Deep learning for Natural Language Processing

Eneko Agirre, Gorka Azkune
Ander Barrena, Oier Lopez de Lacalle
@eagirre @gazkune @4nderB @oierldl #dl4nlp
http://ixa2.si.ehu.eus/eneko/dl4nlp

Sess. 5: Attention, transformers and Natural language inference

Plan for the course

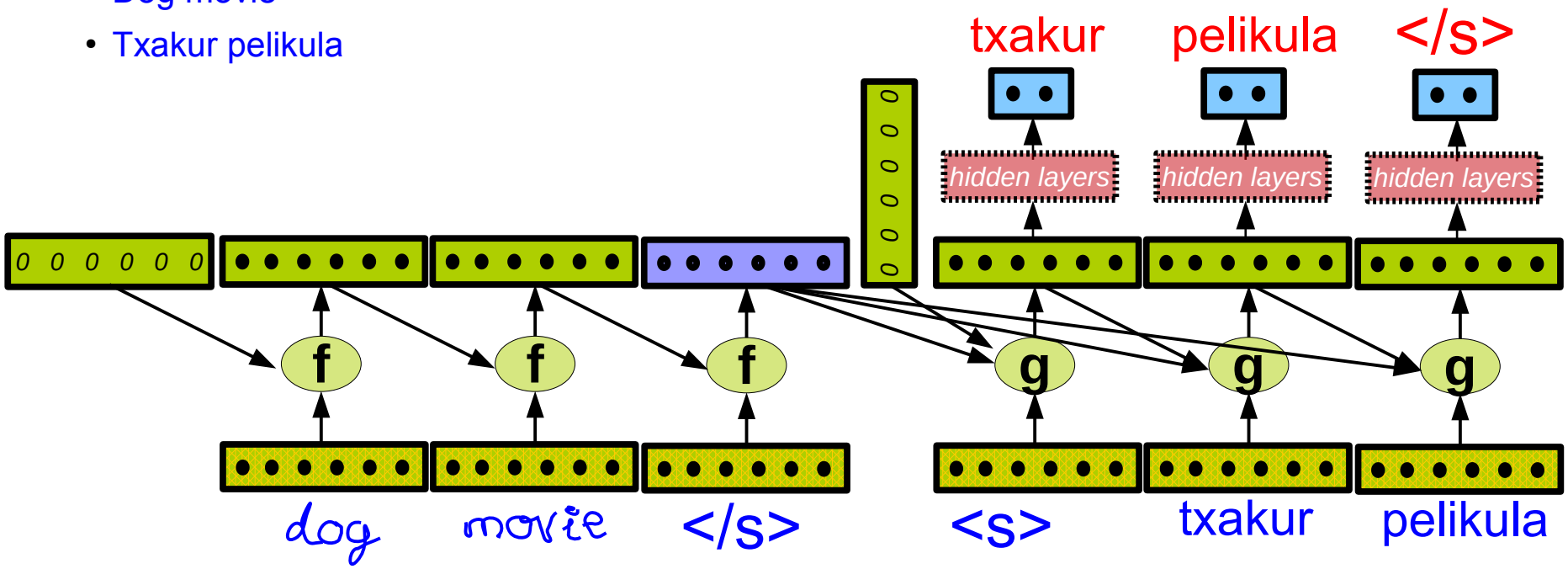
- Introduction: machine learning and NLP
- Multilayer perceptron
- Word representation and
  Recurrent neural networks (RNN)
- Sequence-to-Sequence (seq2seq) and
  Machine Translation
- **Attention, transformers and**
  Natural language inference
- Pre-trained transformers, BERT, GPT
- Bridging the gap between natural languages
  and the visual world

Sequence to sequence for MT

- Combine two RLM: encoder and decoder
- **Train** as regular RLM

Input example: pair of translations:

- Dog movie
- Txakur pelikula

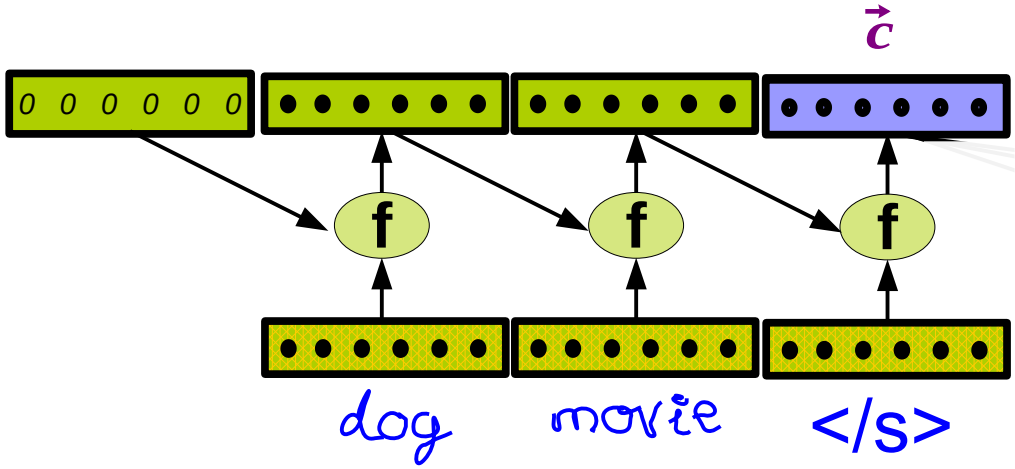


Sequence to sequence for MT

- Combine two RLM: encoder and decoder
- **Train** as regular RLM
- Note that decoder is conditioned on last hidden state from encoder:

$$\vec{h}_r^{encoder} = \tanh\left(W^f\left[\vec{h}_{r-1}^{encoder}, \vec{w}_s\right] + \vec{a}\right)$$

$$\vec{c} = \vec{h}_r^{encoder}$$



Sequence to sequence for MT

- Combine two RLM: encoder and decoder
- **Train** as regular RLM
- Note that decoder is conditioned on last hidden state from encoder:

$$\vec{h}_r^{encoder} = \tanh\left(W^f[\vec{h}_{r-1}^{encoder}, \vec{w}_s] + \vec{a}\right)$$

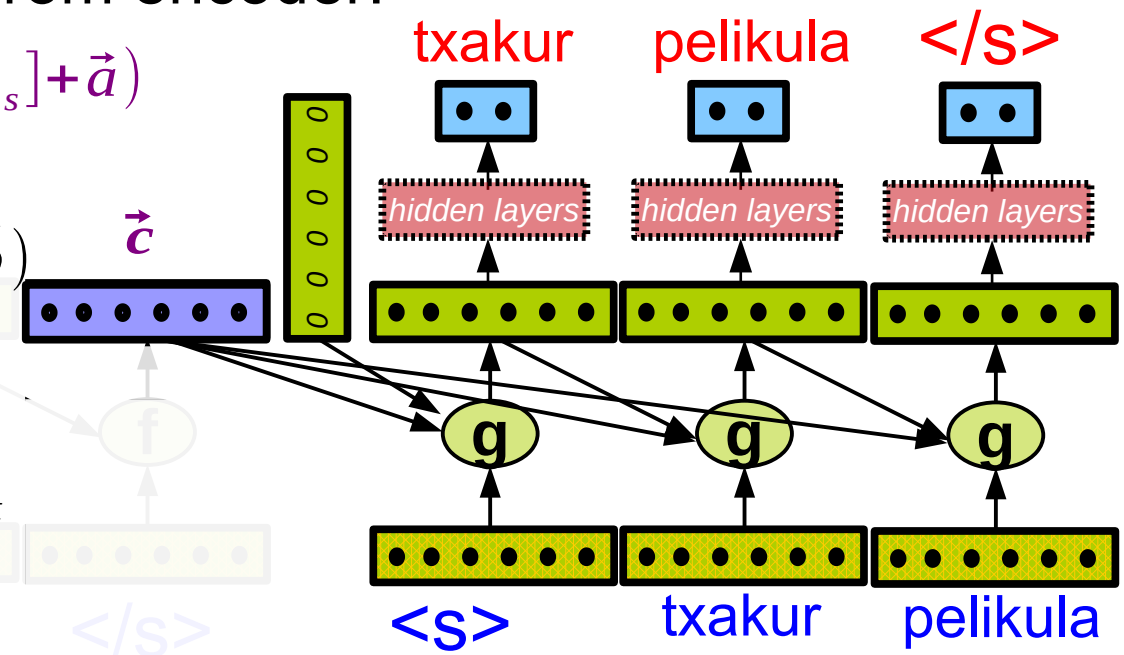
$$\vec{c} = \vec{h}_r^{encoder}$$

$$\vec{h}_t = \tanh\left(W^g[\vec{h}_{t-1}, \vec{w}_t, \vec{c}] + \vec{b}\right)$$

$$\hat{y}_t = softmax\left(W^S \vec{h}_t + \vec{d}\right)$$

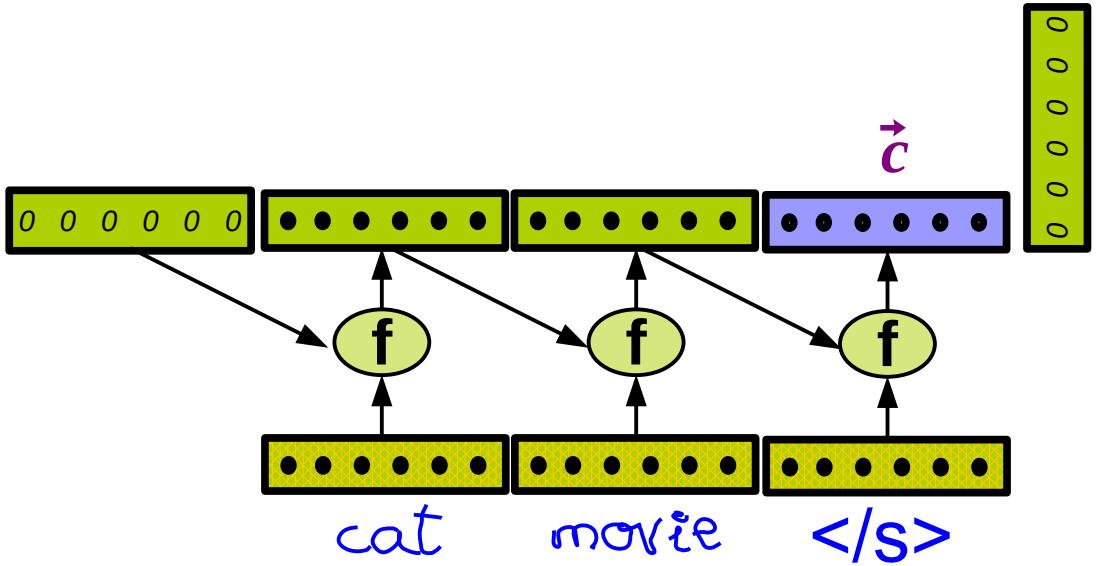
$$p\left(w_1, \ldots, w_m | \vec{c}\right) \approx \prod_{t=0}^{m-1} \hat{y}_{t,correct}$$

$$J_{\text{sentence}} = -\frac{1}{T}\sum_{t=1}^{m} \log \hat{y}_{t,correct}$$



Previously: seq2seq for MT

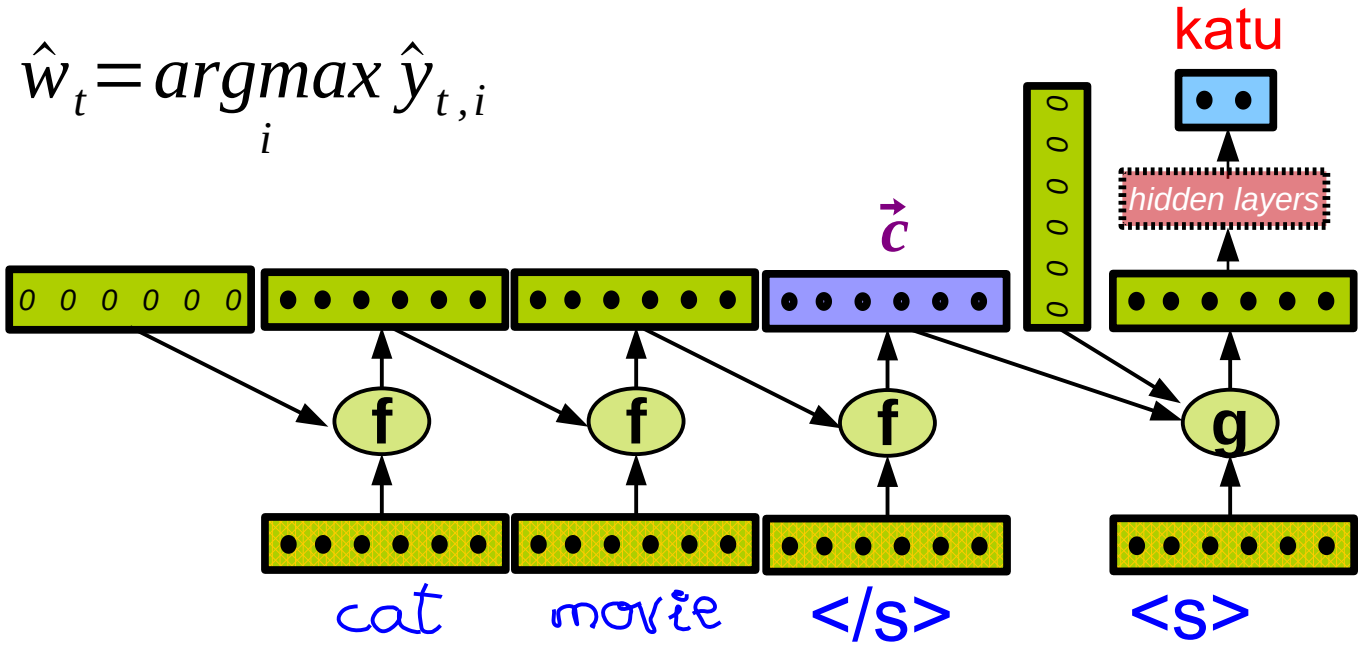
- Test as conditional RLM decoder
  - Compute sentence representation



Previously: seq2seq for MT

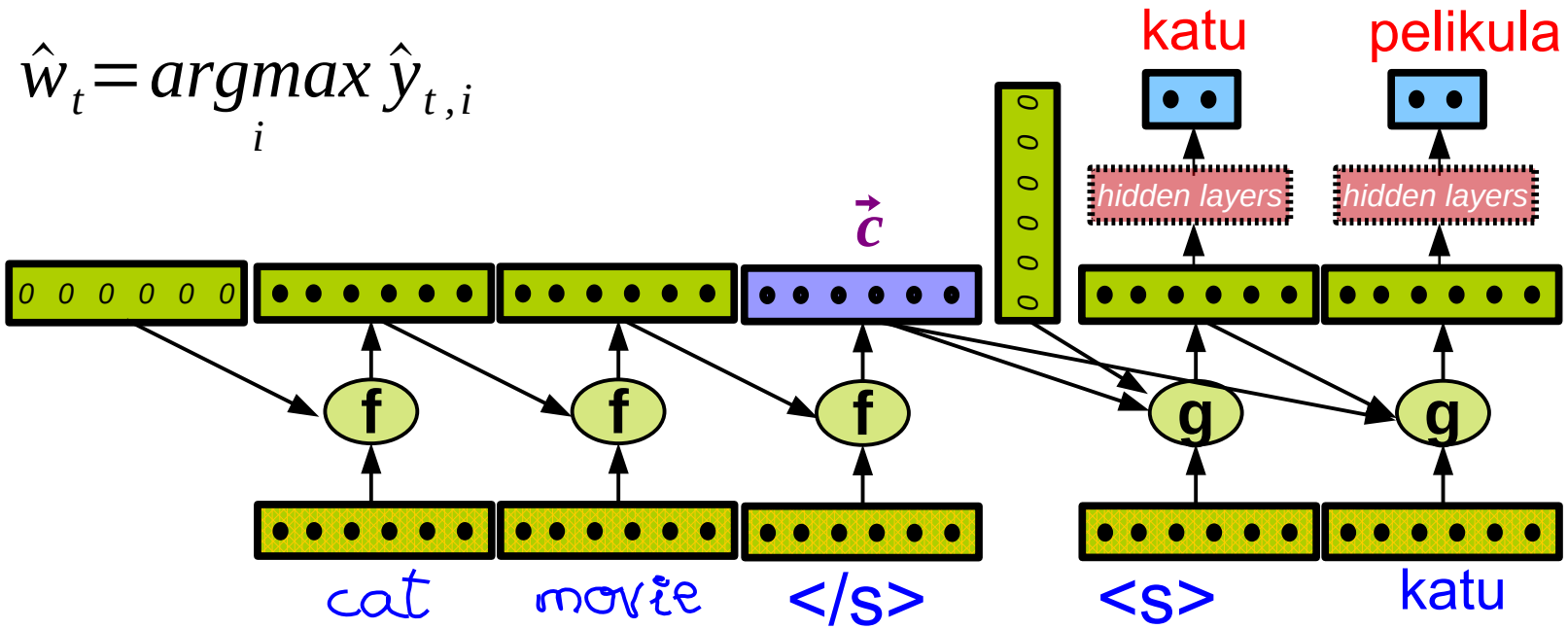
- Test as conditional RLM decoder
  - Compute sentence representation
  - Generate first word

$$\hat{w}_t = \underset{i}{argmax}\, \hat{y}_{t,i}$$



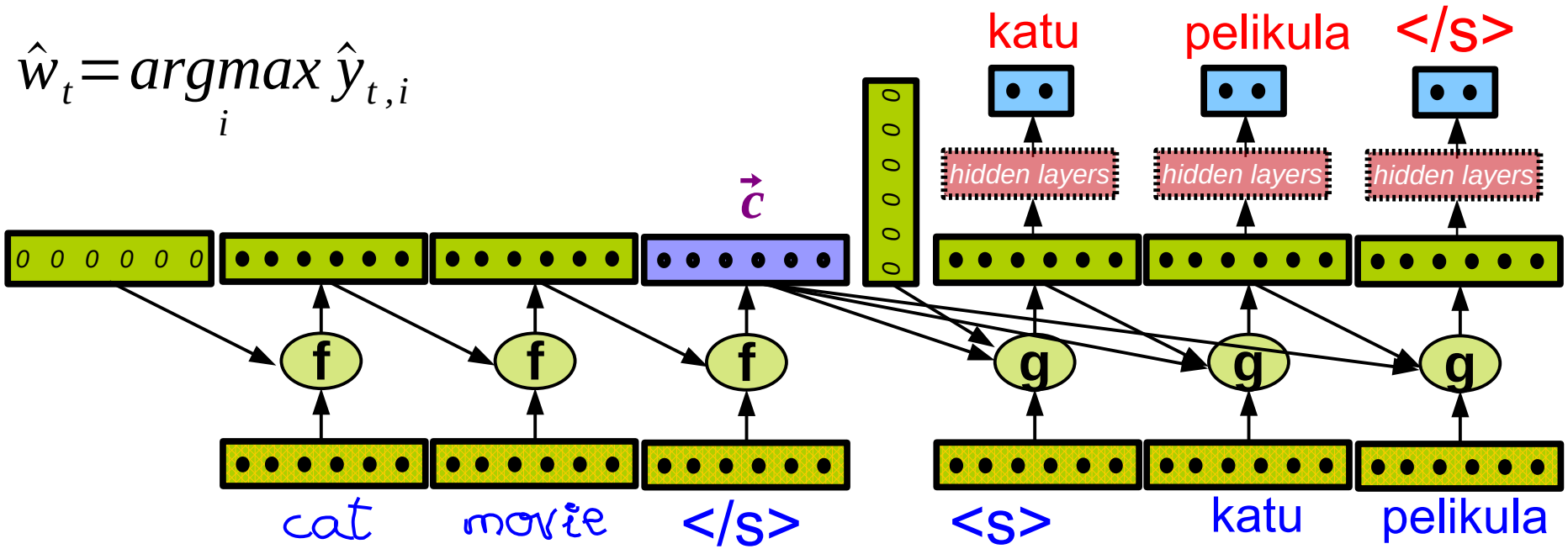
Previously: seq2seq for MT

- Test as conditional RLM decoder
  - Compute sentence representation
  - Generate second word



Previously: seq2seq for MT

- Test as conditional RLM decoder
  - Compute sentence representation
  - Generate next word, stop if </s>



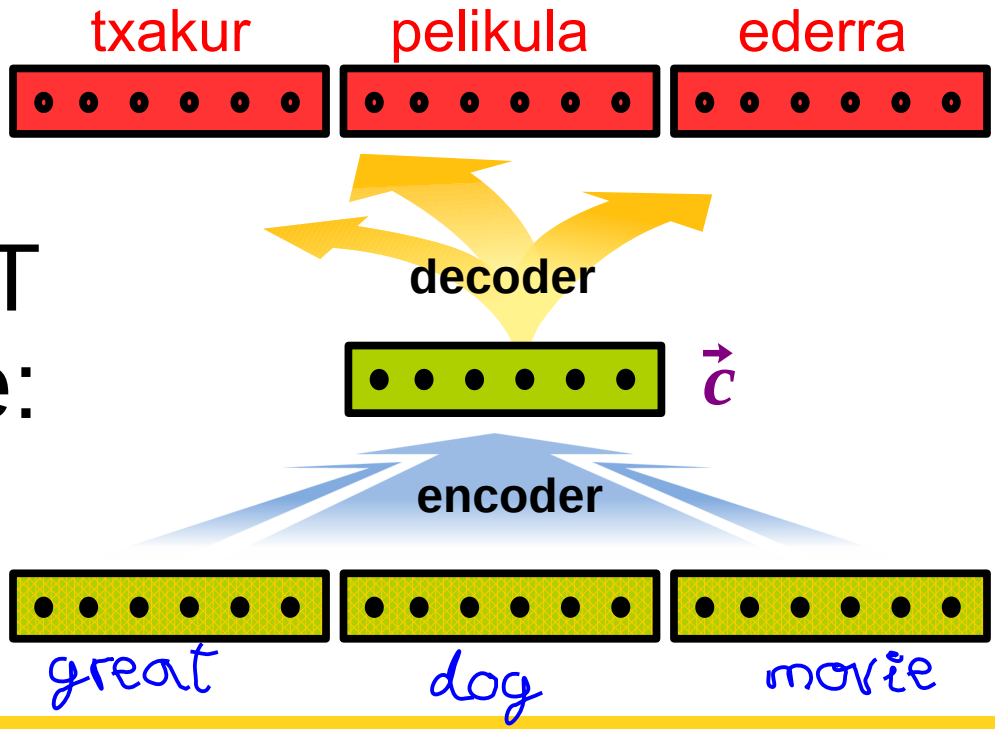
Plan for this session

- **Re-thinking seq2seq:**
  - Attention and memory
  - State of the art NMT: self-attention (transformers)
  - Amazing things:
    - Multilingual MT
    - MT without any bilingual data
- Evaluating sentence representations

Re-thinking seq2seq

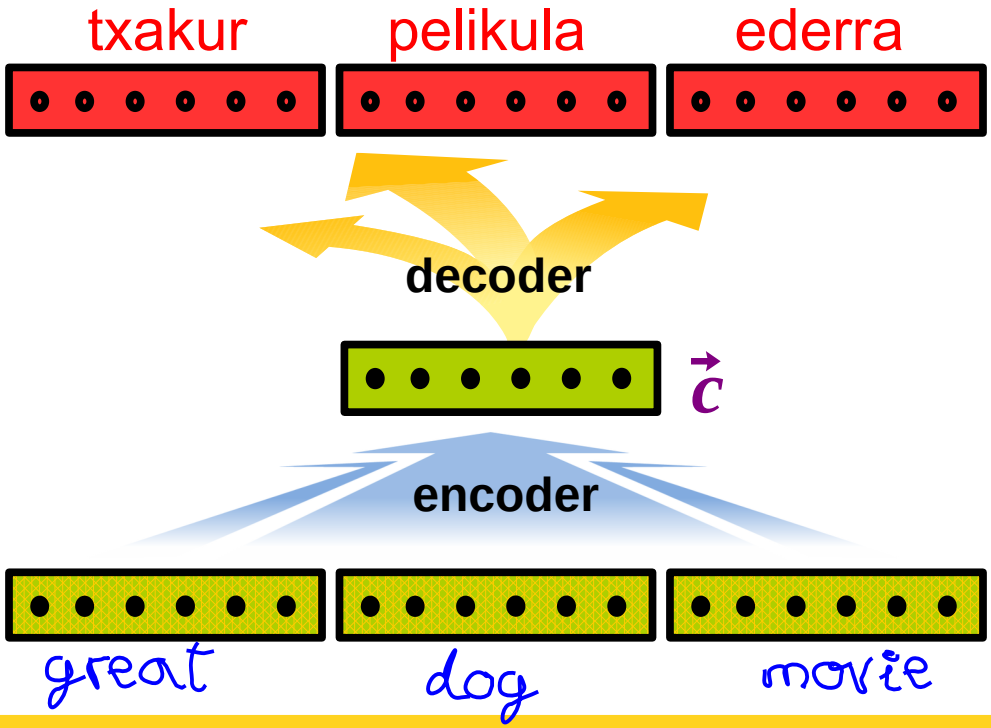
- Encode input sequence into vector $\vec{c}$
- Decode $\vec{c}$ into target sequence

The existing NMT proposal at the time:



Re-thinking seq2seq

"You can't cram the meaning or a whole %&!$# sentence into a single $&!#* vector!"
(Ray Mooney – Cho's talk at dl4mt workshop 2015)



Re-thinking seq2seq for NMT

Inspiration from human translators

- Summarize what has been translated so far
- Find the next thing to be translated
- Write the next target symbol
- Iterate

Translating “Esa película me gustó” into Basque:

> ?

