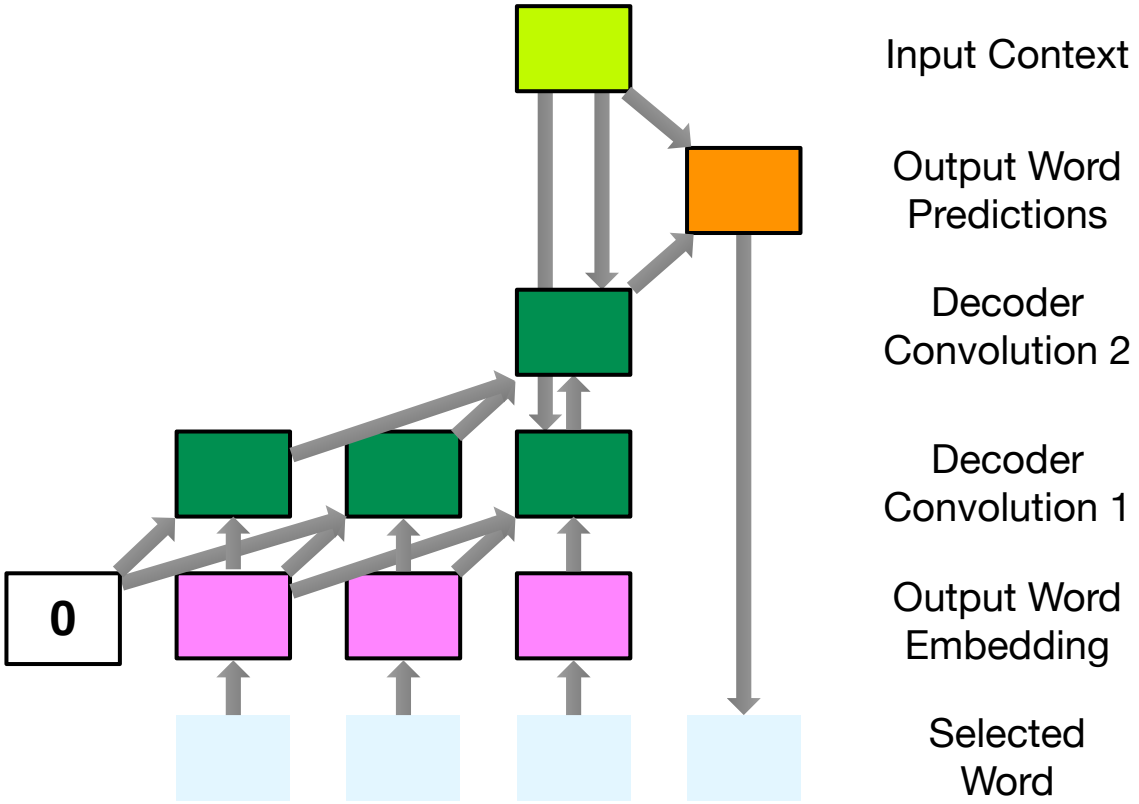
- Convolutions over output words
- Only previously produced output words (still left-to-right decoding)

Convolutional Decoder



- Inclusion of Input context
- Context result of attention mechanism (similar to previous)