Esa película me gustó

> Pelikula ?

Esa pelicula me gustó

> Pelikula hori ?

Esa película me gustó

> Pelikula hori gustatu ?

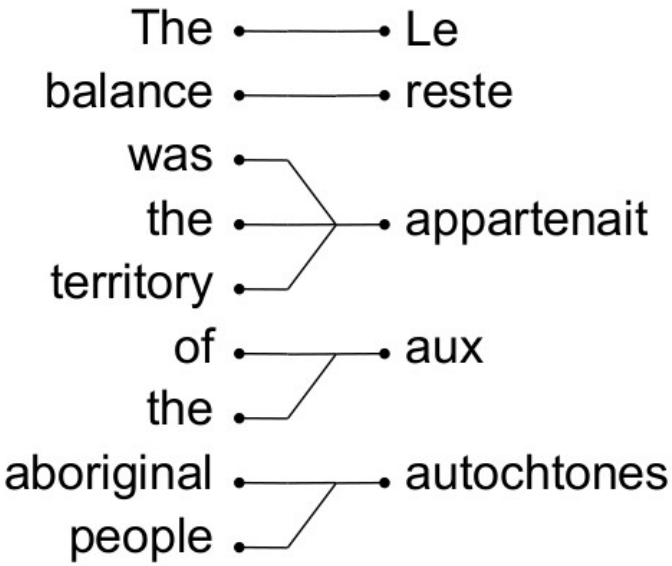
Esa película me gustó

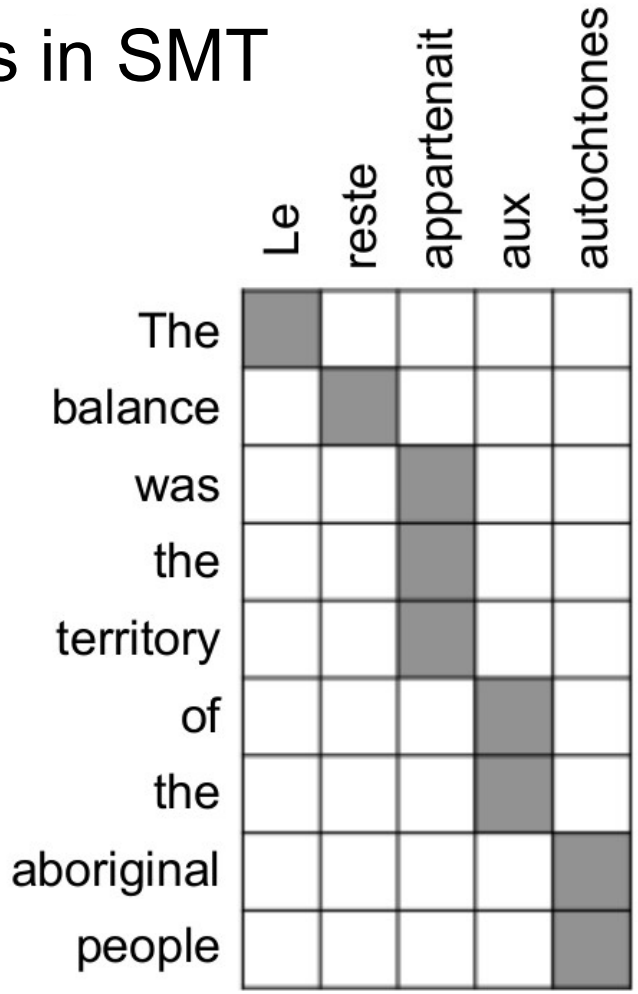
> Pelikula hori gustatu zitzaidan

Source: https://modela.eus/es/traductor

Re-thinking seq2seq for NMT

Inspiration from alignment models in SMT

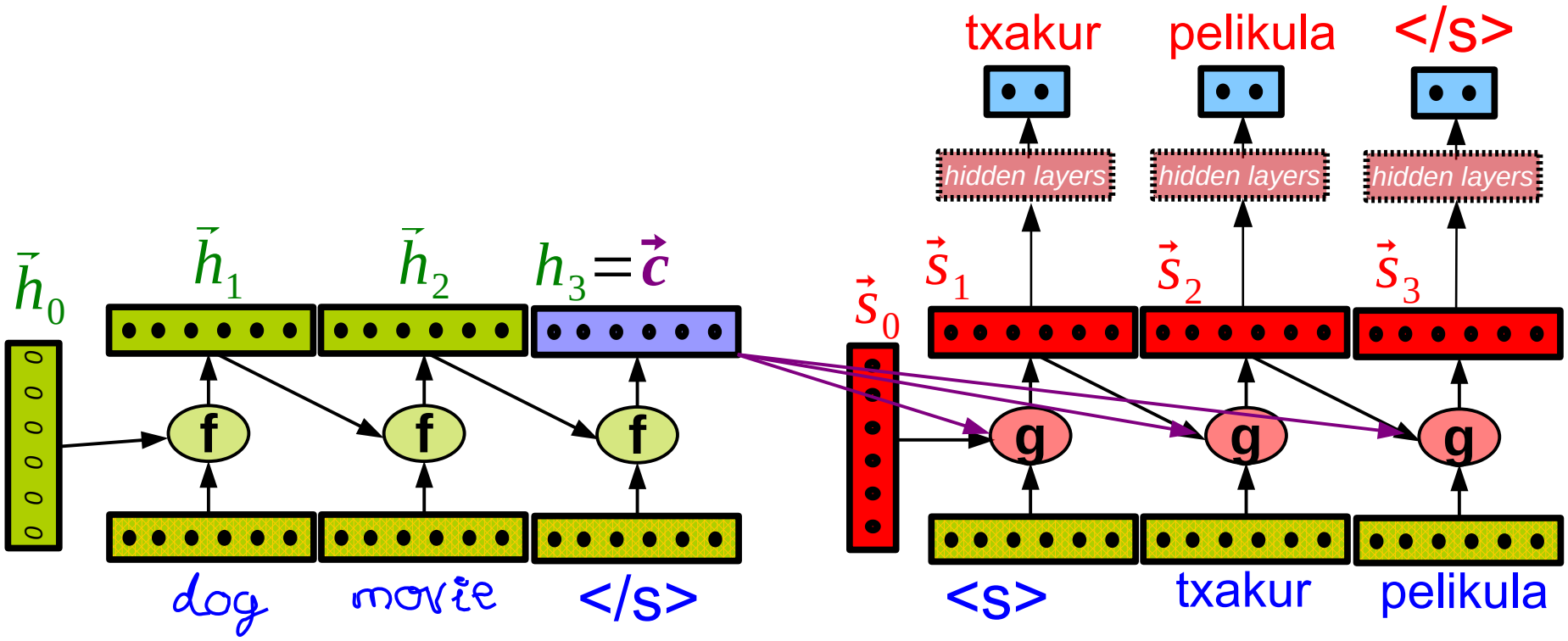




Source: cs224n lecture 10 Chris Manning, Richard Socher

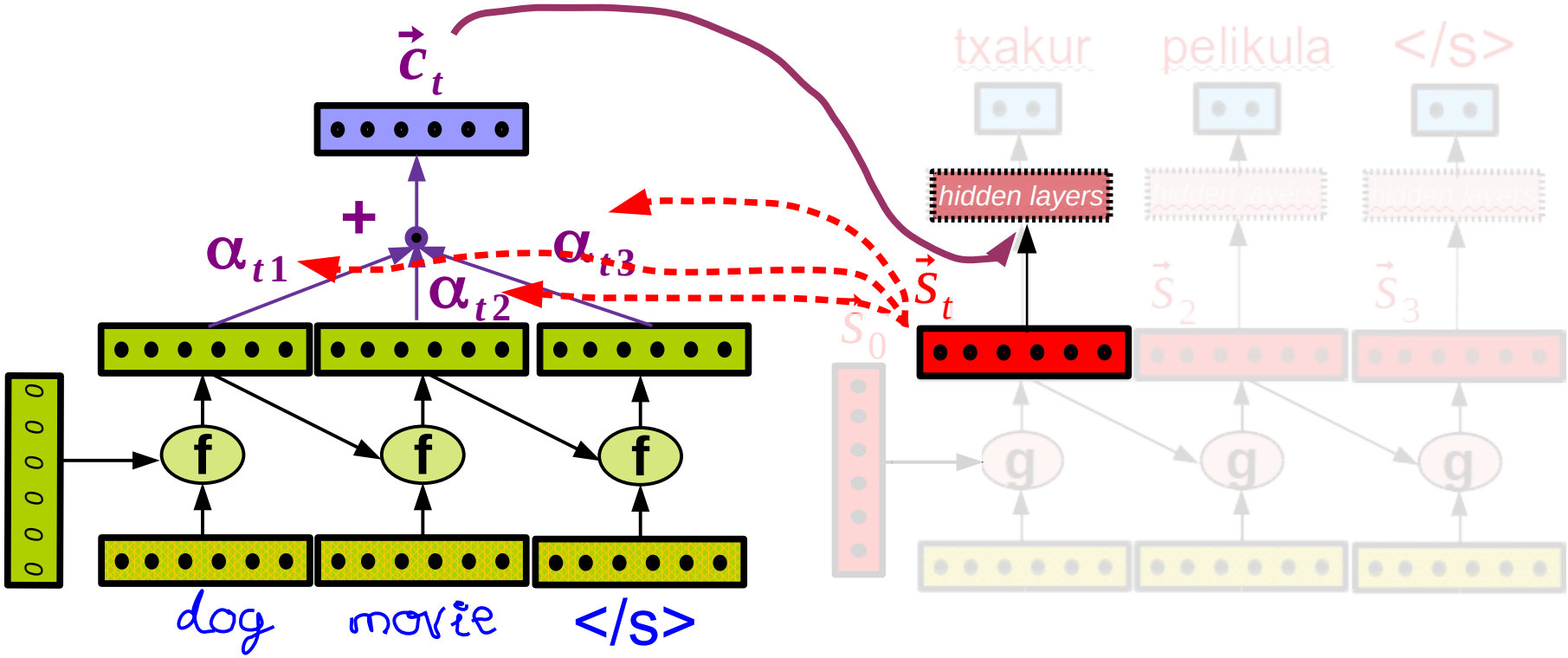
Re-thinking seq2seq for NMT

How can we access the necessary information at each decoding step?



Re-thinking seq2seq for NMT

Learn alignments jointly with the translation (!)
(Bahdanu et al. 2015; Luong et al. 2015)

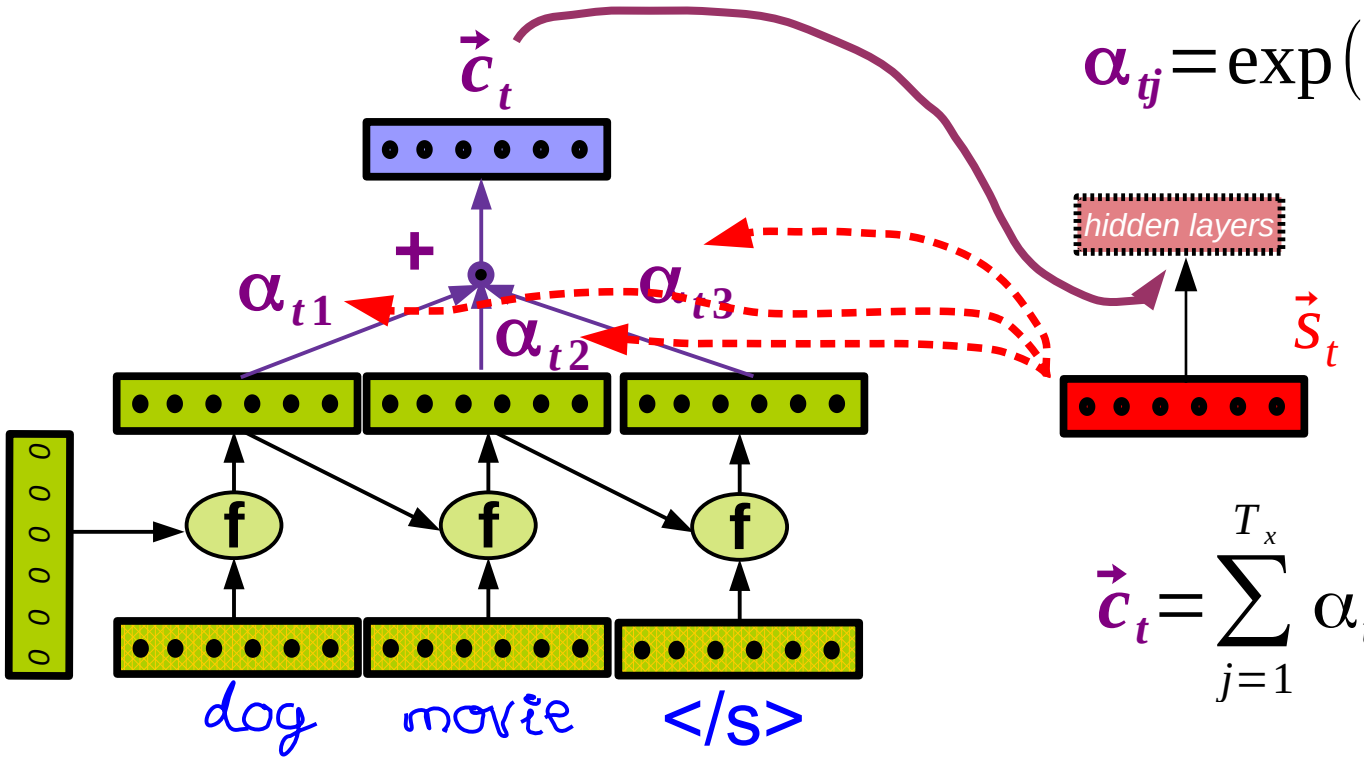


Re-thinking seq2seq for NMT

Learn alignments jointly with the translation (!)

$$e_{tj} = a(\vec{s}_t, \vec{h}_j) = \vec{s}_t W_a \vec{h}_j$$

$$\alpha_{tj} = \exp(e_{tj}) / \sum_{k=1}^{T_x} \exp(e_{tk})$$

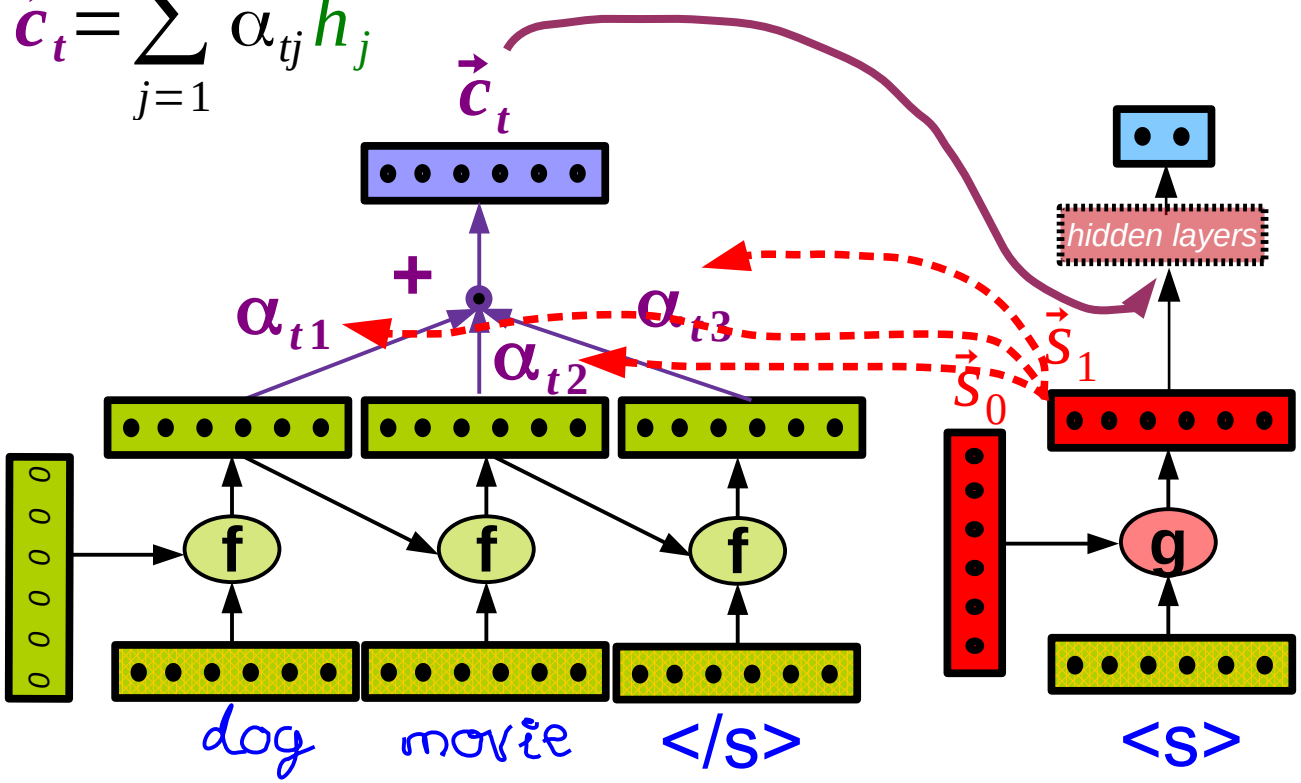


$$\vec{c}_t = \sum_{j=1}^{T_x} \alpha_{tj} \vec{h}_j$$

Re-thinking seq2seq for NMT

$$e_{tj} = a(\vec{s}_t, \vec{h}_j) = \vec{s}_t W_a \vec{h}_j \qquad \alpha_{tj} = \exp(e_{tj}) / \sum_{k=1}^{T_x} \exp(e_{tk})$$

$$\vec{c}_t = \sum_{j=1}^{T_x} \alpha_{tj} \vec{h}_j$$

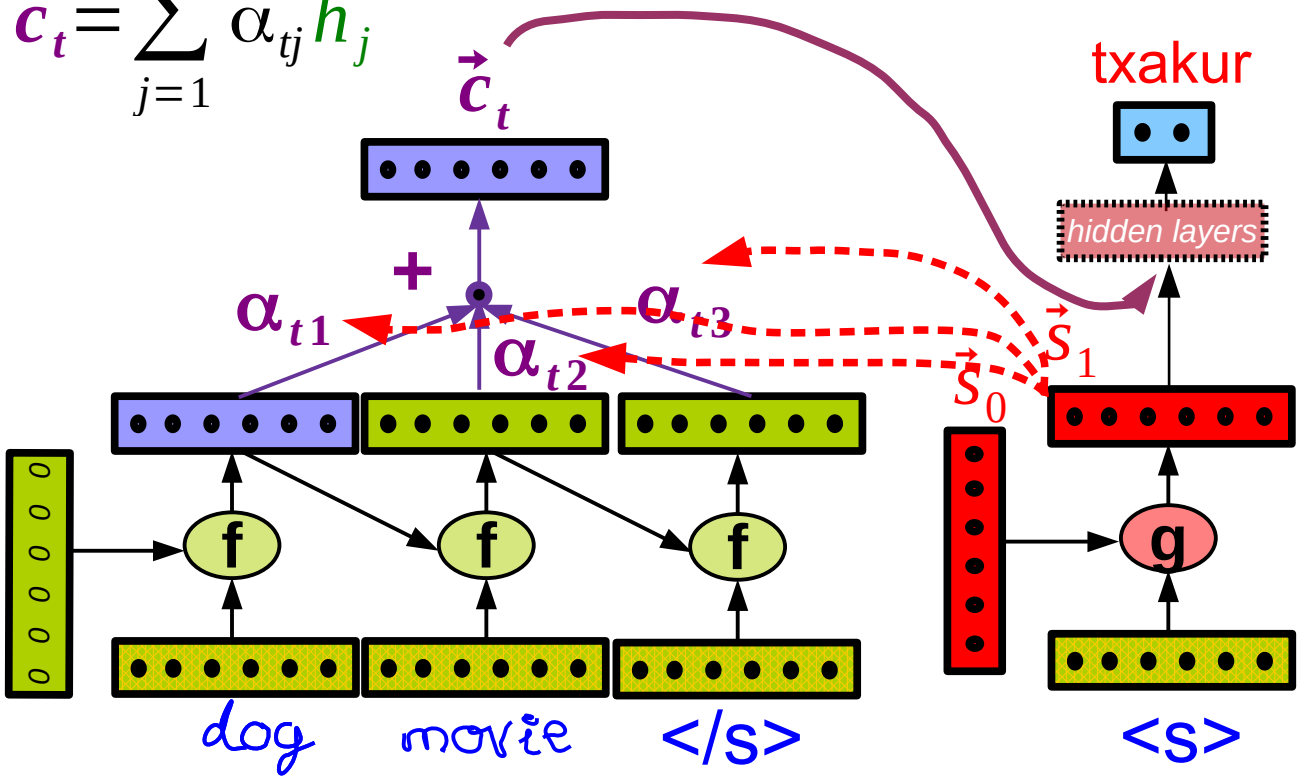


t=1

Re-thinking seq2seq for NMT

$$e_{tj} = a(\vec{s}_t, \vec{h}_j) = \vec{s}_t W_a \vec{h}_j \qquad \alpha_{tj} = \exp(e_{tj}) / \sum_{k=1}^{T_x} \exp(e_{tk})$$

$$\vec{c}_t = \sum_{j=1}^{T_x} \alpha_{tj} \vec{h}_j$$

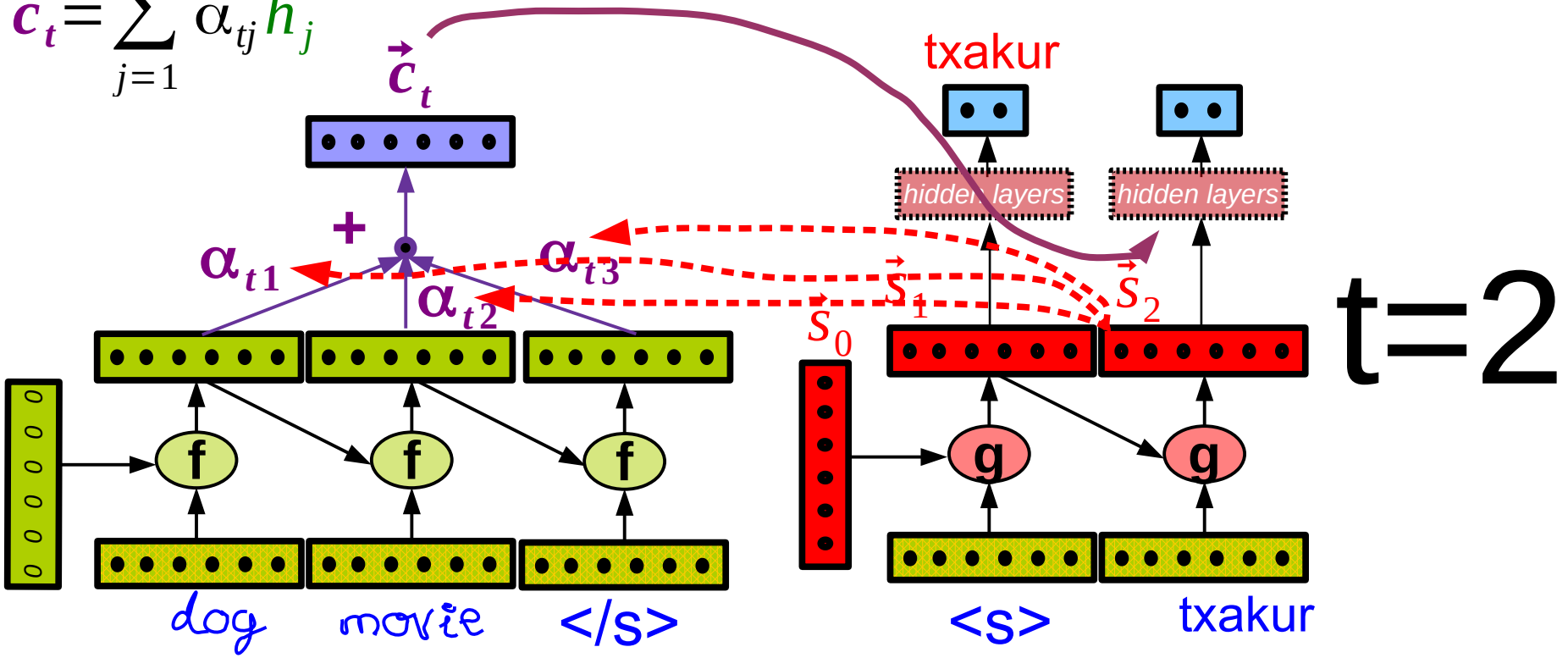


t=1

Re-thinking seq2seq for NMT

$$e_{tj} = a(\vec{s}_t, \vec{h}_j) = \vec{s}_t W_a \vec{h}_j \qquad \alpha_{tj} = \exp(e_{tj}) / \sum_{k=1}^{T_x} \exp(e_{tk})$$

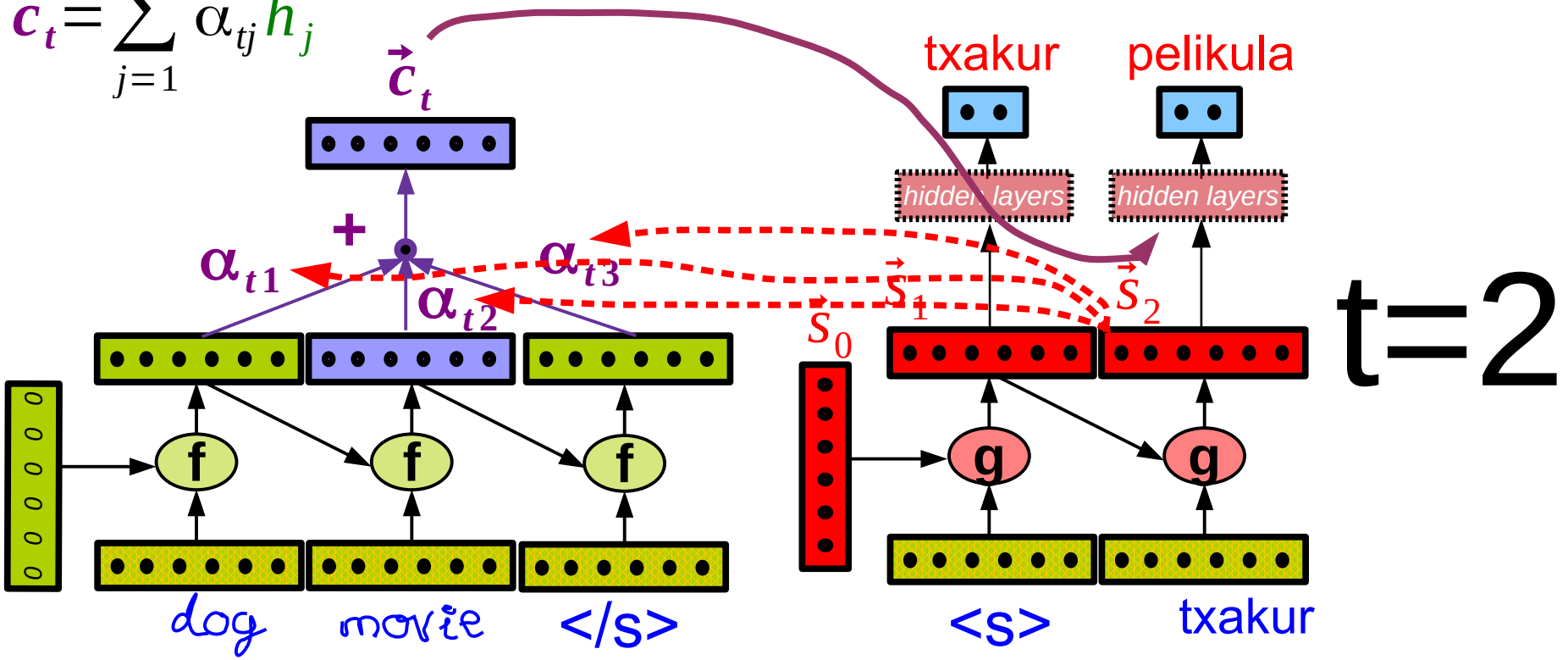
$$\vec{c}_t = \sum_{j=1}^{T_x} \alpha_{tj} \vec{h}_j$$



Re-thinking seq2seq for NMT

$$e_{tj} = a(\vec{s}_t, \vec{h}_j) = \vec{s}_t W_a \vec{h}_j \qquad \alpha_{tj} = \exp(e_{tj}) / \sum_{k=1}^{T_x} \exp(e_{tk})$$

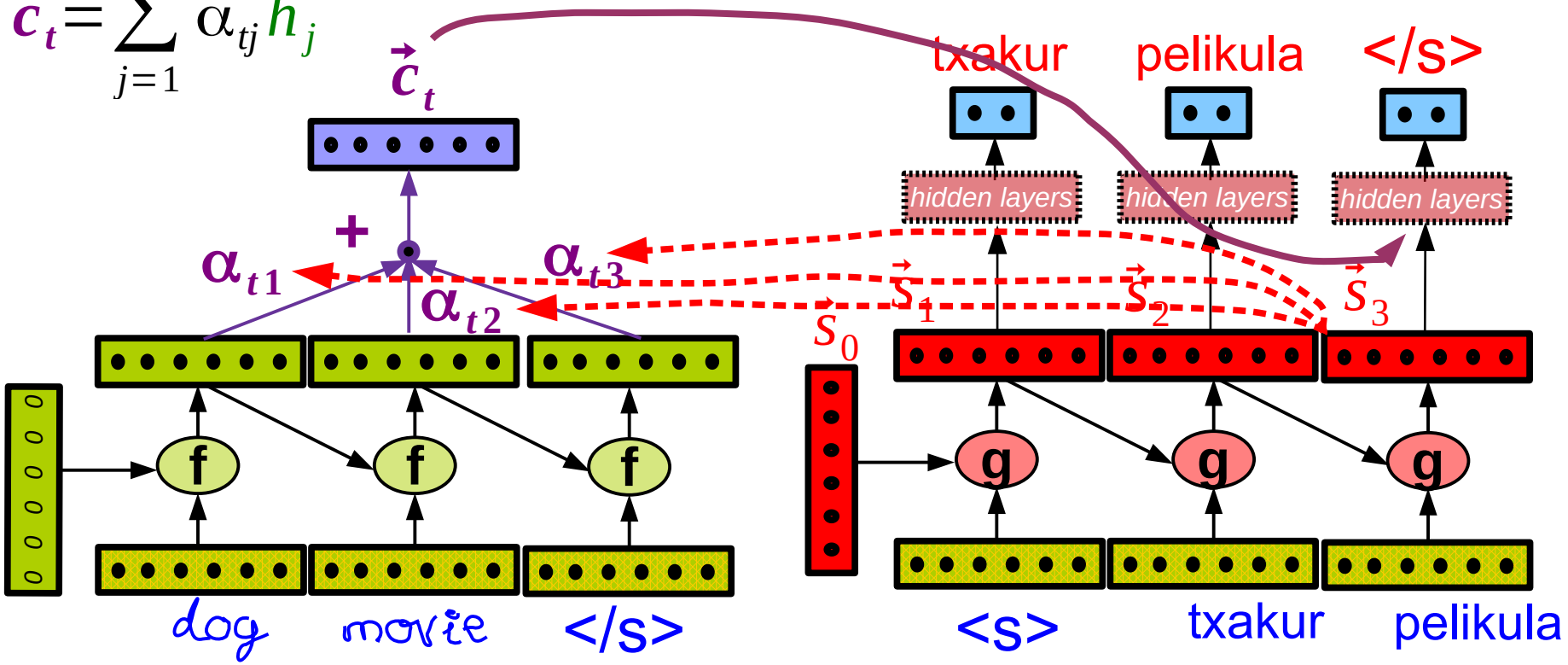
$$\vec{c}_t = \sum_{j=1}^{T_x} \alpha_{tj} \vec{h}_j$$



Re-thinking seq2seq for NMT

$$e_{tj} = a(\vec{s}_t, \vec{h}_j) = \vec{s}_t W_a \vec{h}_j \qquad \alpha_{tj} = \exp(e_{tj}) / \sum_{k=1}^{T_x} \exp(e_{tk})$$

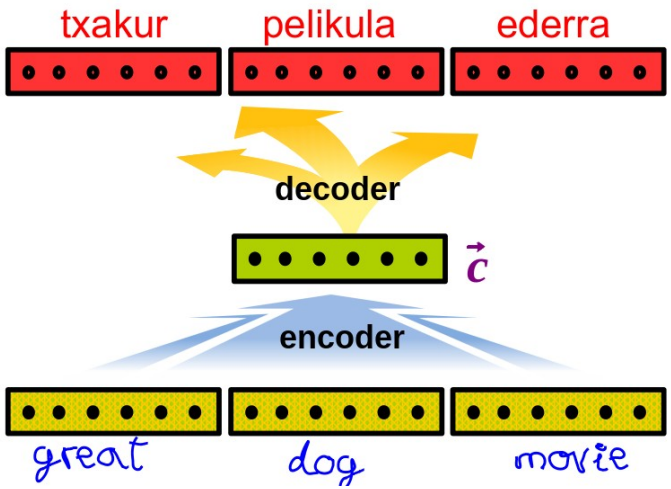
$$\vec{c}_t = \sum_{j=1}^{T_x} \alpha_{tj} \vec{h}_j$$



Re-thinking seq2seq for NMT

Back to sentence representations:

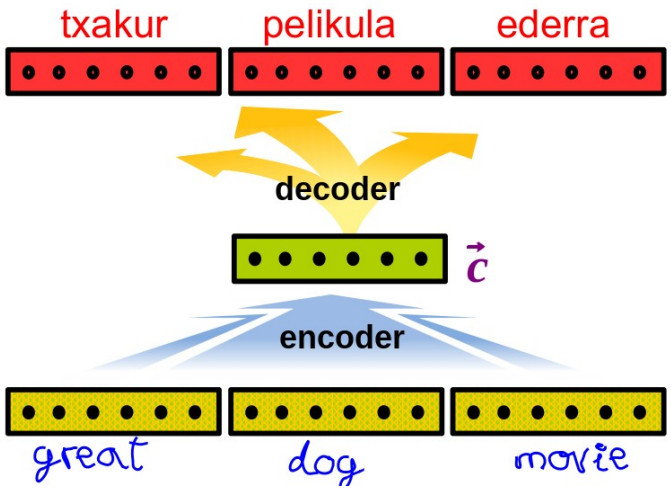
- Instead of last hidden state ...

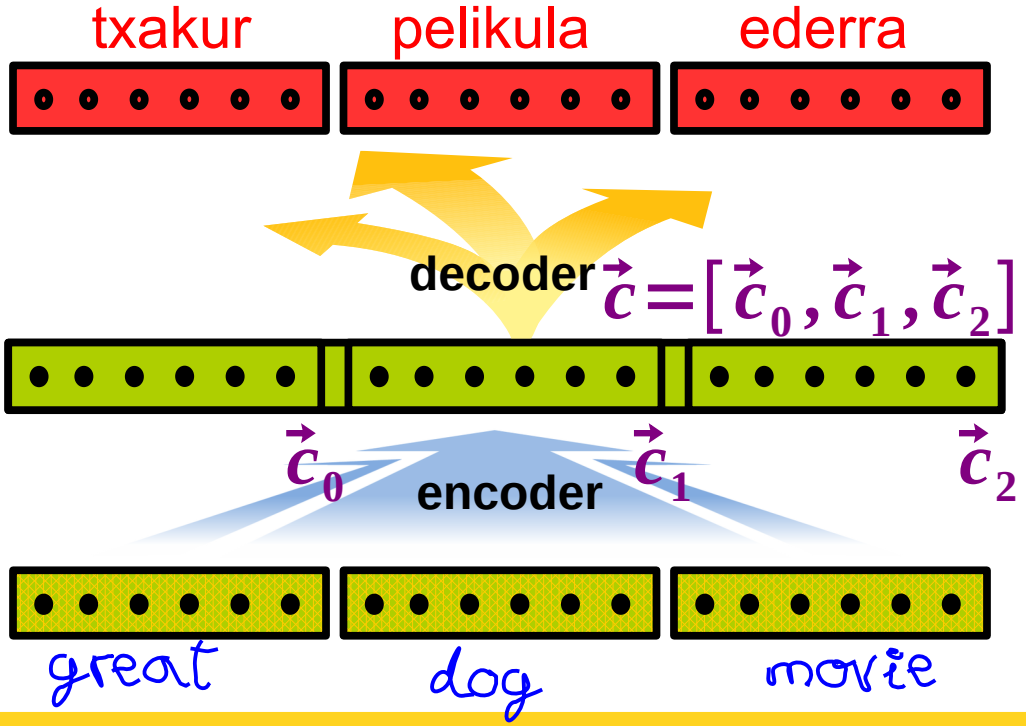


Re-thinking seq2seq for NMT

Back to sentence representations:

- Instead of hidden state ...
- Attention: **all** hidden states

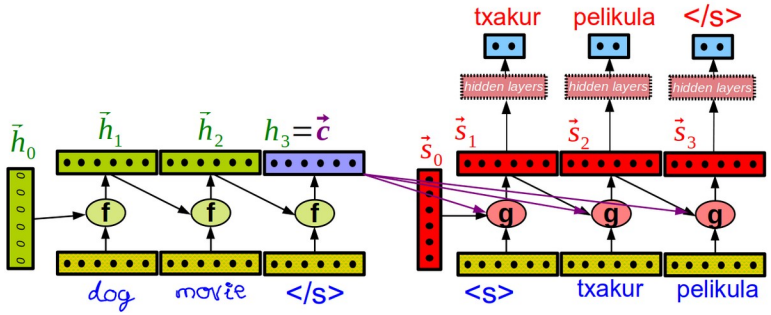




Re-thinking seq2seq for NMT

How do we train?

Do we need special training for attention? Any differences with respect to our previous NMT model?

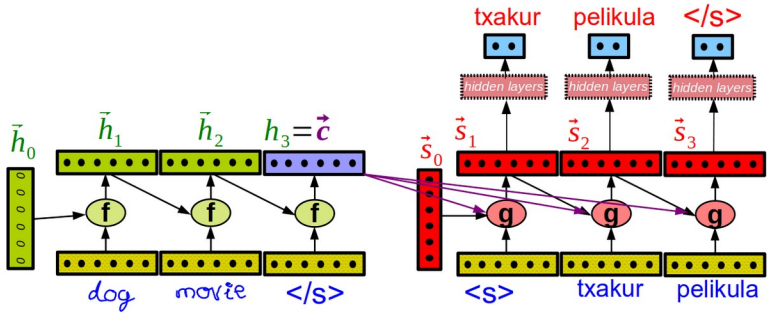


Re-thinking seq2seq for NMT

How do we train?

Do we need special training for attention?Any differences with respect to our previous NMT model?

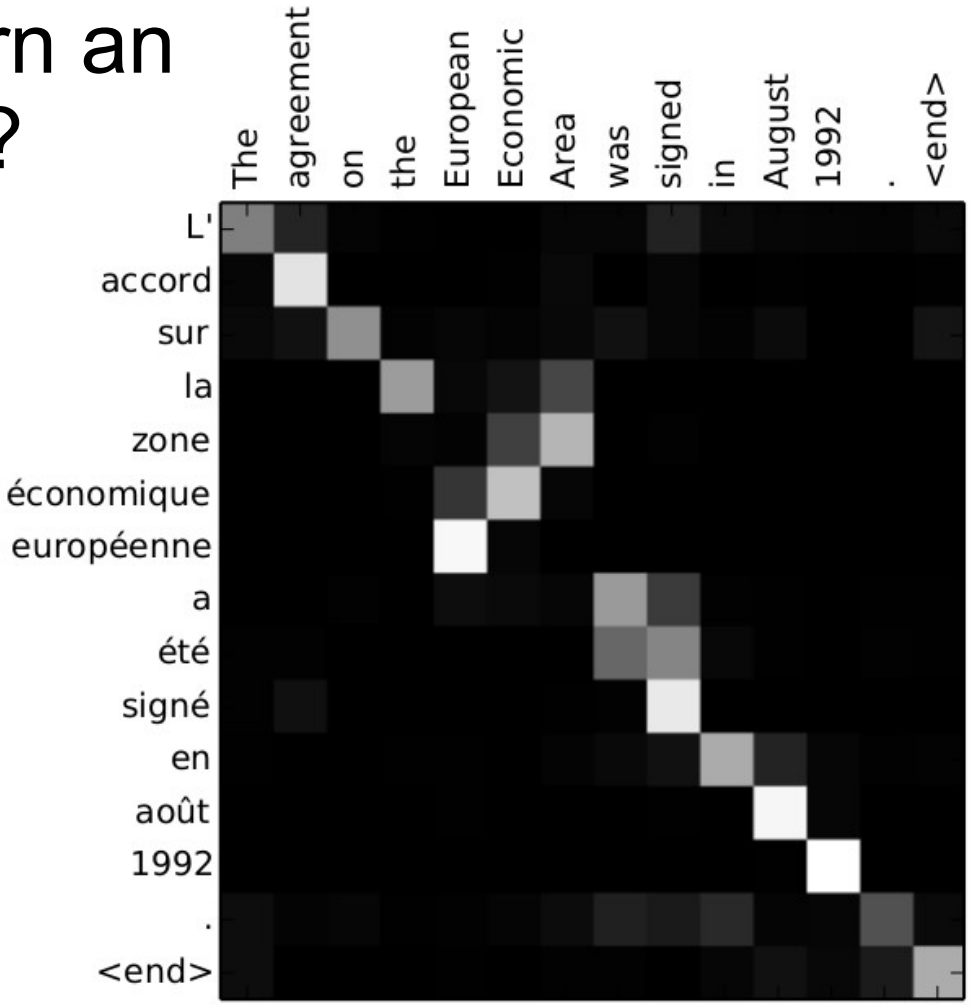
- Same loss function
- Same training strategy:
  - Gradients have changed!!
    But thanks to autograd we do not need to write code for them.



Re-thinking seq2seq for NMT

Does it really learn an alignment model?

Plot of α_{ij}



Source: Bahdanu et al. 2015 - ICLR

Re-thinking seq2seq for NMT

Does it work well?