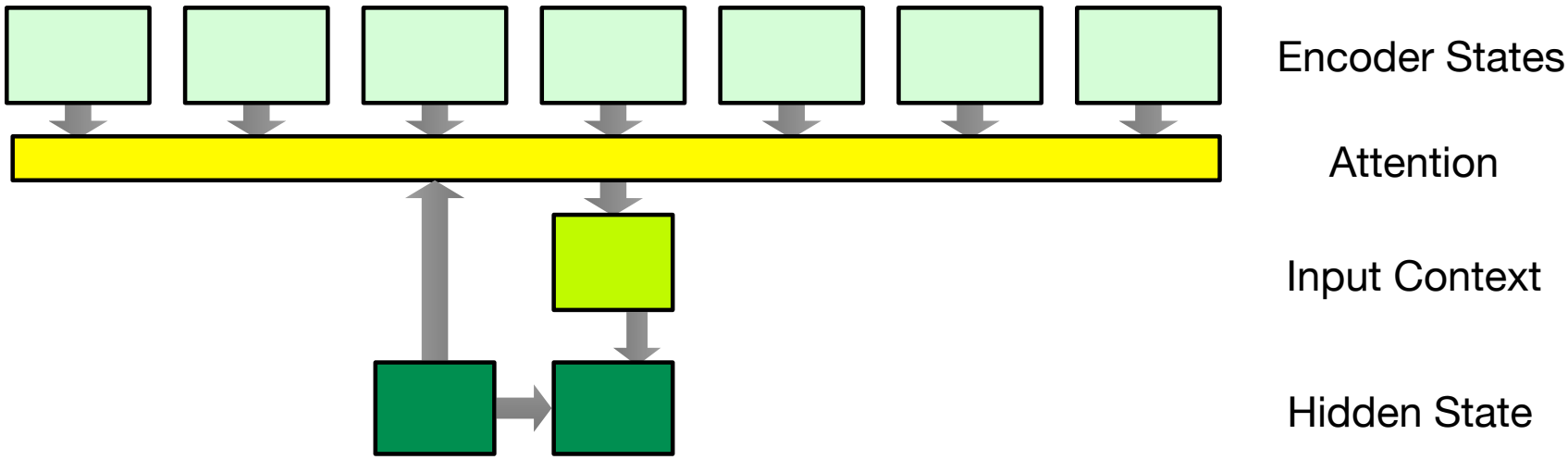
Convolutional Decoder



- Predict output word distribution
- Select output word

self-attention

Attention



- Compute association between last hidden state and encoder states

Attention Math

- Input word representation h_k
- Decoder state s_j
- Computations

$$a_{jk} = \frac{1}{|h|} s_j h_k^T \quad \text{raw association}$$

$$\alpha_{jk} = \frac{\exp(a_{jk})}{\sum_\kappa \exp(a_{j\kappa})} \quad \text{normalized association (softmax)}$$

$$\text{self-attention}(h_j) = \sum_k \alpha_{j\kappa} h_k \quad \text{weighted sum}$$