System	Ppl	BLEU
Winning WMT'14 system – *phrase-based + large LM* (Buck et al., 2014)		20.7
Existing NMT systems		
RNNsearch (Jean et al., 2015)		16.5
RNNsearch + unk replace (Jean et al., 2015)		19.0
RNNsearch + unk replace + large vocab + *ensemble* 8 models (Jean et al., 2015)		**21.6**
Our NMT systems		
Base	10.6	11.3
Base + reverse	9.9	12.6 (+*1.3*)
Base + reverse + dropout	8.1	14.0 (+*1.4*)
Base + reverse + dropout + global attention (*location*)	7.3	16.8 (+*2.8*)
Base + reverse + dropout + global attention (*location*) + feed input	6.4	18.1 (+*1.3*)
Ensemble 8 models + unk replace		**23.0** (+*2.1*)

Table 1: **WMT'14 English-German results** – shown are the perplexities (ppl) and the *tokenized* BLEU

Source: Luong et al. 2015 - EMNLP

Attention is useful

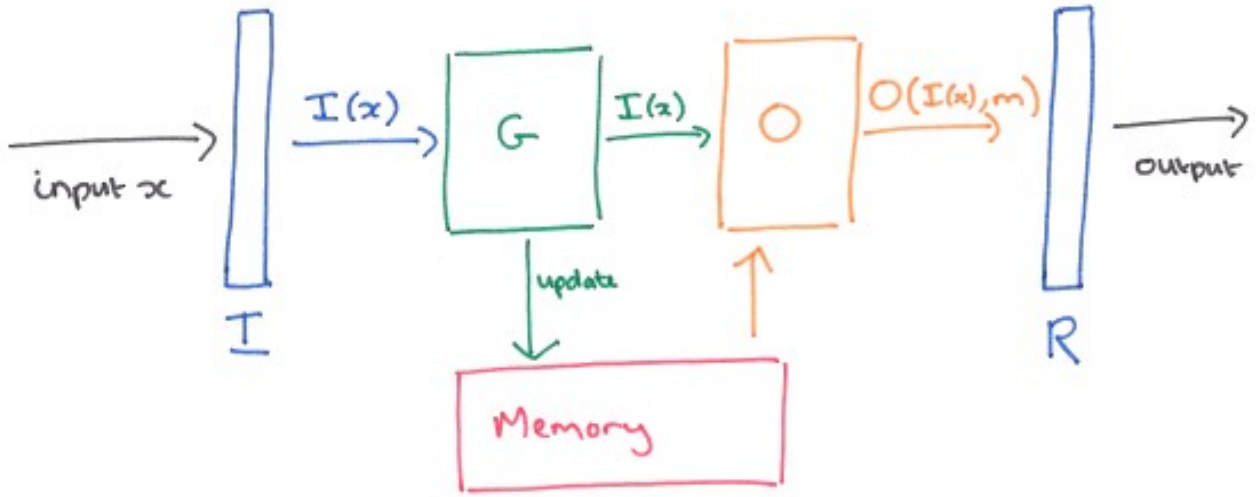
Caption generation:

Source: Xu et al. 2015 - Arxiv 1502.03044

Attention is useful

Memory networks (Sukhbaatar et al. 2016) (also Neural Turing Machine) :

- Set up external knowledge base (memory)
- Use input to select memory cells via attention
- Update selected memory cells



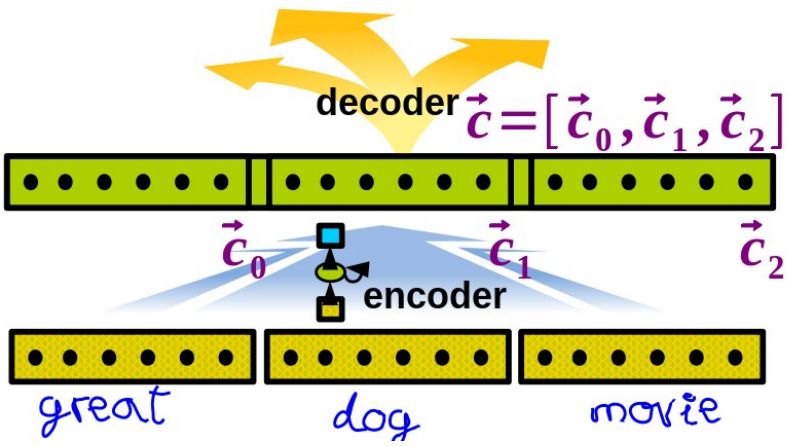
Plan for this session

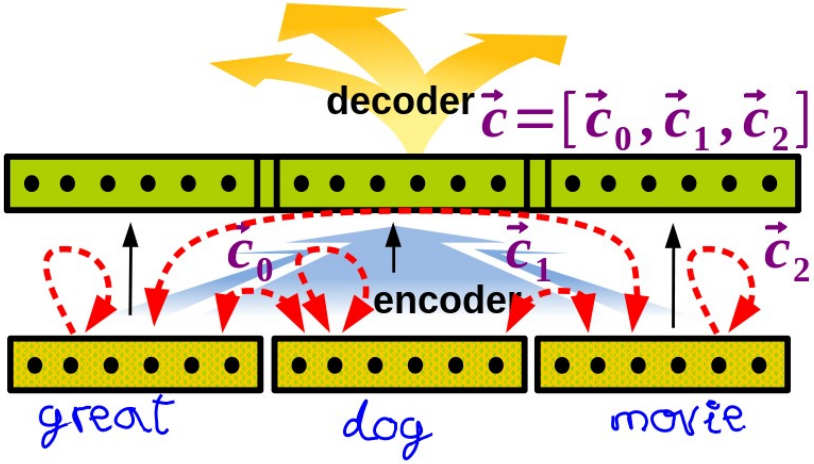
- **Re-thinking seq2seq:**
  - Attention and memory
  - **State of the art NMT: self-attention (transformers)**
  - Amazing things:
    - Multilingual MT
    - MT without any bilingual data
- Evaluating sentence representations

Attention is all you need (!?)

Transformer (Vaswani et al. 2017)

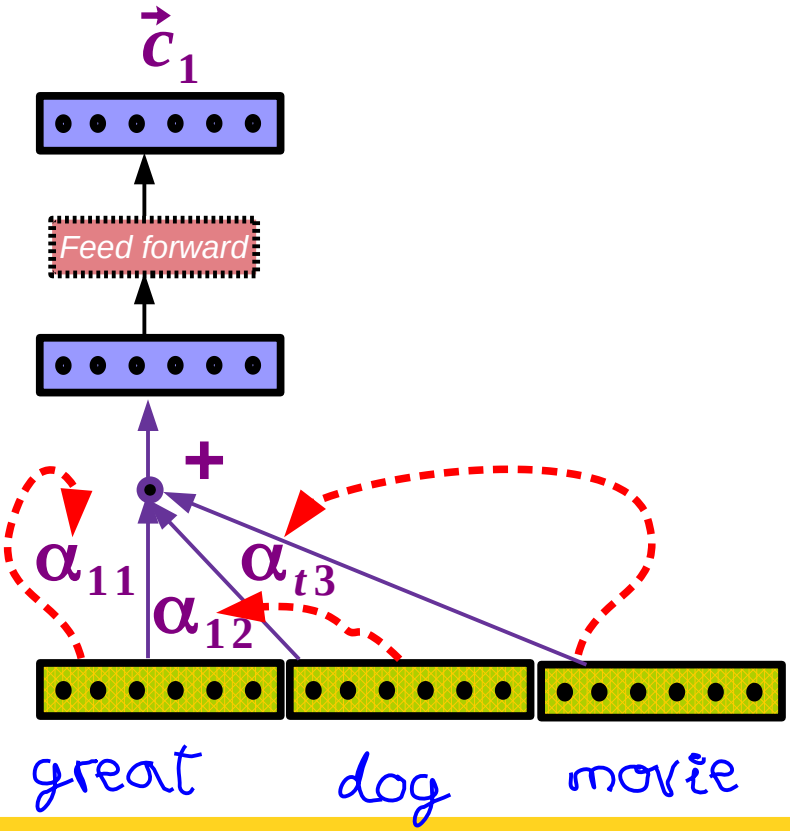
- Extend attention to self-attention: source (and target) with itself
- Build hidden states using feed forward networks





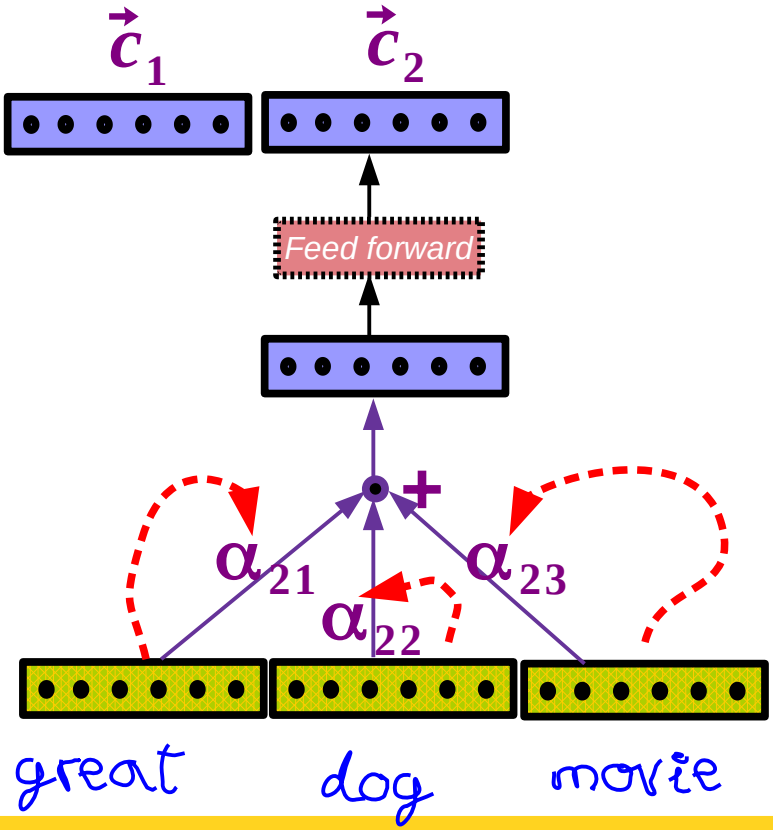
Attention is all you need (!?)

Transformer (Vaswani et al. 2017)



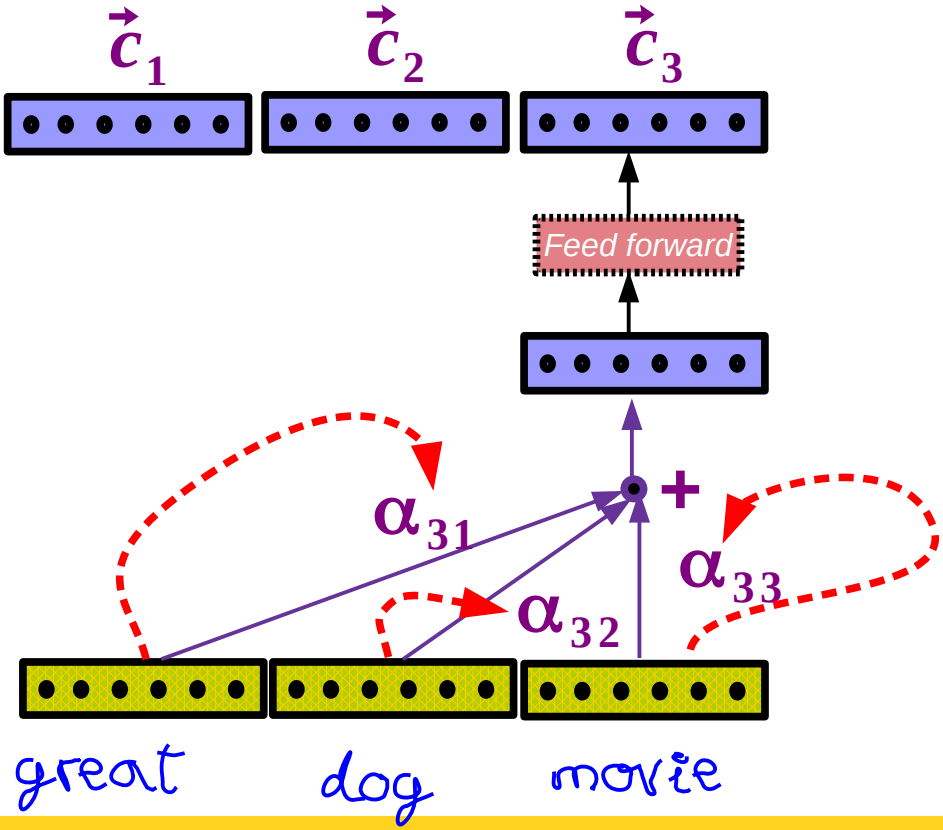
Attention is all you need (!?)

Transformer (Vaswani et al. 2017)



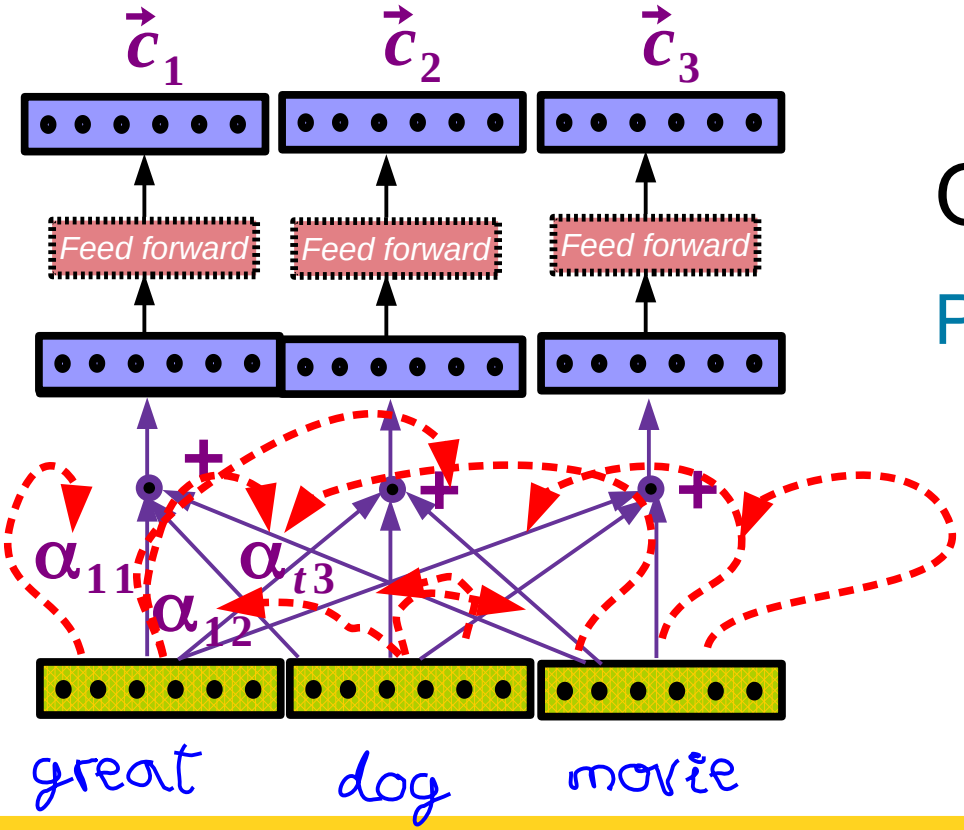
Attention is all you need (!?)

Transformer (Vaswani et al. 2017)



Attention is all you need (!?)

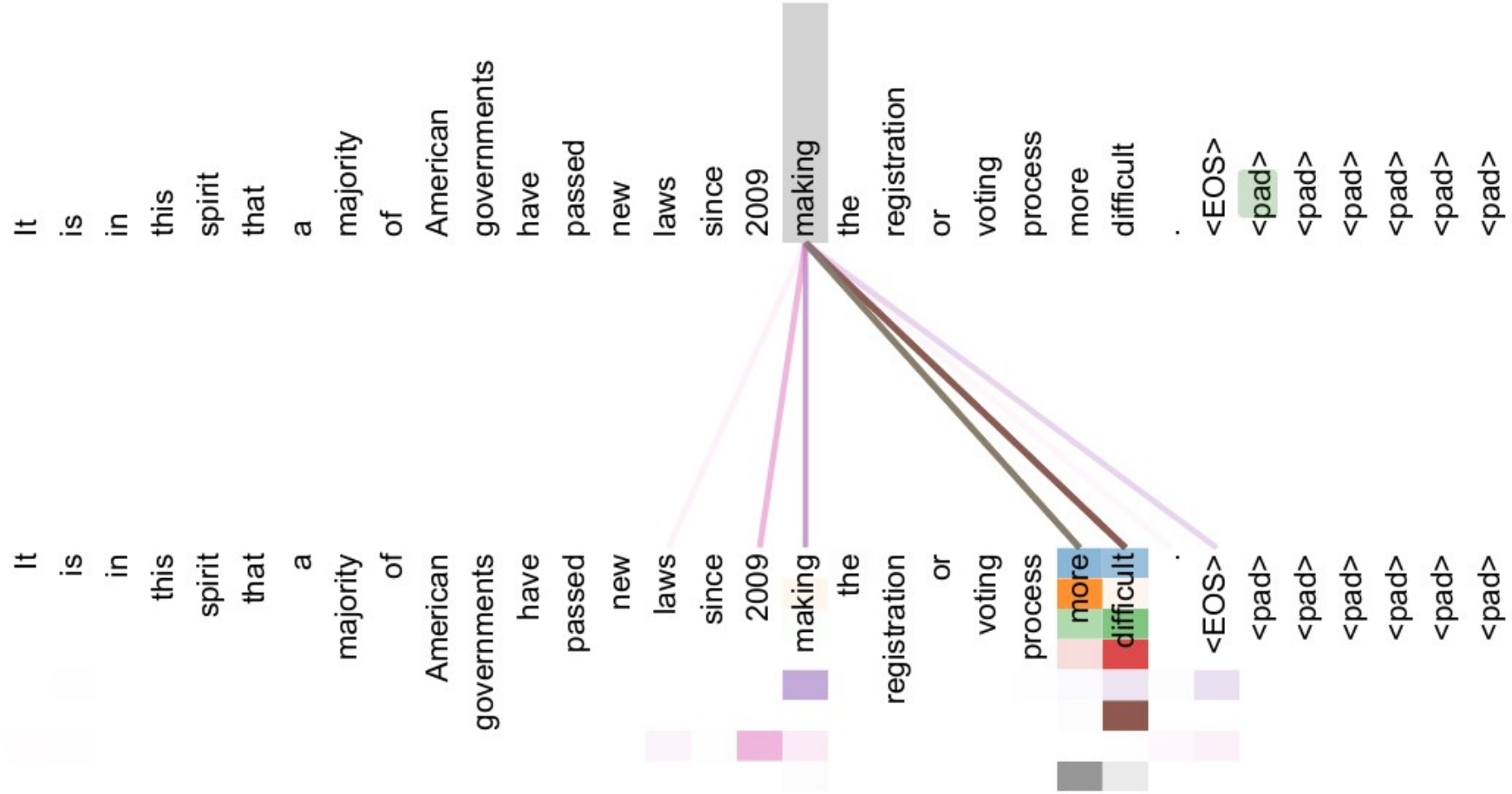
Transformer (Vaswani et al. 2017)



One forward pass!

Promotional video MT

Attention is all you need (!?)



Source: (Vaswani et al. 2017)

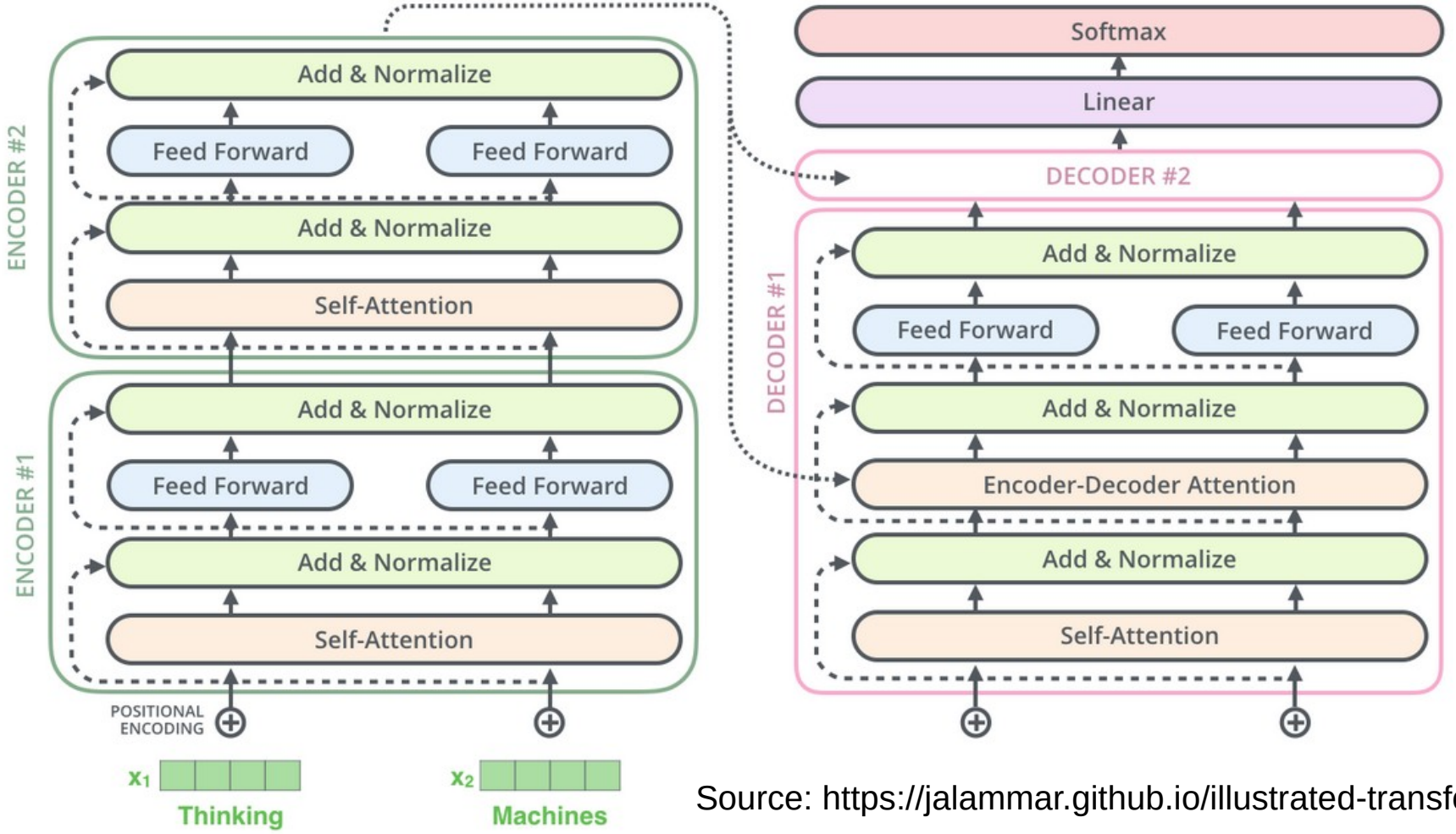
Attention is all you need (!?)

Table 1: Maximum path lengths, per-layer complexity and minimum number of sequential operations for different layer types. n is the sequence length, d is the representation dimension, k is the kernel size of convolutions and r the size of the neighborhood in restricted self-attention.

Layer Type	Complexity per Layer	Sequential Operations	Maximum Path Length
Self-Attention	$O(n^2 \cdot d)$	$O(1)$	$O(1)$
Recurrent	$O(n \cdot d^2)$	$O(n)$	$O(n)$
Convolutional	$O(k \cdot n \cdot d^2)$	$O(1)$	$O(log_k(n))$
Self-Attention (restricted)	$O(r \cdot n \cdot d)$	$O(1)$	$O(n/r)$

Source: (Vaswani et al. 2017)

Attention is all you need (!?)



Source: https://jalammar.github.io/illustrated-transformer/

Attention is all you need (!?)

Table 2: The Transformer achieves better BLEU scores than previous state-of-the-art models on the English-to-German and English-to-French newstest2014 tests at a fraction of the training cost.

Model	BLEU		Training Cost (FLOPs)	
	EN-DE	EN-FR	EN-DE	EN-FR
ByteNet [18]	23.75			
Deep-Att + PosUnk [39]		39.2		$1.0 \cdot 10^{20}$
GNMT + RL [38]	24.6	39.92	$2.3 \cdot 10^{19}$	$1.4 \cdot 10^{20}$
ConvS2S [9]	25.16	40.46	$9.6 \cdot 10^{18}$	$1.5 \cdot 10^{20}$
MoE [32]	26.03	40.56	$2.0 \cdot 10^{19}$	$1.2 \cdot 10^{20}$
Deep-Att + PosUnk Ensemble [39]		40.4		$8.0 \cdot 10^{20}$
GNMT + RL Ensemble [38]	26.30	41.16	$1.8 \cdot 10^{20}$	$1.1 \cdot 10^{21}$
ConvS2S Ensemble [9]	26.36	**41.29**	$7.7 \cdot 10^{19}$	$1.2 \cdot 10^{21}$
Transformer (base model)	27.3	38.1	**$3.3 \cdot 10^{18}$**	
Transformer (big)	**28.4**	**41.8**	$2.3 \cdot 10^{19}$	

Source: http://nlp.seas.harvard.edu/2018/04/03/attention.html (with code!!)

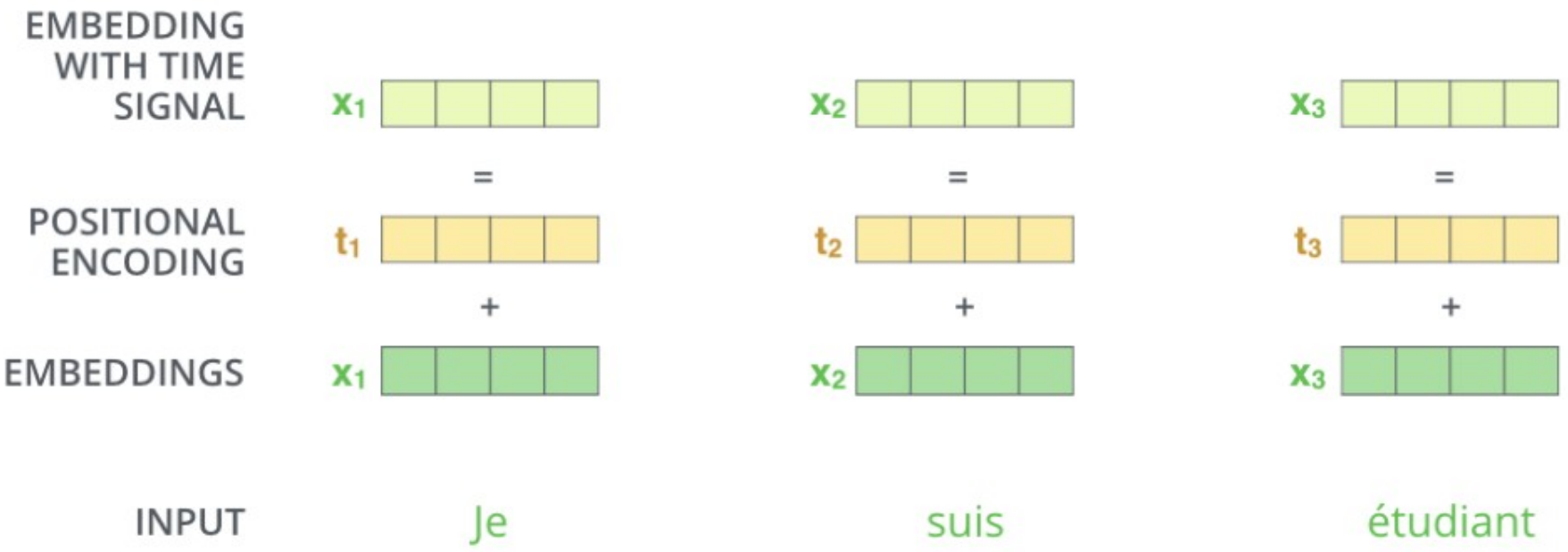
Attention is all you need (!?)

In addition:

- Multi-head
- Positional encodings
- Layer normalization
- Unknown words (out-of-vocabulary OOV)

Attention is all you need (!?)

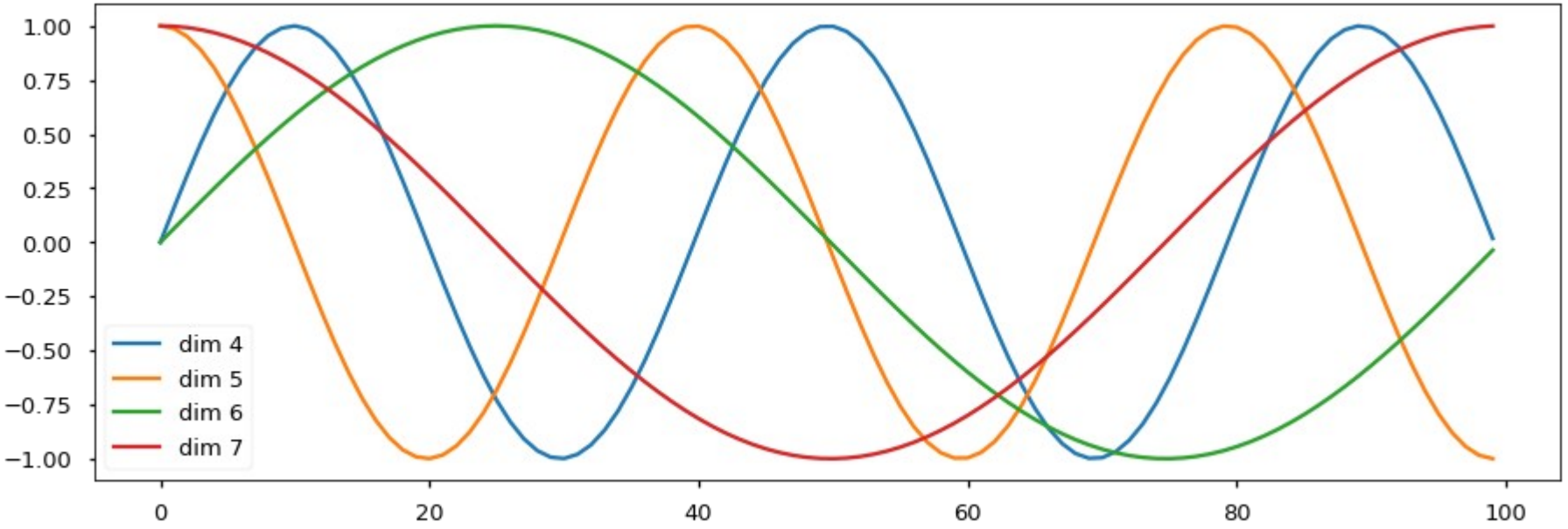
Positional encodings: sin/cos for each dimension (different freq / offset)



Source: https://jalammar.github.io/illustrated-transformer/

Attention is all you need (!?)

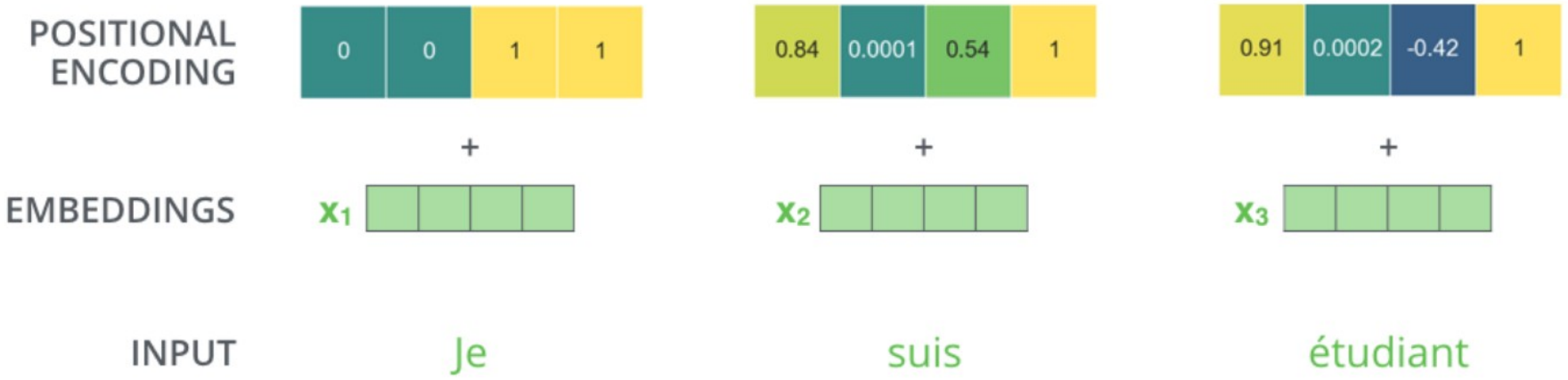
Positional encodings: sin/cos for each dimension (different freq / offset)



Source: http://nlp.seas.harvard.edu/2018/04/03/attention.html

Attention is all you need (!?)

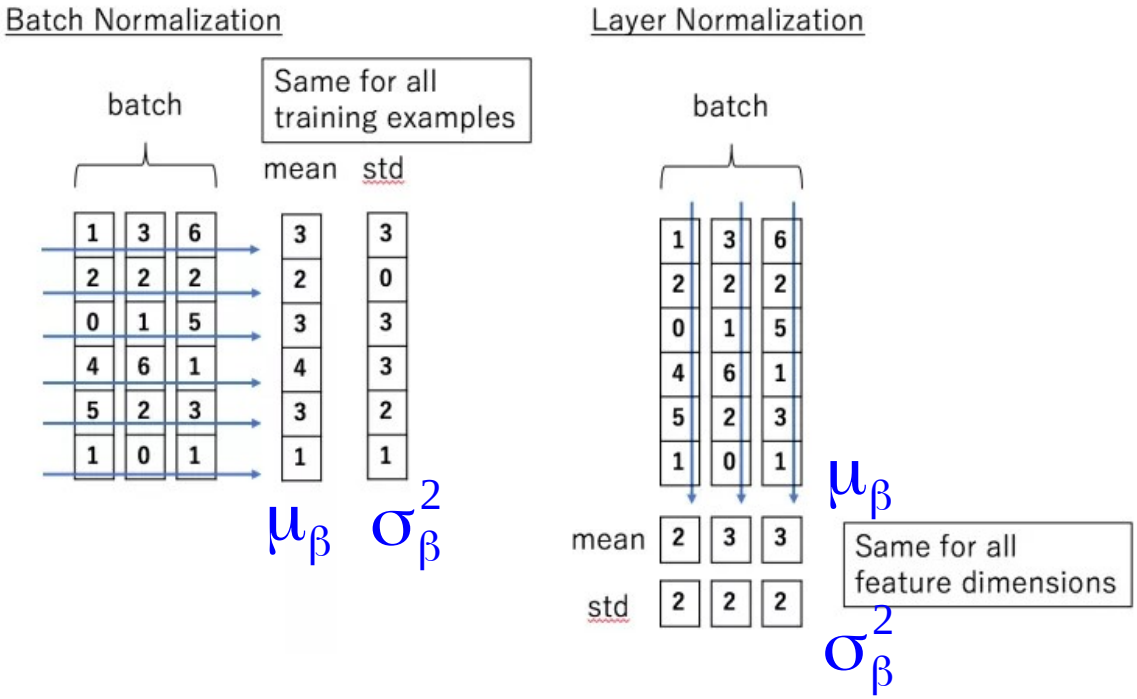
Positional encodings: sin/cos for each dimension (different freq / offset)



Source: https://jalammar.github.io/illustrated-transformer/

Attention is all you need (!?)

Layer normalization



$$\hat{x}_i \leftarrow \frac{x_i - \mu_\beta}{\sqrt{\sigma_\beta^2 + \epsilon}}$$

Source: https://arxiv.org/pdf/1502.03167v3.pdf

Attention is all you need (!?)

Out Of Vocabulary: fixed vocabulary size means some words need to be mapped to <UNK> token.

Very important for morphologically complex languages.

Solutions:

- Copy input tokens into output (**MT only**)
- Add character tokens for OOV words

  Miki → <B>M <M>i <M>k <E>i
- **Word segments to vocabulary (BPE, sentencepiece)**

  Jet makers feud over seat width with big orders at stake

  _J et _makers _fe ud _over _seat _width _with _big _orders ...
-

Attention is all you need (!?)

In addition:

- Multi-head
- Positional encodings
- Layer normalization
- Unknown words (out-of-vocabulary OOV)

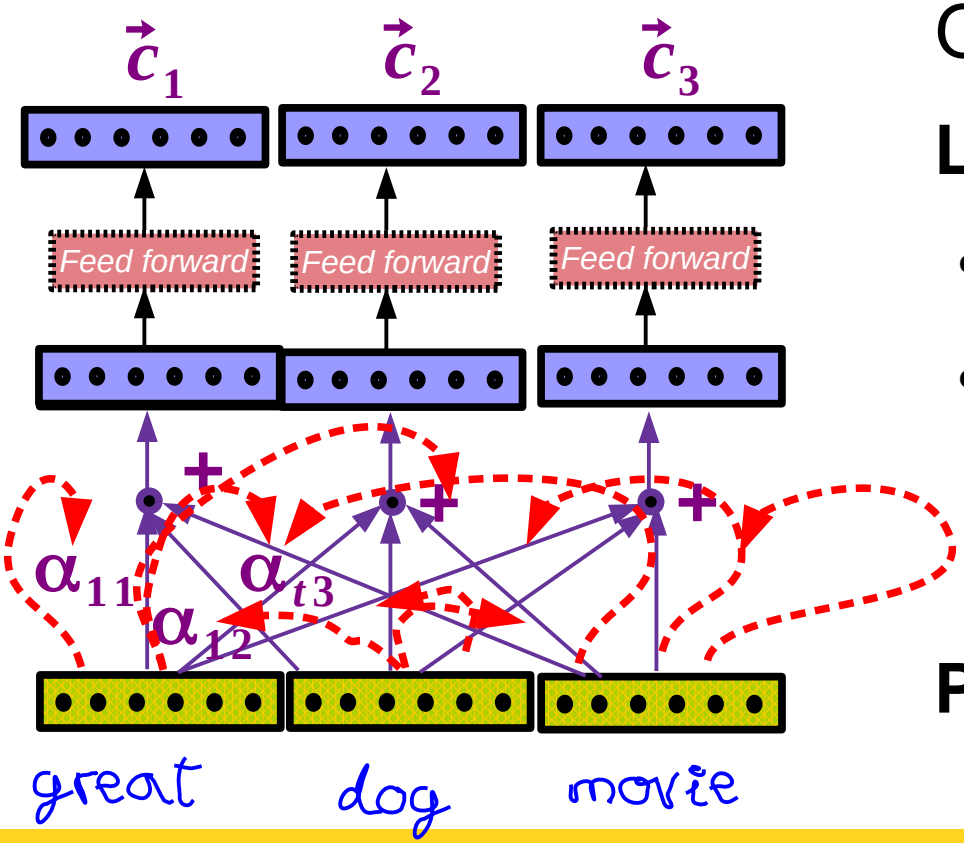
And for optimal NMT:

- Optimization with schedule: Adam, plus
  increase learning rate on warmup to decrease it later
- Tight batches with minimum padding
- Larger batch sizes (25K tokens source/target, 50K total)
- Dimensions: embeddings 512, inner-layer 2048
- Multi-GPU (8), beam search for decoding

http://nlp.seas.harvard.edu/2018/04/03/attention.html with code

Attention is all you need (!?)

Transformer (Vaswani et al. 2017)



One forward pass!

Limitations:

- **Max. sentence length**
- **No fixed-dimensional vector**

Padding is not a problem

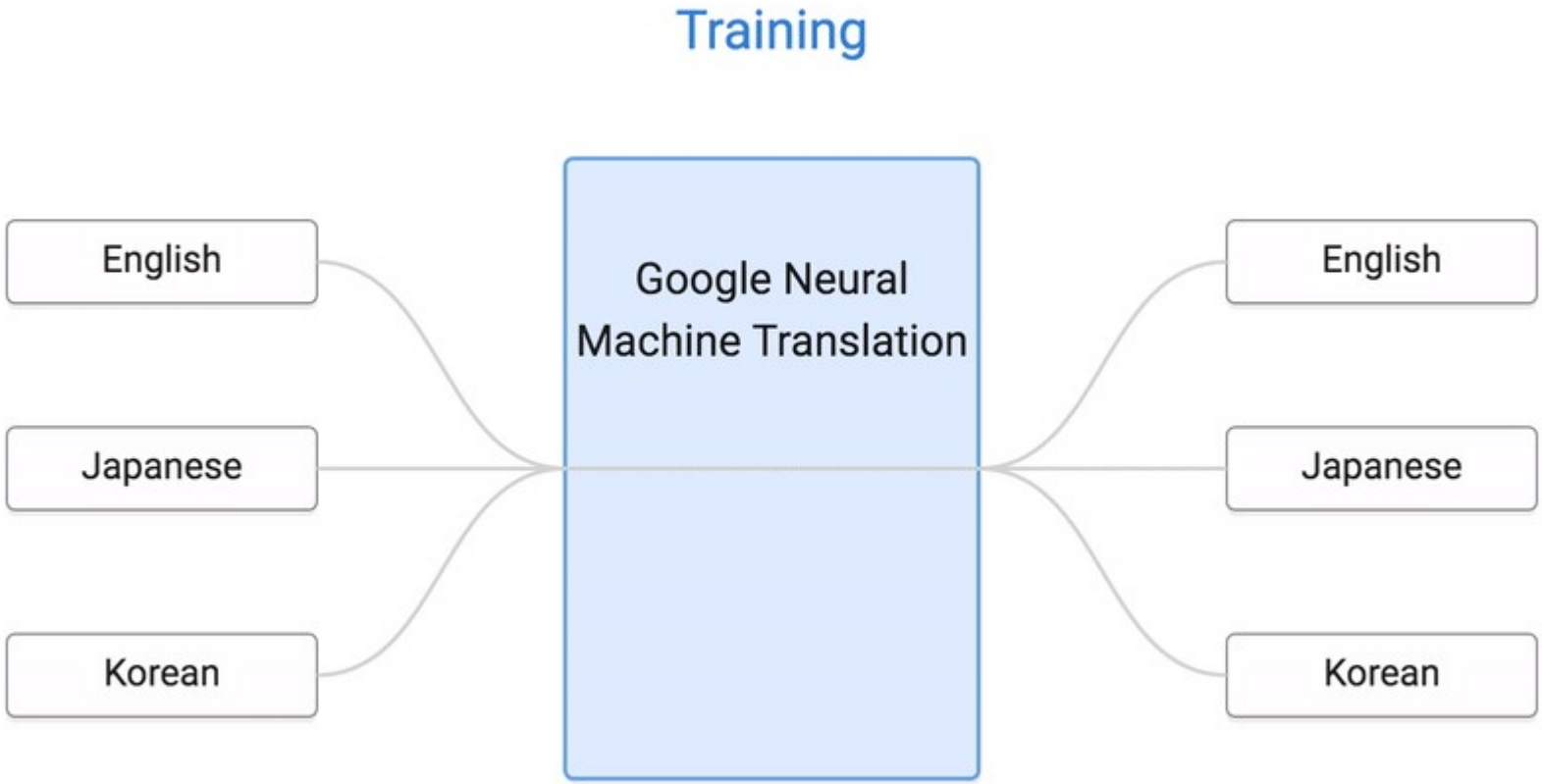
Plan for this session

- **Re-thinking seq2seq:**
  - Attention and memory
  - State of the art NMT: self-attention (transformers)
  - **Amazing things:**
    - Multilingual MT
    - MT without any bilingual data
- Evaluating sentence representations

Amazing things with NMT

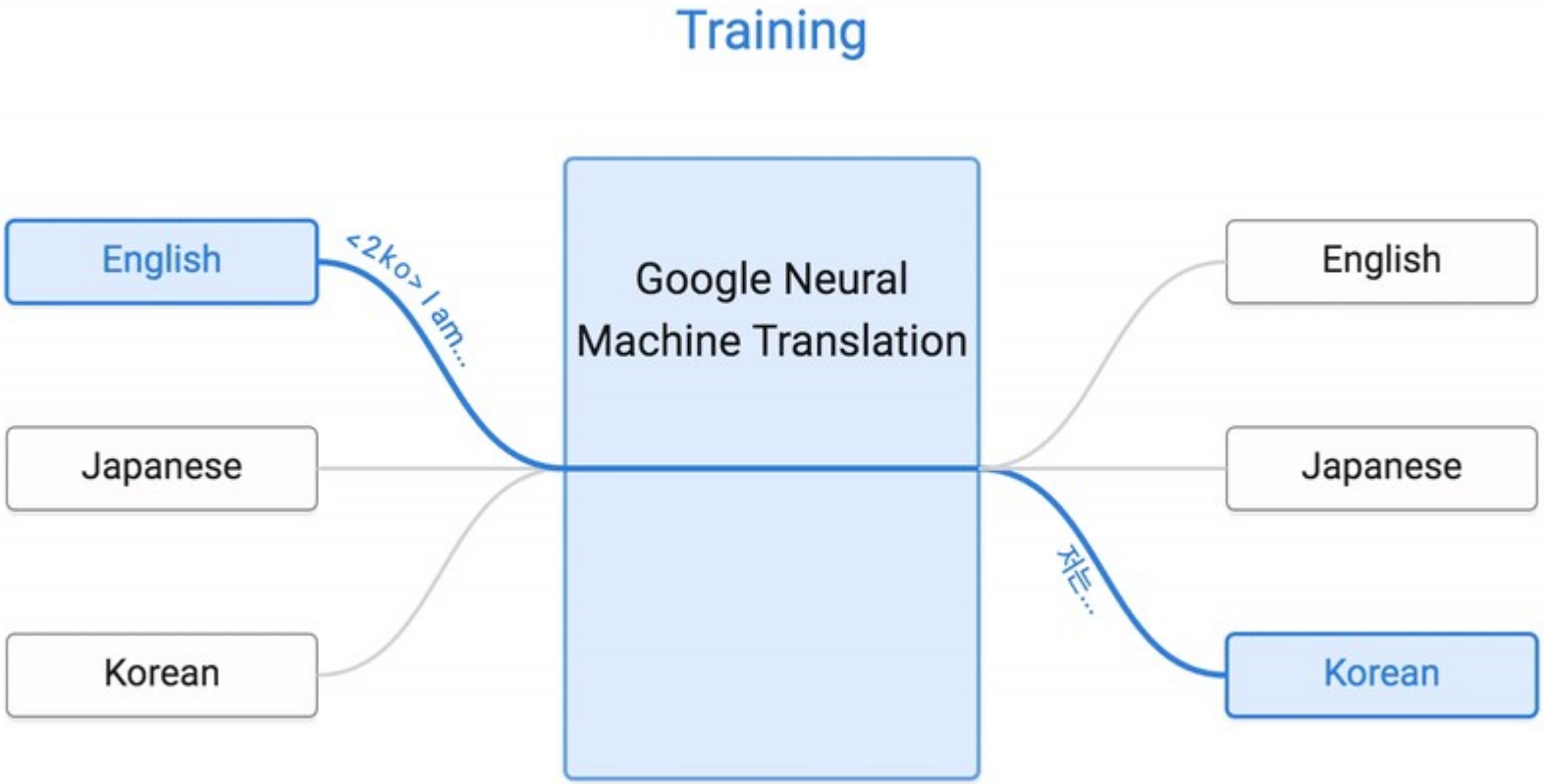
- Given parallel data for
  - English ↔ Korean
  - English ↔ Japanese
- Is it possible to translate for unseen pairs?
  - Japanese ↔ Korean?
- Same encoder and decoder for all languages
  - Add single token to define desired output language: **<2Ko>**