Self-Attention

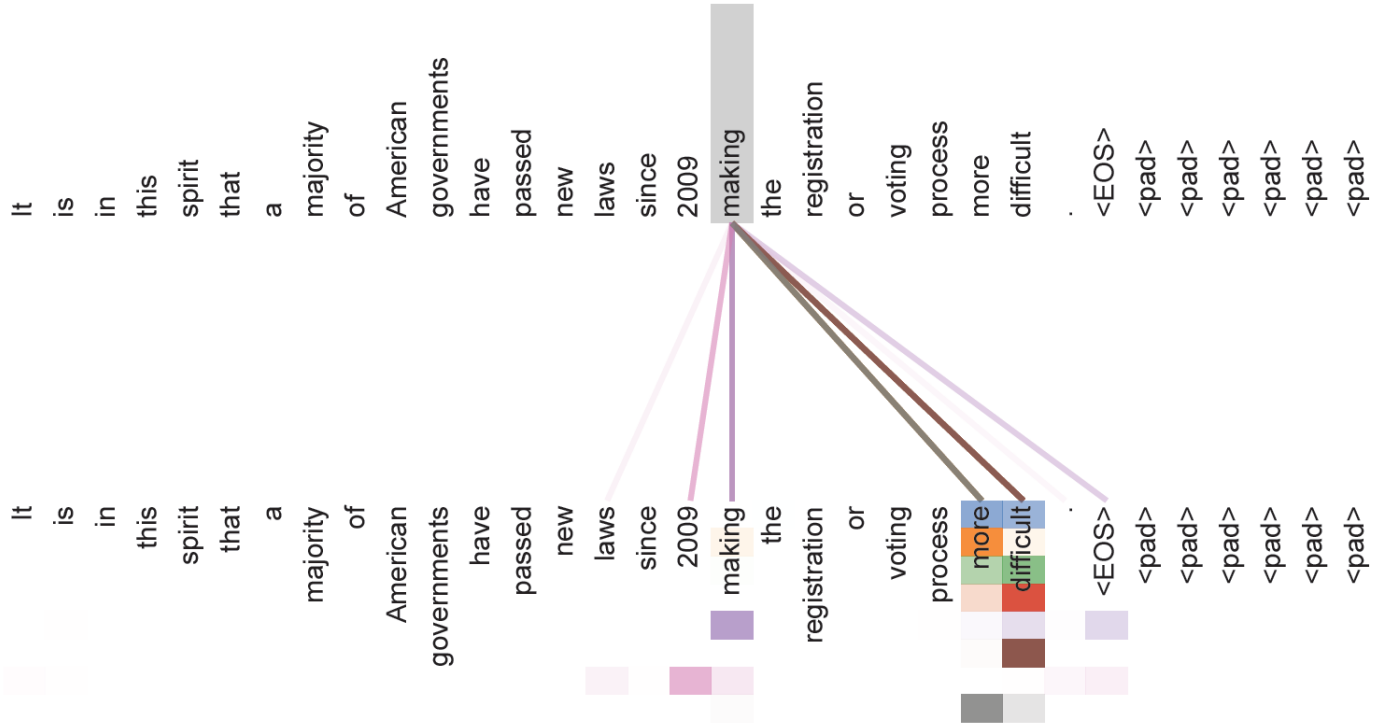
- Attention

$$a_{jk} = \frac{1}{|h|} s_j h_k^T$$

- Self-attention

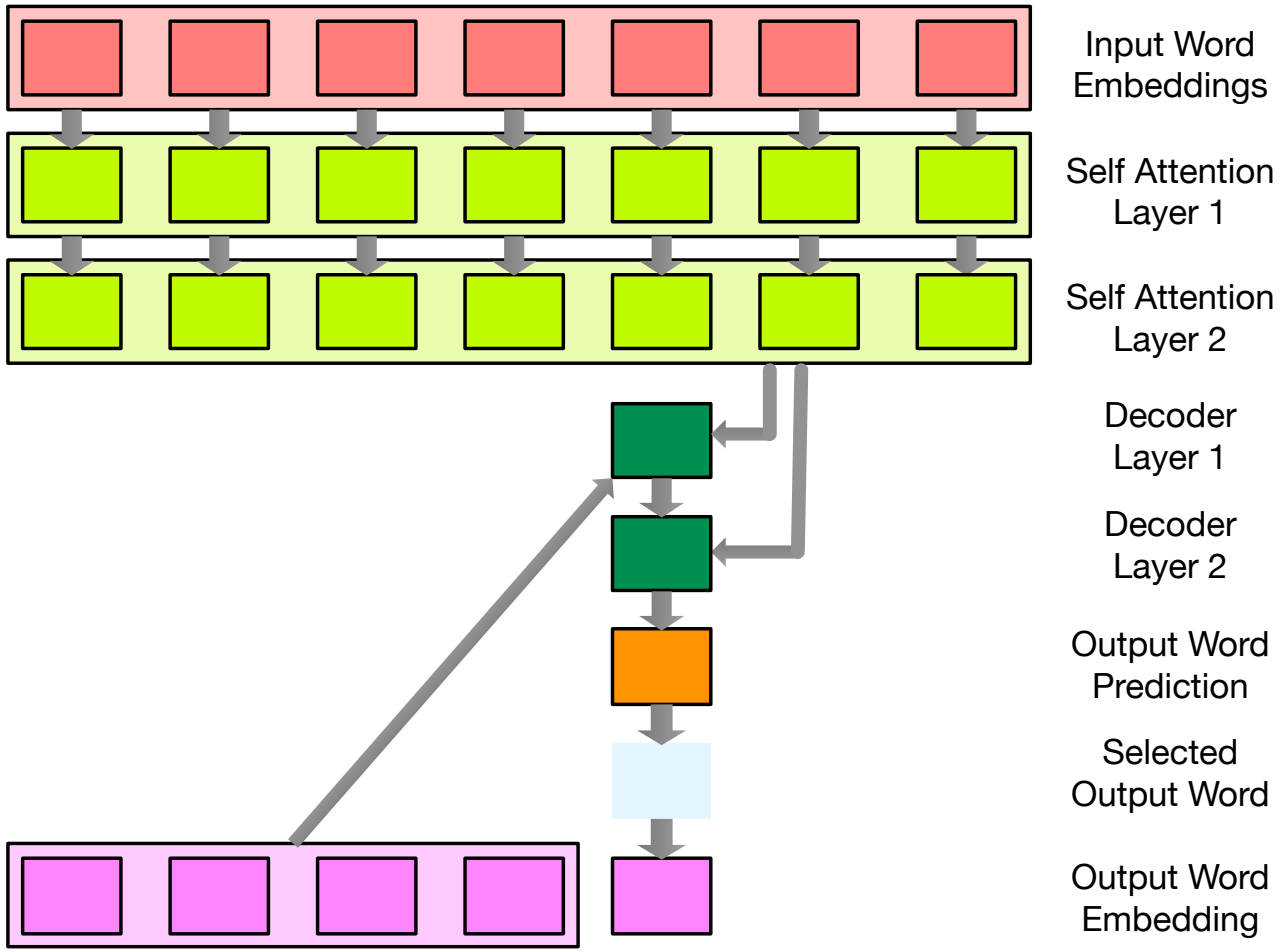
$$a_{jk} = \frac{1}{|h|} h_j h_k^T$$

Why?



- Refine representation of word with related words
  making ... more difficult refines *making*
- Good: more parallelizable than recurrent neural network
- Good: wide context when refining representation of a word

Stacked Attention in Decoder



Where Are We Now?

- Recurrent neural network with attention currently dominant model
- Still many challenges
- New proposals in Spring 2017
  - convolutions (Facebook)
  - self-attention (Google)
- Too early to tell if either becomes the new paradigm
- Open source implementations are available

questions?