Multilingual NMT



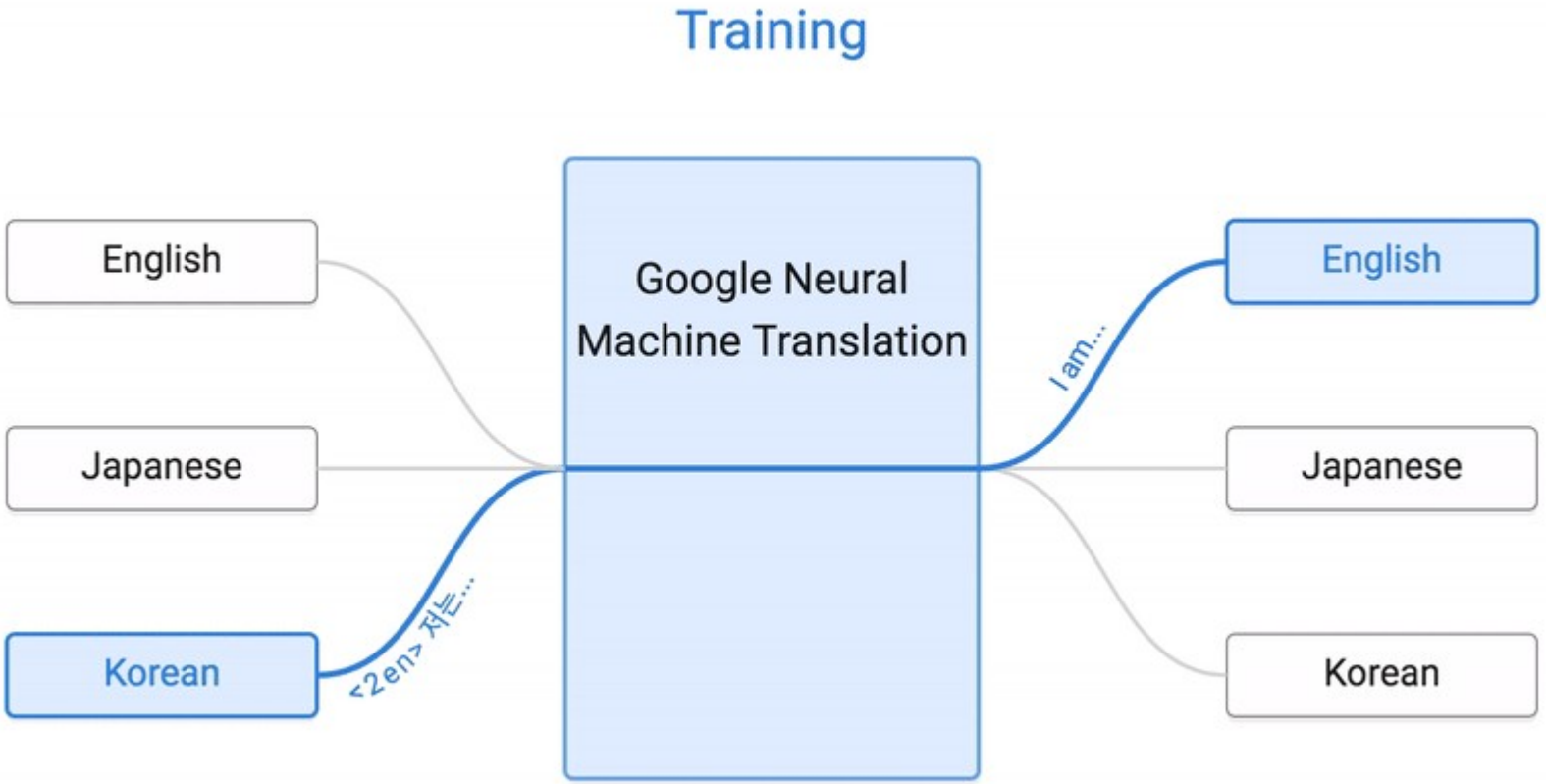
https://research.googleblog.com/2016/11/zero-shot-translation-with-googles.html

Multilingual NMT



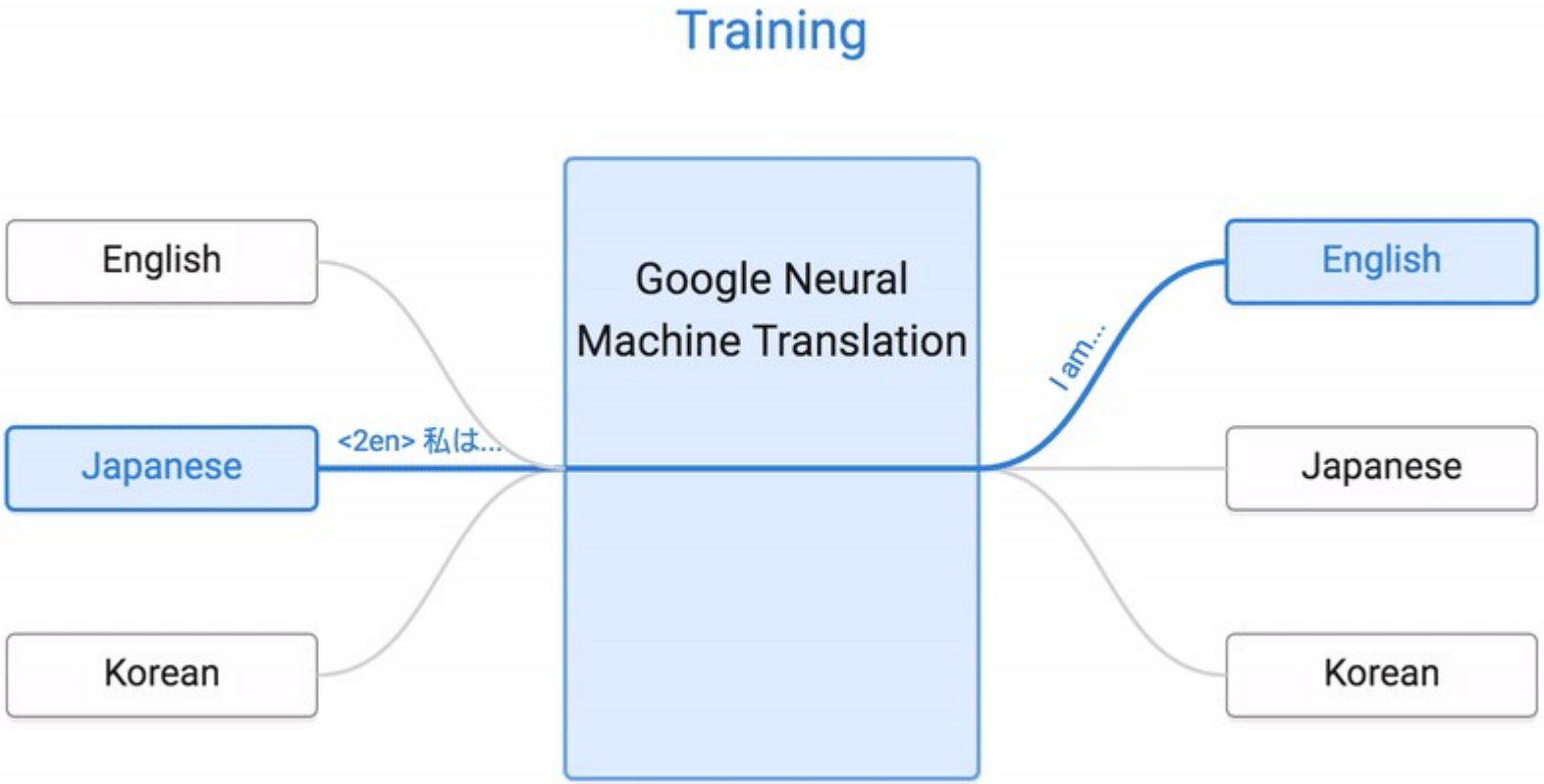
https://research.googleblog.com/2016/11/zero-shot-translation-with-googles.html

Multilingual NMT



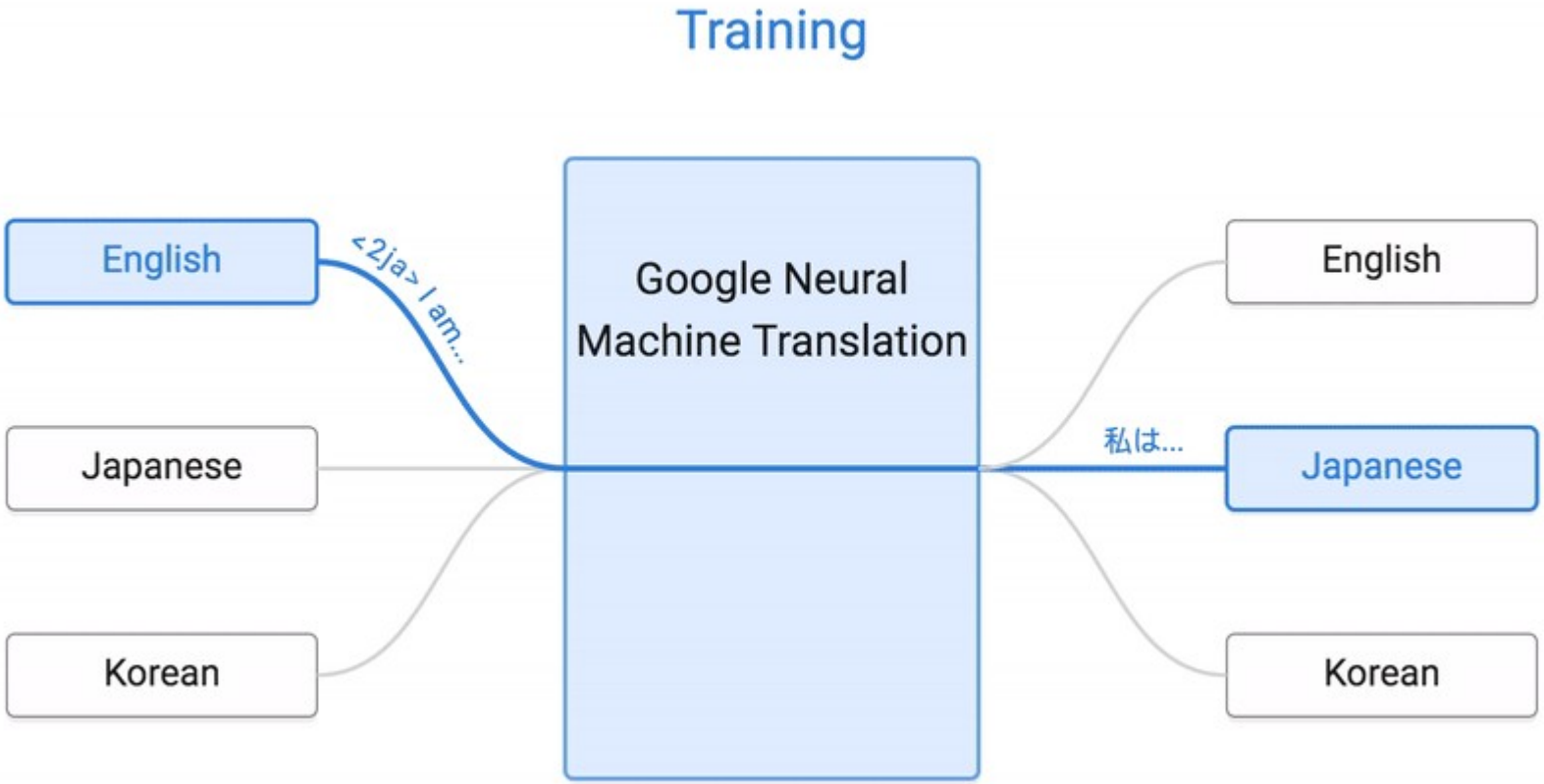
https://research.googleblog.com/2016/11/zero-shot-translation-with-googles.html

Multilingual NMT



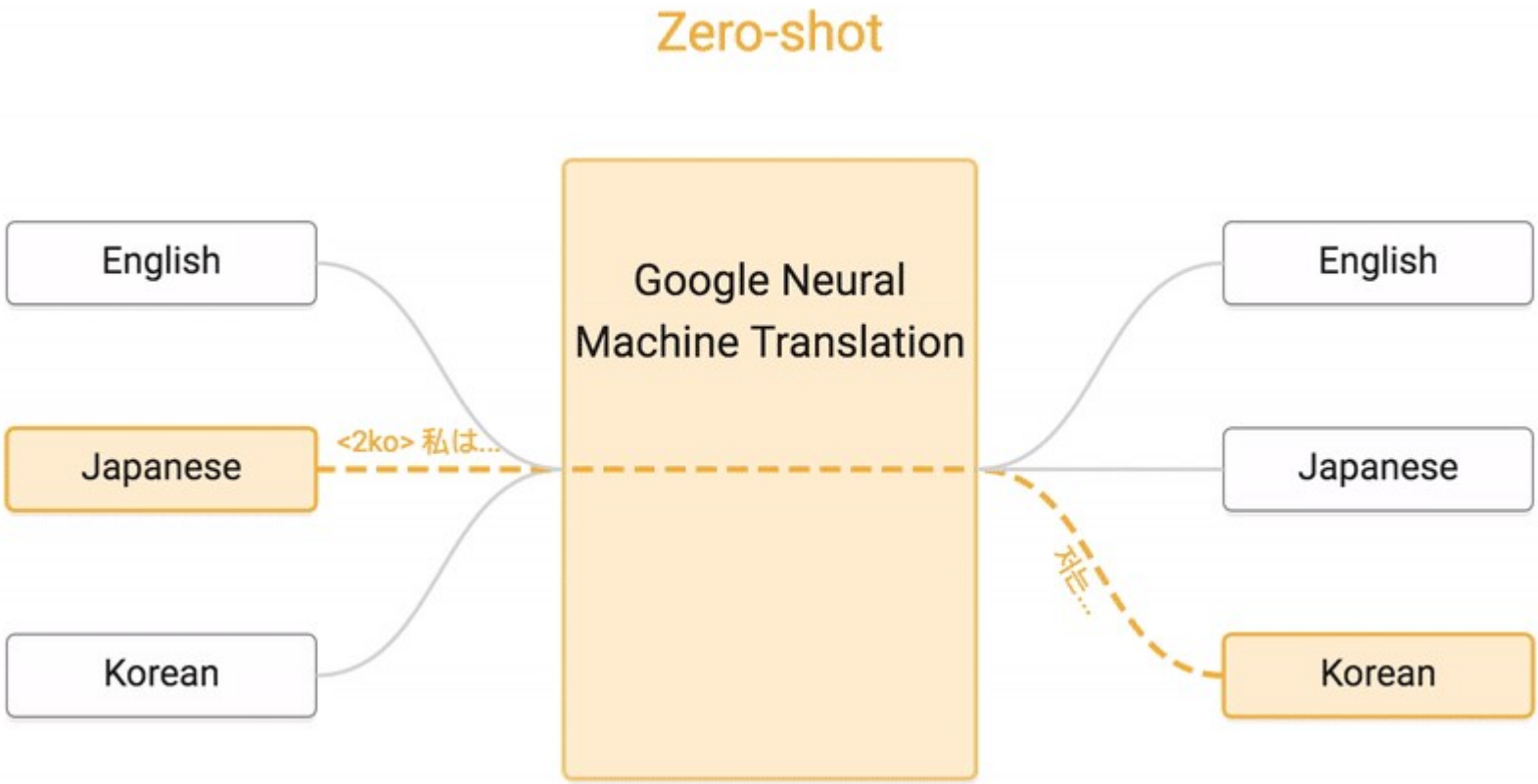
https://research.googleblog.com/2016/11/zero-shot-translation-with-googles.html

Multilingual NMT



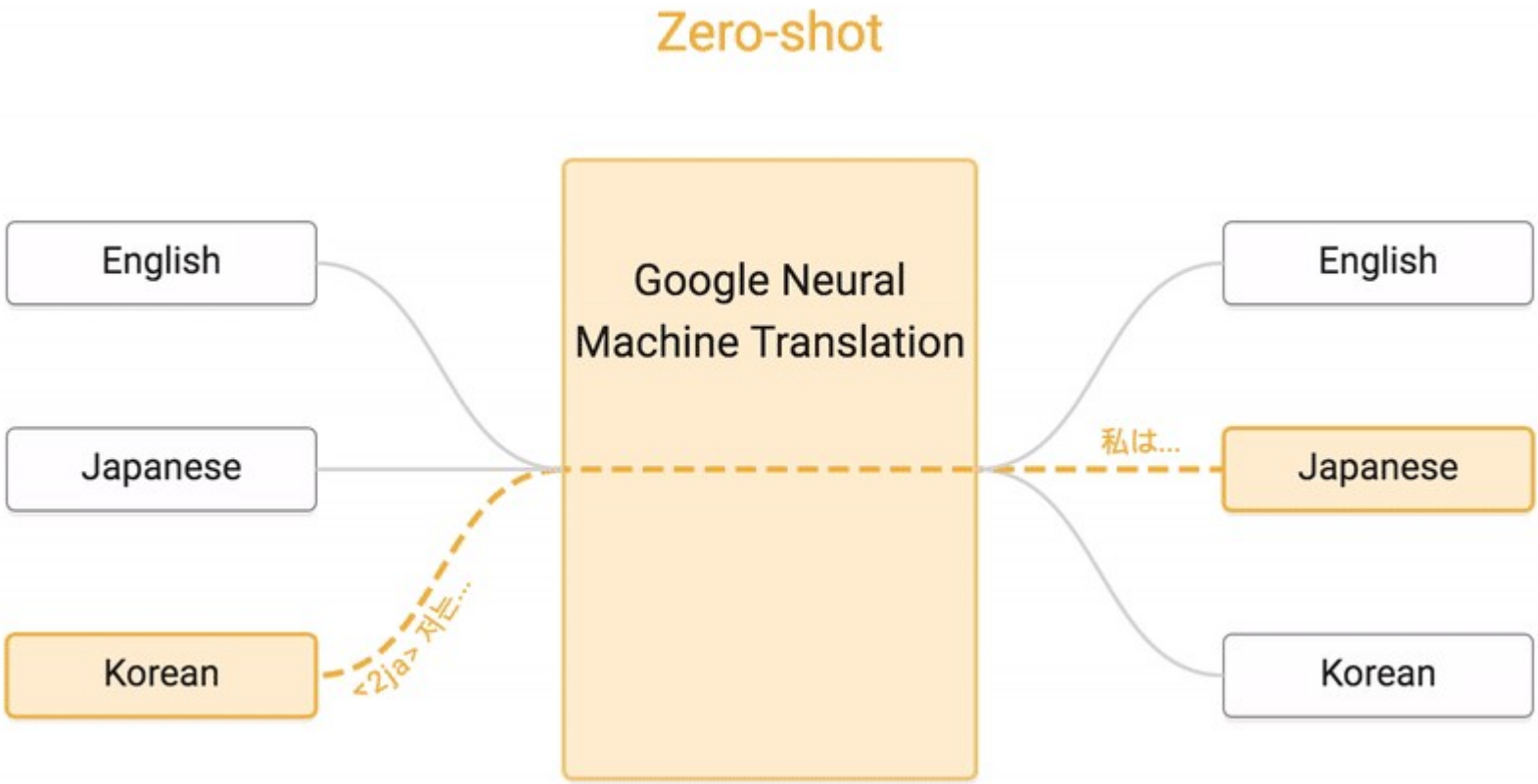
https://research.googleblog.com/2016/11/zero-shot-translation-with-googles.html

Multilingual NMT



https://research.googleblog.com/2016/11/zero-shot-translation-with-googles.html

Multilingual NMT

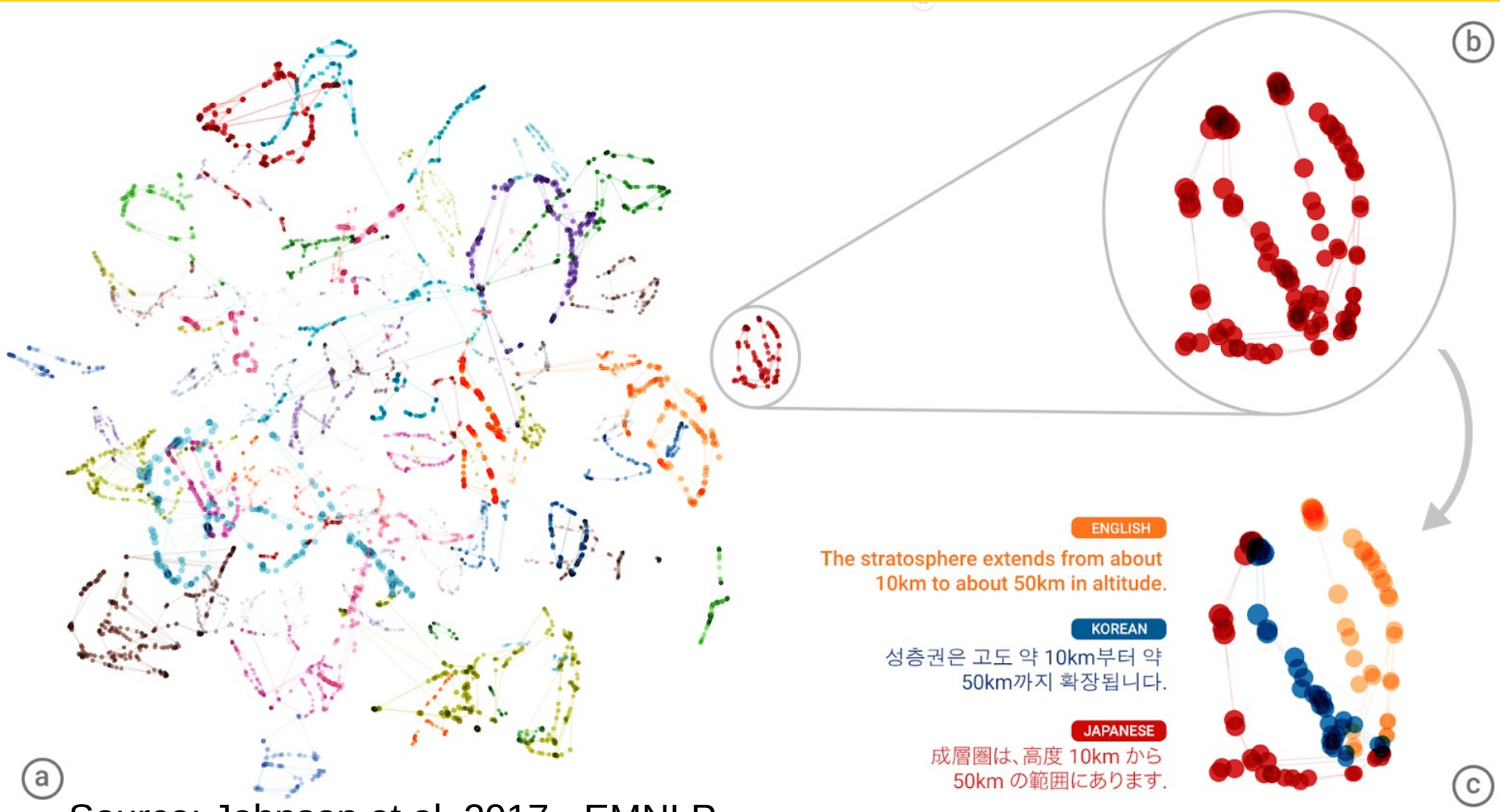


https://research.googleblog.com/2016/11/zero-shot-translation-with-googles.html

Multilingual NMT

- Same NMT system, no changes
  - Training: <2es> How are you? ¿Cómo estás?
  - Test: <2es> Great dog movie
  - Shared sentencepiece model: 32K
  - Train with all available parallel data:
    possibly over- or under-sampling some language pairs
  - More time to train than single language pair models
  - Larger batch sizes, slightly higher initial learning rate

Multilingual NMT: Interlingua (?)



Source: Johnson et al. 2017 - EMNLP

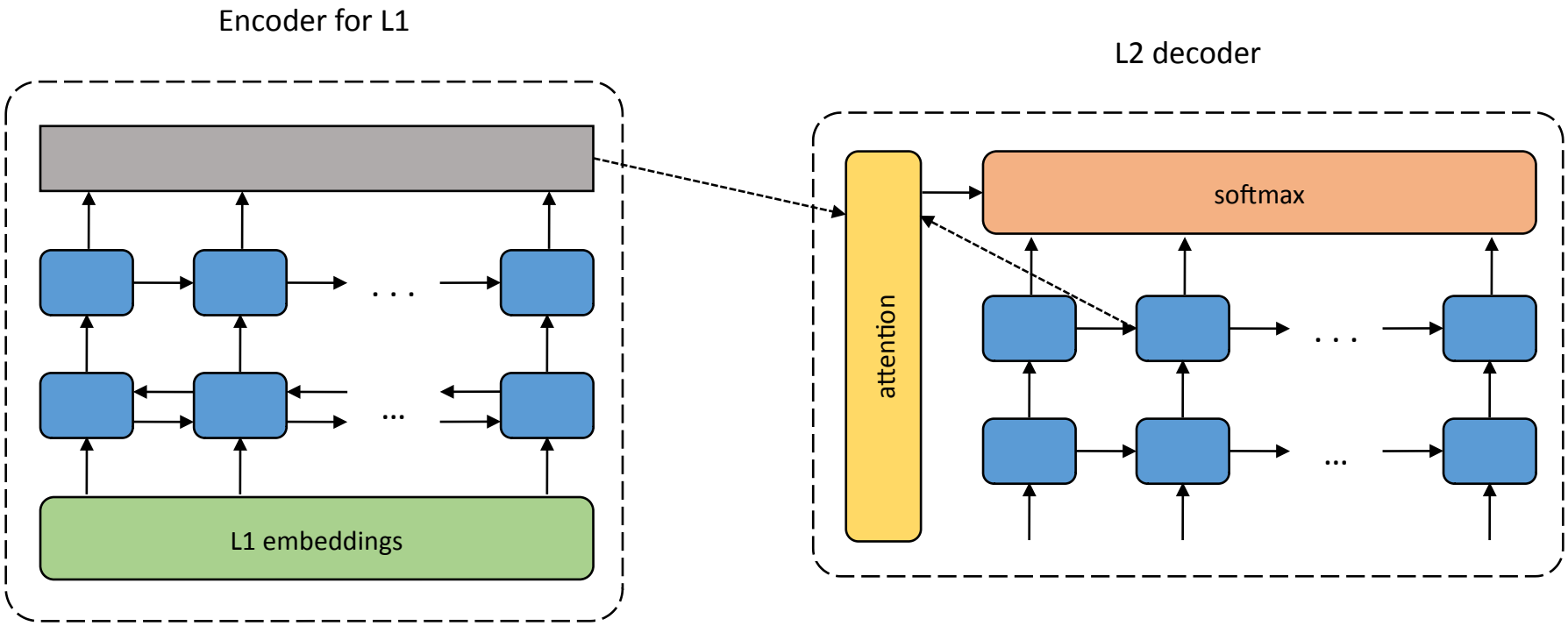
Plan for this session

- **Re-thinking seq2seq:**
  - Attention and memory
  - State of the art NMT: self-attention (transformers)
  - **Amazing things:**
    - Multilingual MT
    - MT without any bilingual data
- Evaluating sentence representations

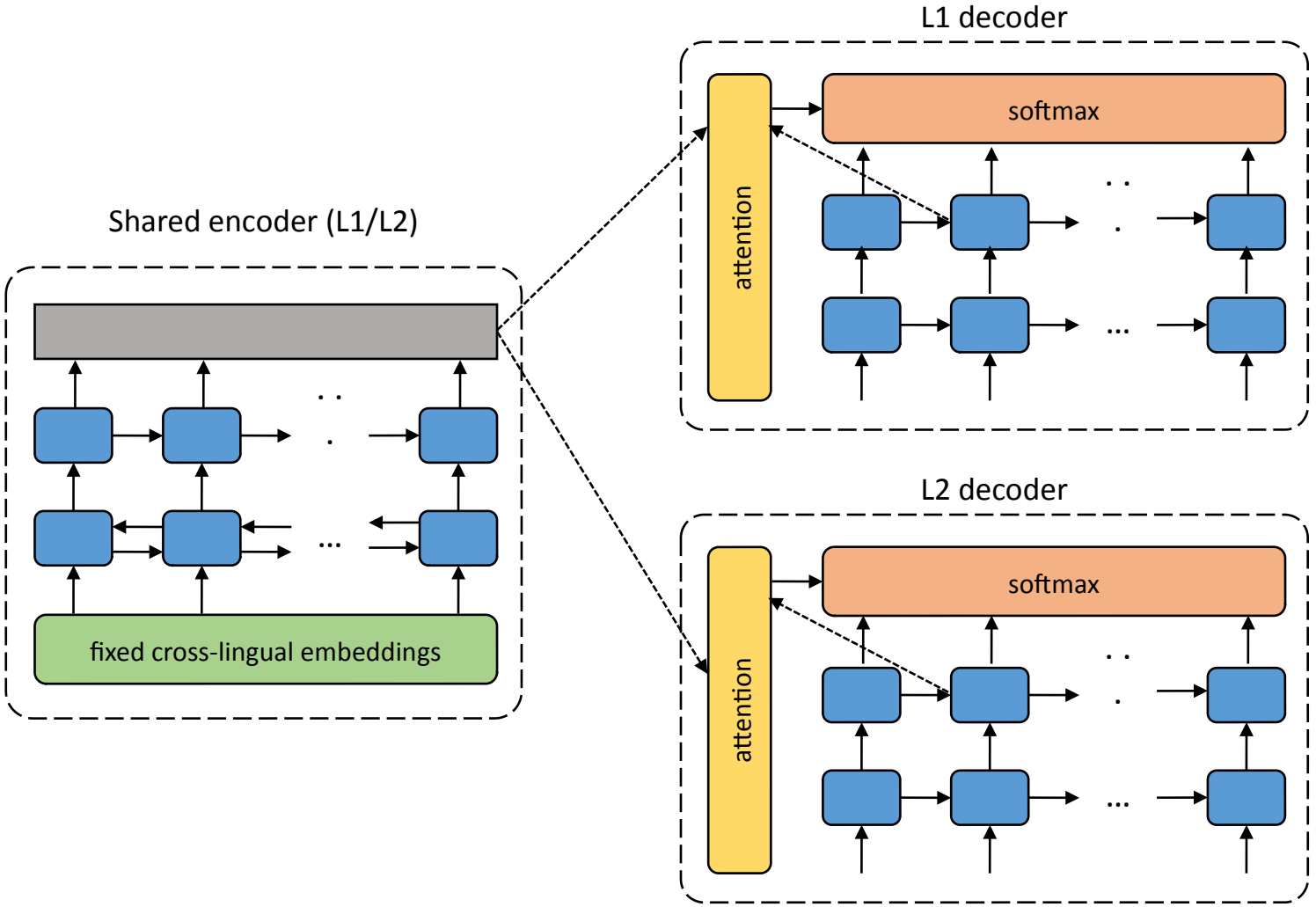
Unsupervised NMT

- Given:
  - Some books in Chinese
  - Some **other** books in Arabic
  - A person who does not know Chinese nor Arabic
- Can a person learn to translate from Chinese to Arabic or vice versa?
- A program first developed here can!

Unsupervised NMT



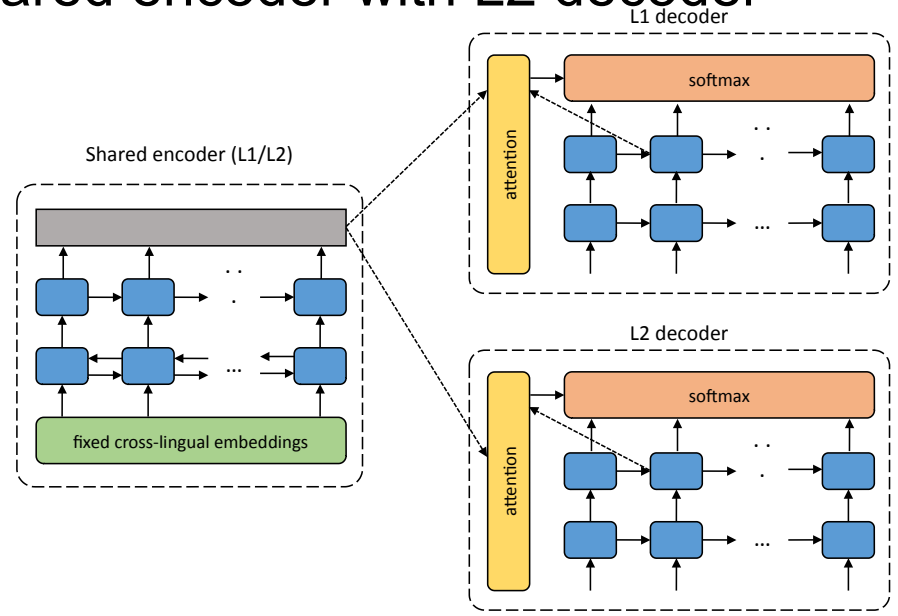
Unsupervised NMT



Unsupervised NMT

Previously: cross-lingual embeddings

- Train as autoencoder:
  - Given sentence in L1 train shared encoder with L1 decoder
  - Given sentence in L2 train shared encoder with L2 decoder
- Test:
  - Given sentence in L1 produce sentence in L2
  - Given sentence in L2 produce sentence in L1

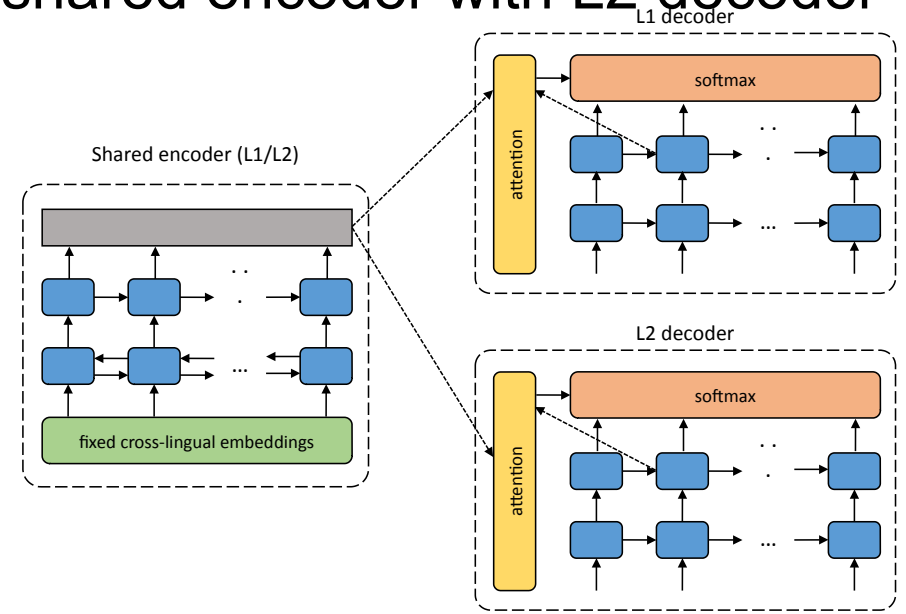


Unsupervised NMT

Previously: cross-lingual embeddings

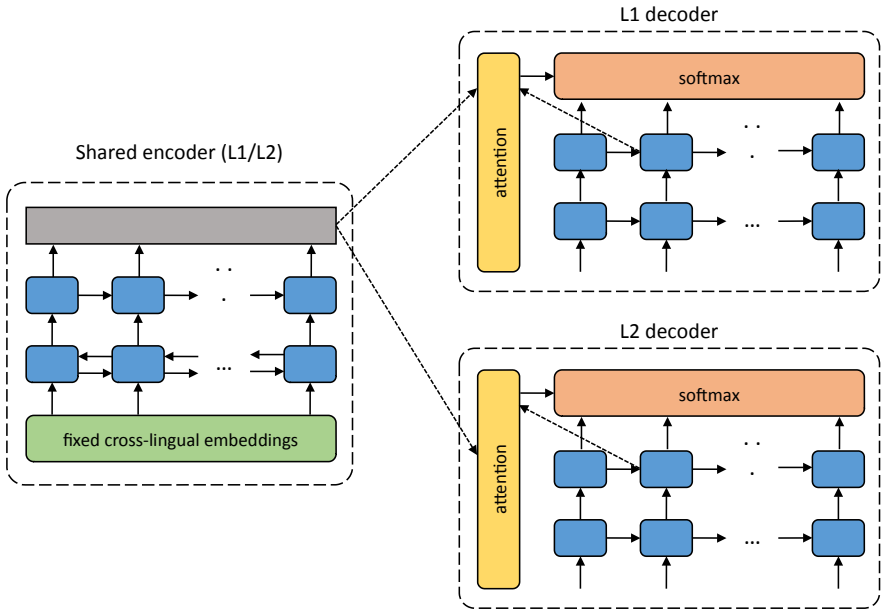
- Train as autoencoder:
  - Given sentence in L1 train shared encoder with L1 decoder
  - Given sentence in L2 train shared encoder with L2 decoder
- Test:
  - Given sentence in L1 produce sentence in L2
  - Given sentence in L2 produce sentence in L1

Fails!! Why?



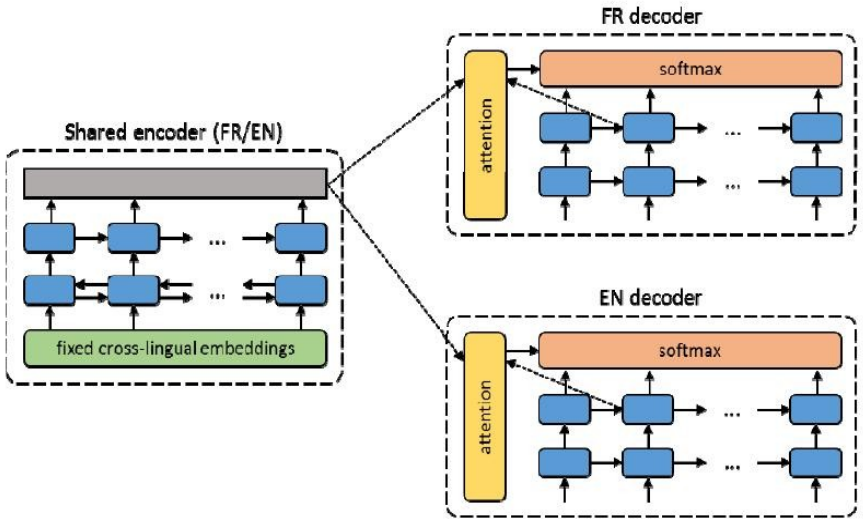
Unsupervised NMT

- More informative tasks:
  - Denoising autoencoder: introduce noise in input
  - Backtranslation: Given sentence in L1 translate to L2, and train shared encoder with L1 decoder to recover original sentence



Unsupervised NMT

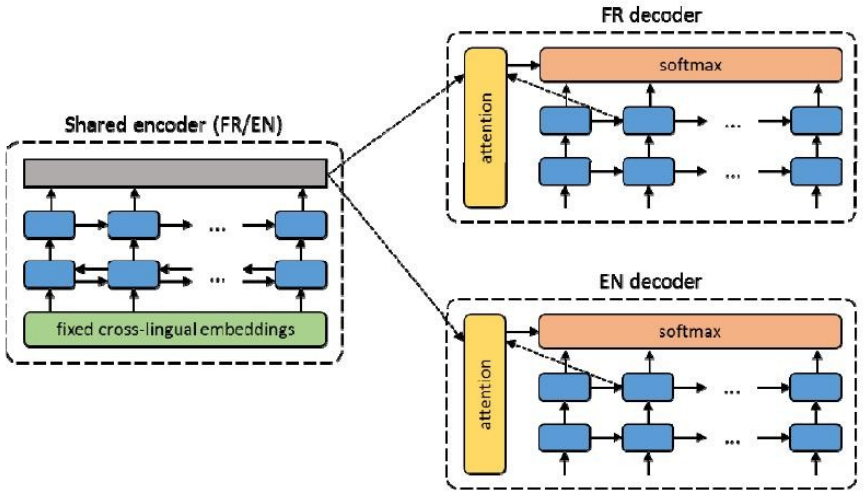
Training



Unsupervised NMT

Training

Une fusillade a eu lieu à l'aéroport international de Los Angeles.

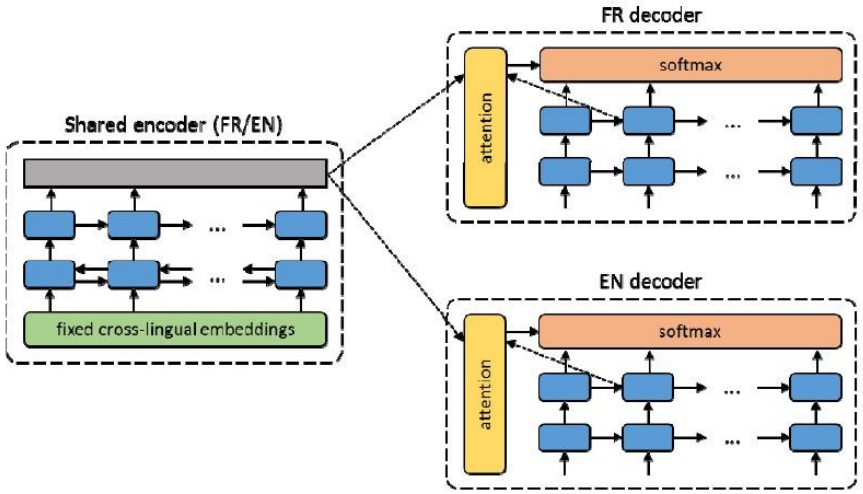


Unsupervised NMT

Training

- Supervised

Une fusillade a eu lieu à l'aéroport international de Los Angeles.

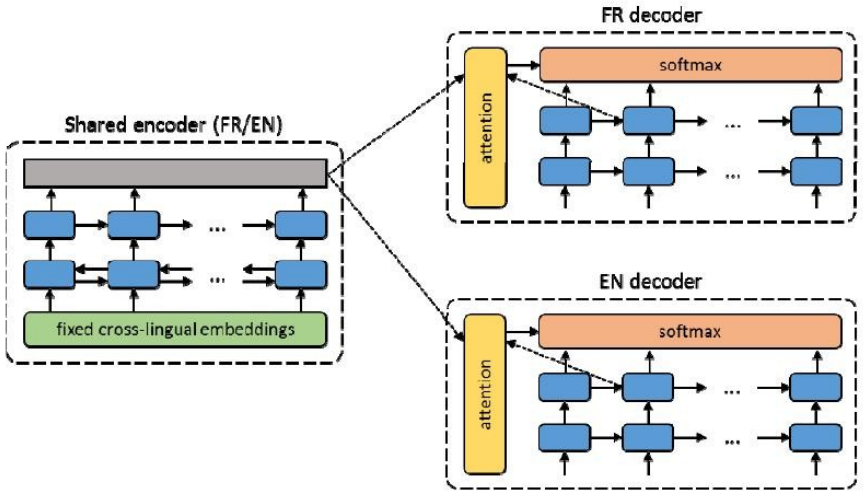


Unsupervised NMT

Training

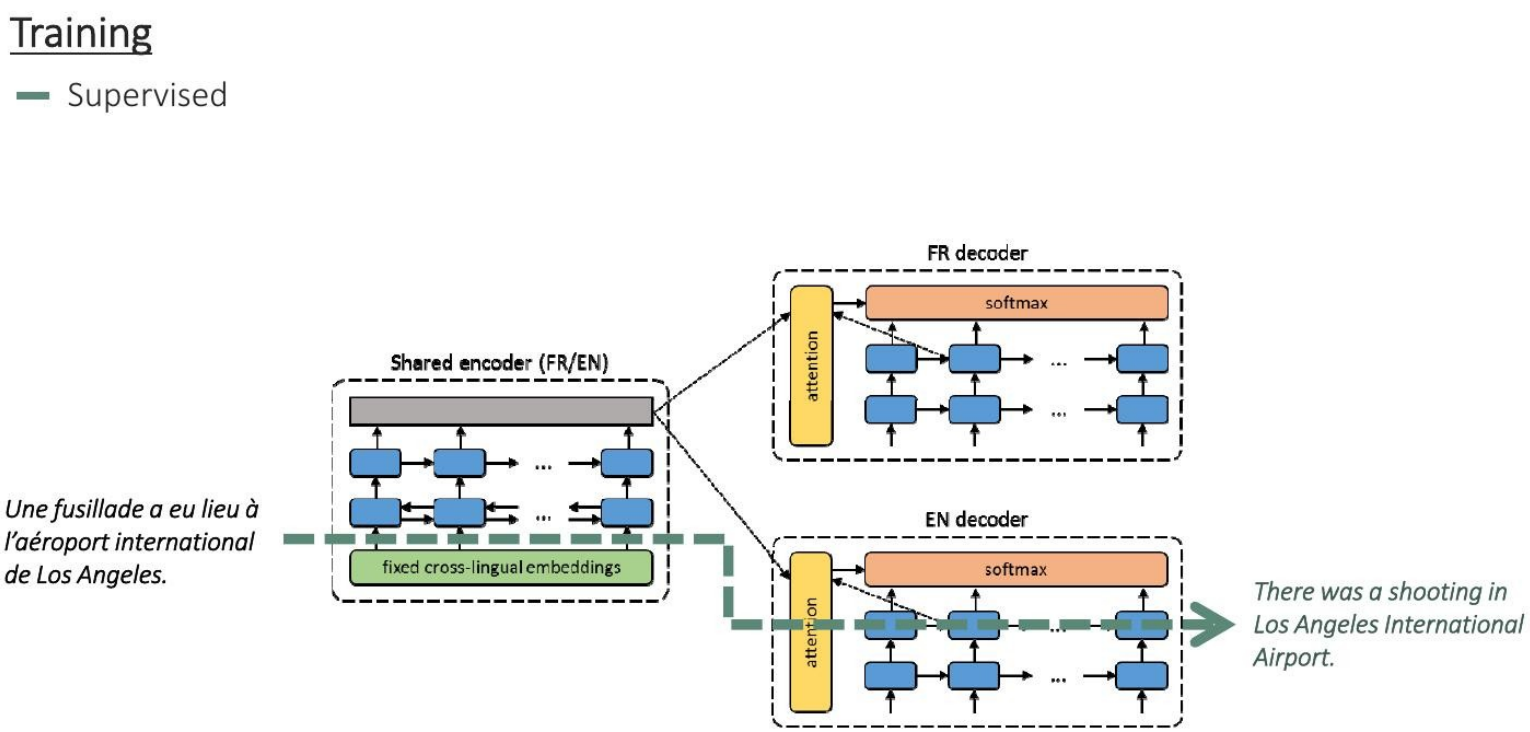
- Supervised

Une fusillade a eu lieu à l'aéroport international de Los Angeles.

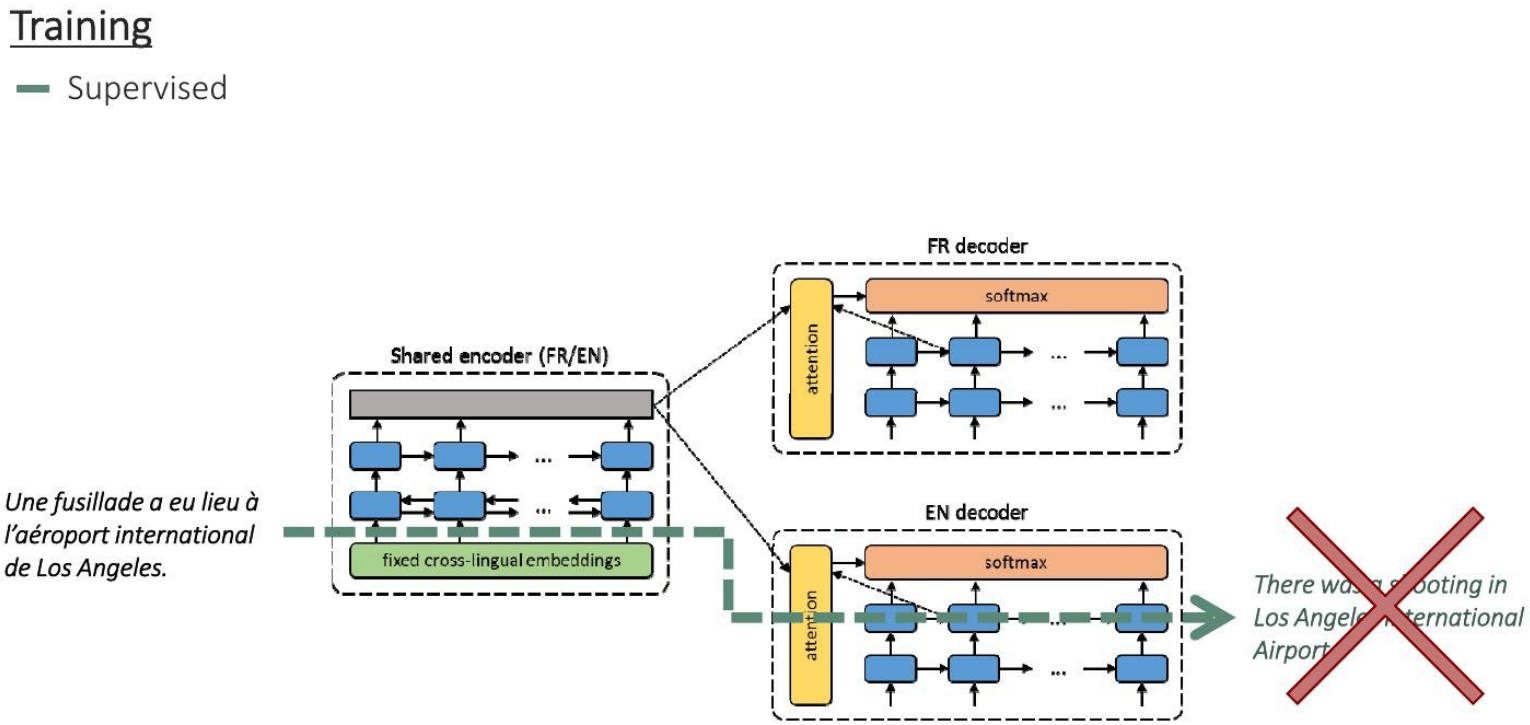


There was a shooting in Los Angeles International Airport.

Unsupervised NMT



Unsupervised NMT

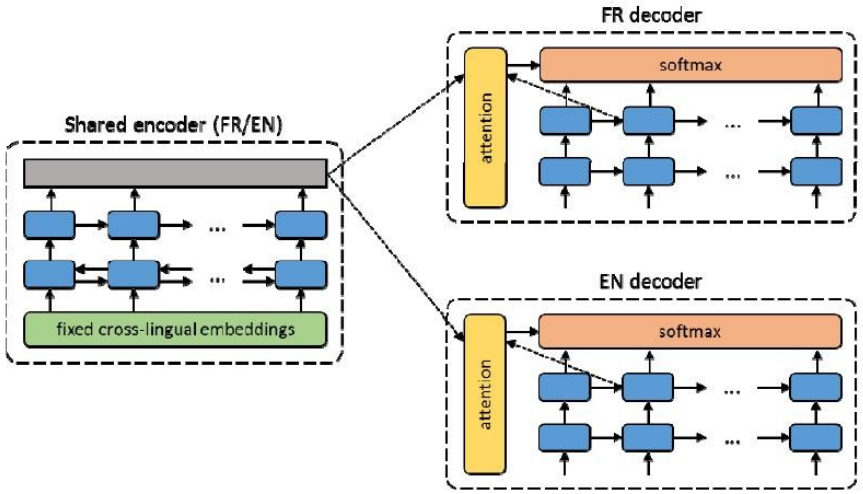


Unsupervised NMT

Training

- ~~Supervised~~

Une fusillade a eu lieu à l'aéroport international de Los Angeles.

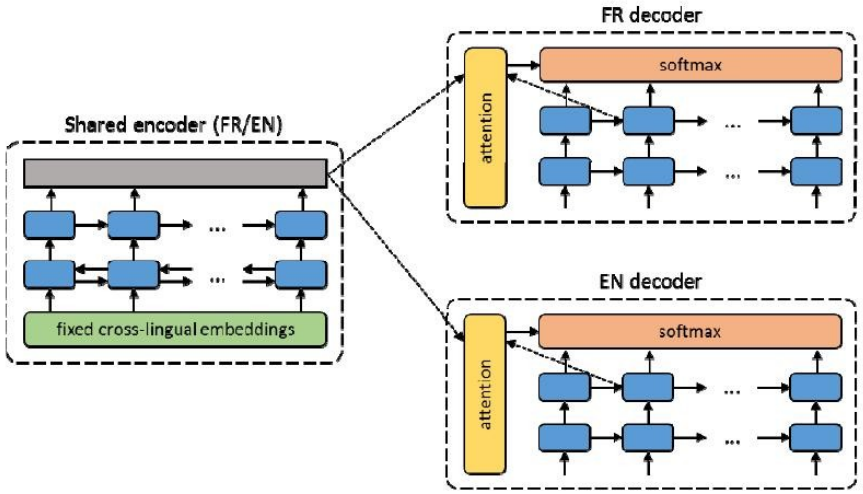


Unsupervised NMT

Training

- ~~Supervised~~
- Autoencoder

Une fusillade a eu lieu à l'aéroport international de Los Angeles.

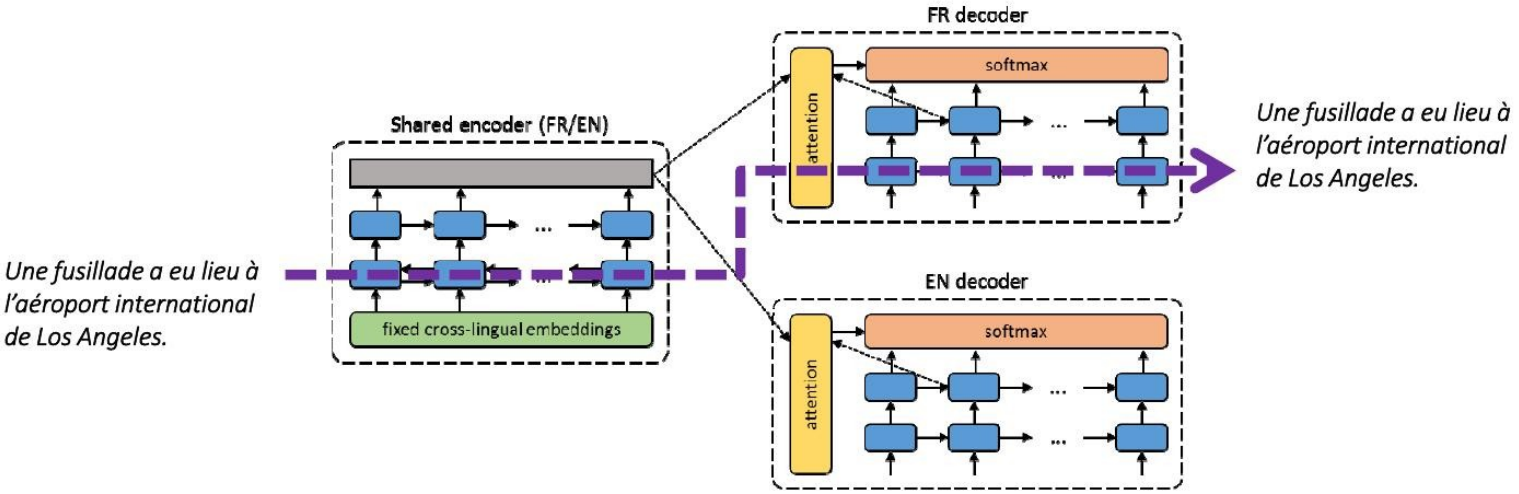


Une fusillade a eu lieu à l'aéroport international de Los Angeles.

Unsupervised NMT

Training

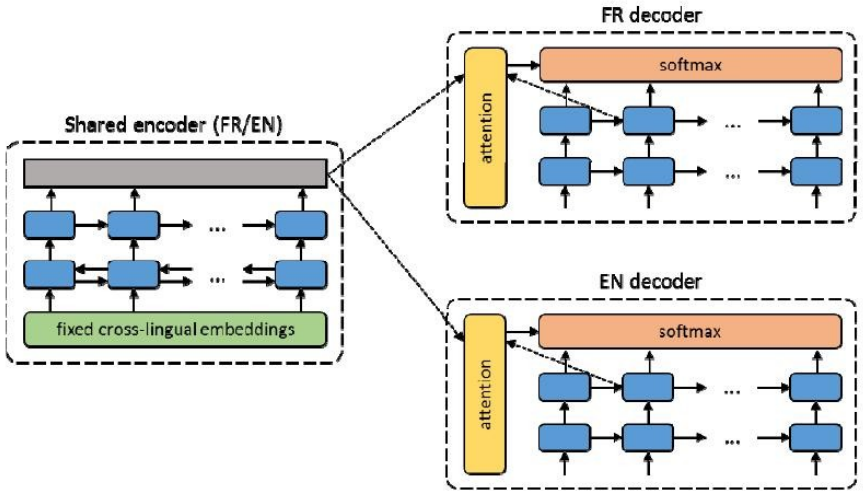
- ~~Supervised~~
- Autoencoder



Unsupervised NMT

Training

- ~~Supervised~~
- Denoising Autoencoder



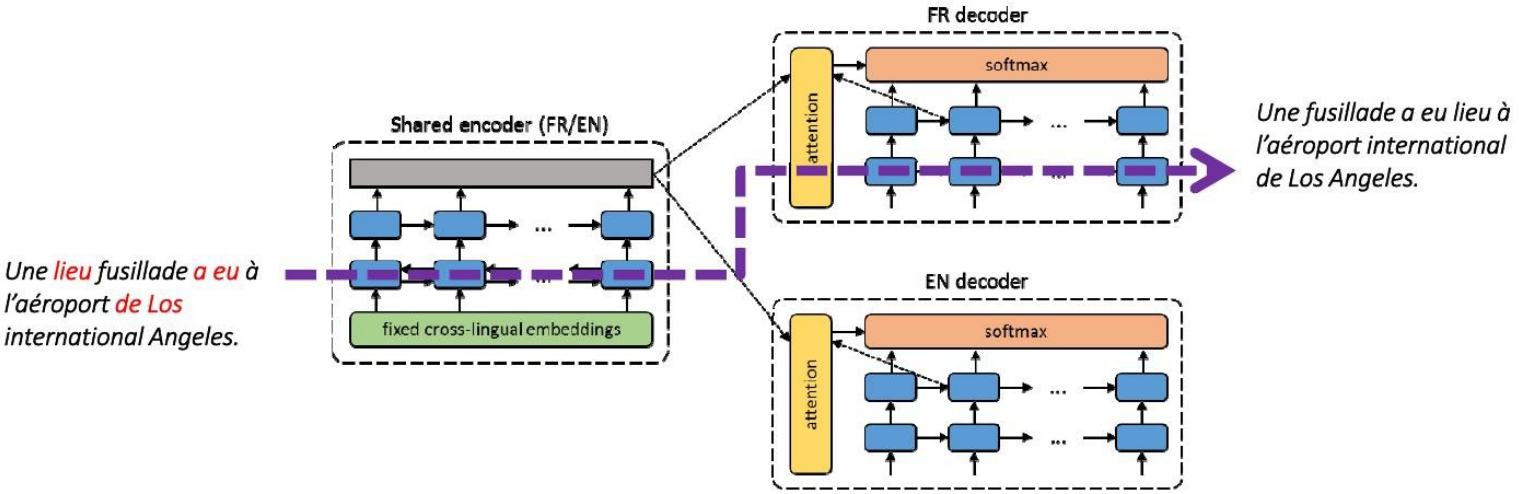
Une lieu fusillade a eu à l'aéroport de Los international Angeles.

Une fusillade a eu lieu à l'aéroport international de Los Angeles.

Unsupervised NMT

Training

- ~~Supervised~~
- Denoising Autoencoder

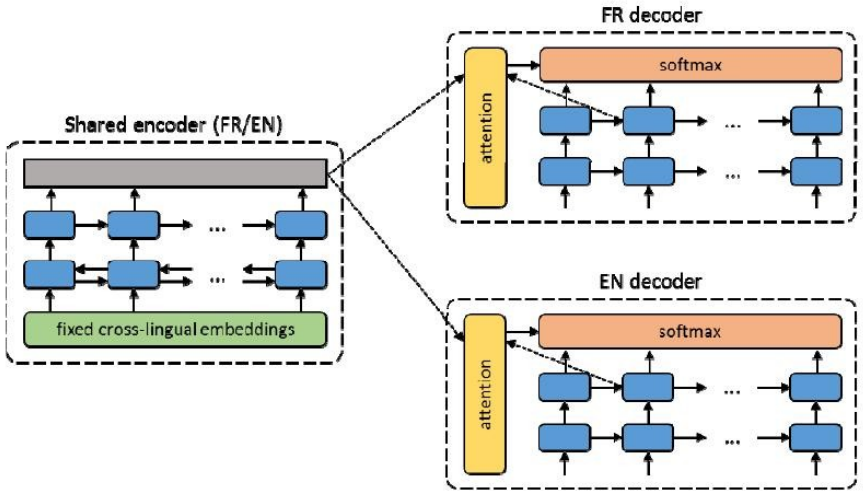


Unsupervised NMT

Training

- ~~Supervised~~
- Denoising Autoencoder

There a shooting was in Airport Los Angeles International.

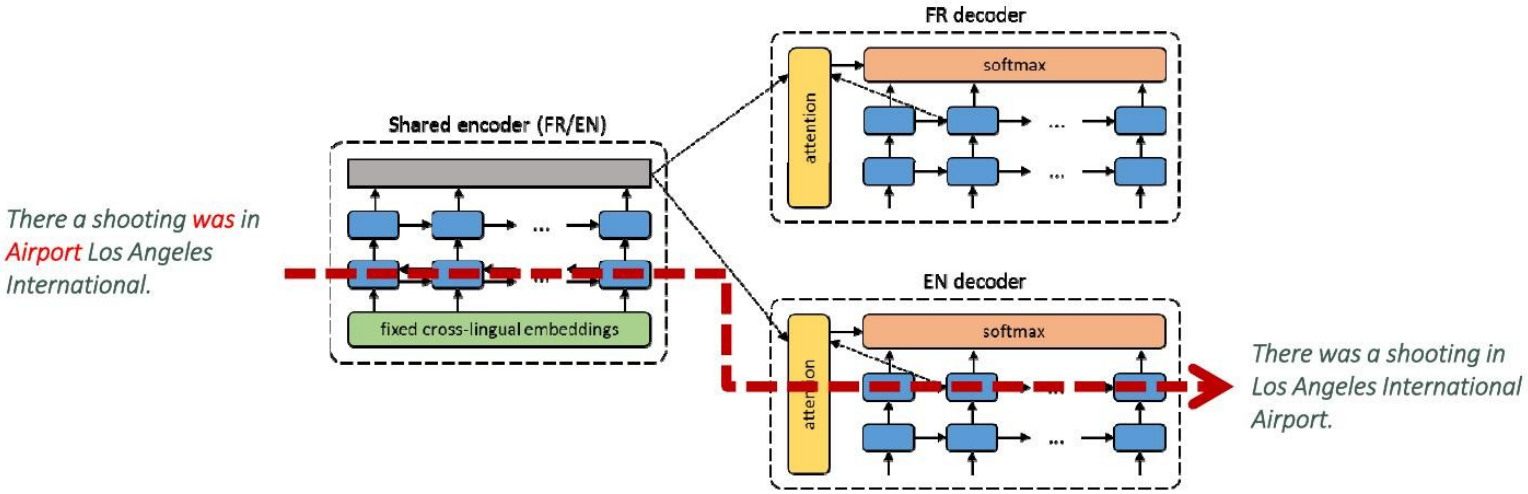


There was a shooting in Los Angeles International Airport.

Unsupervised NMT

Training

- ~~Supervised~~
- Denoising Autoencoder

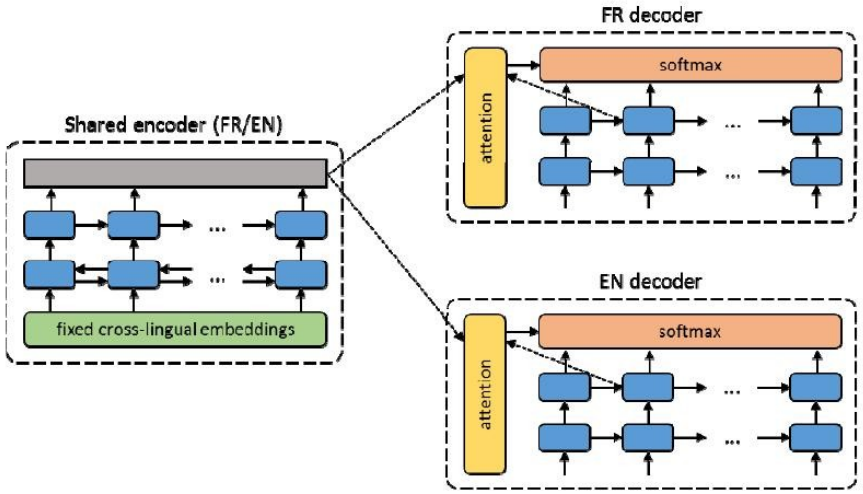


Unsupervised NMT

Training

- ~~Supervised~~
- Denoising
- Backtranslation

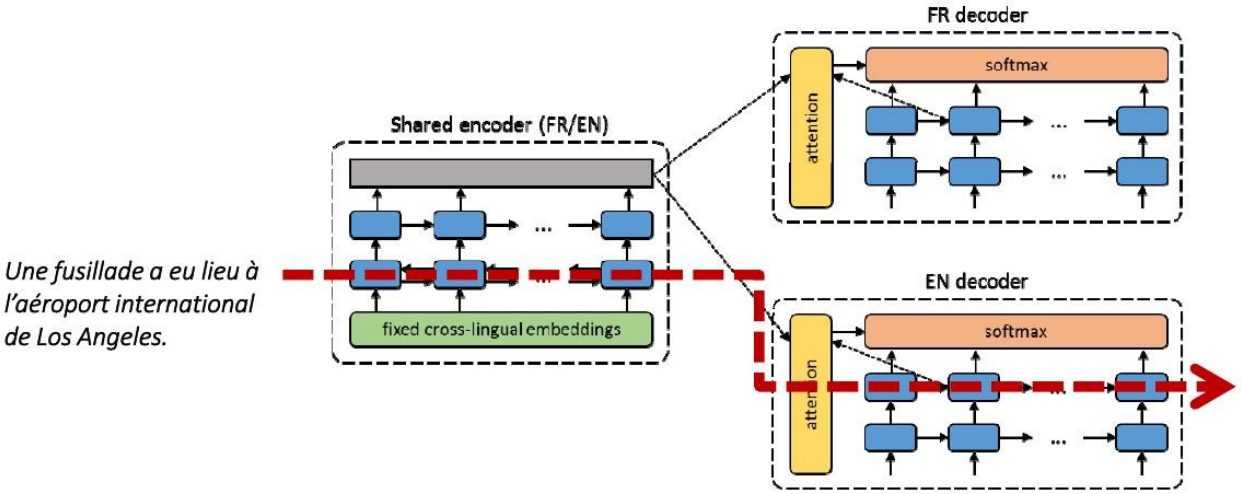
Une fusillade a eu lieu à l'aéroport international de Los Angeles.



Unsupervised NMT

Training

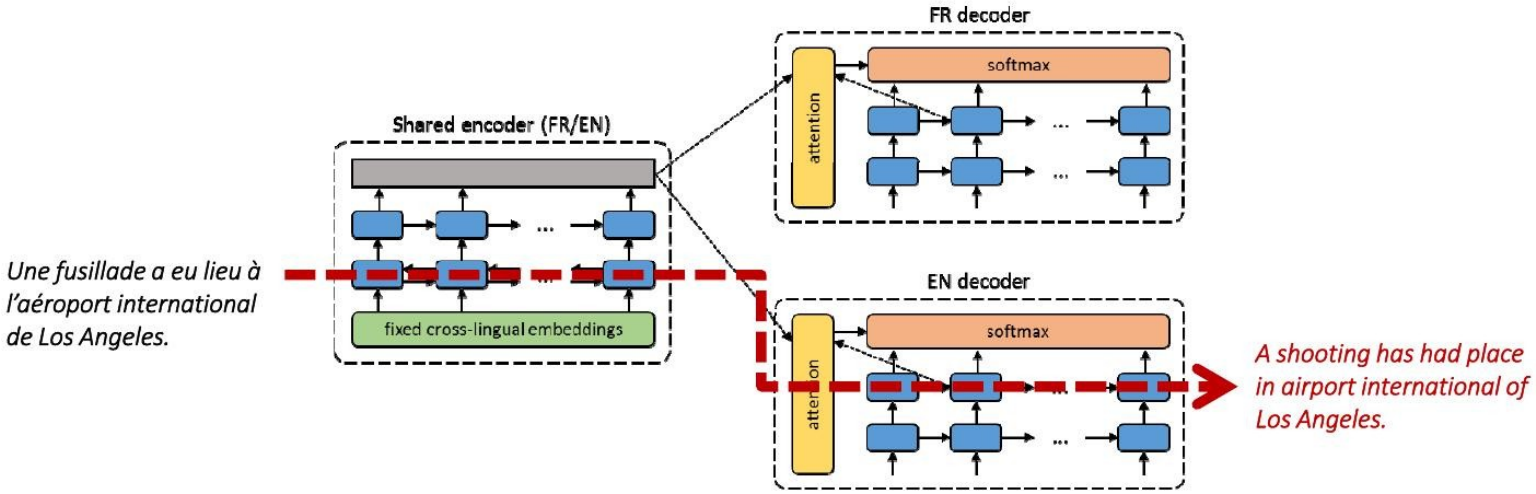
- ~~Supervised~~
- Denoising
- Backtranslation



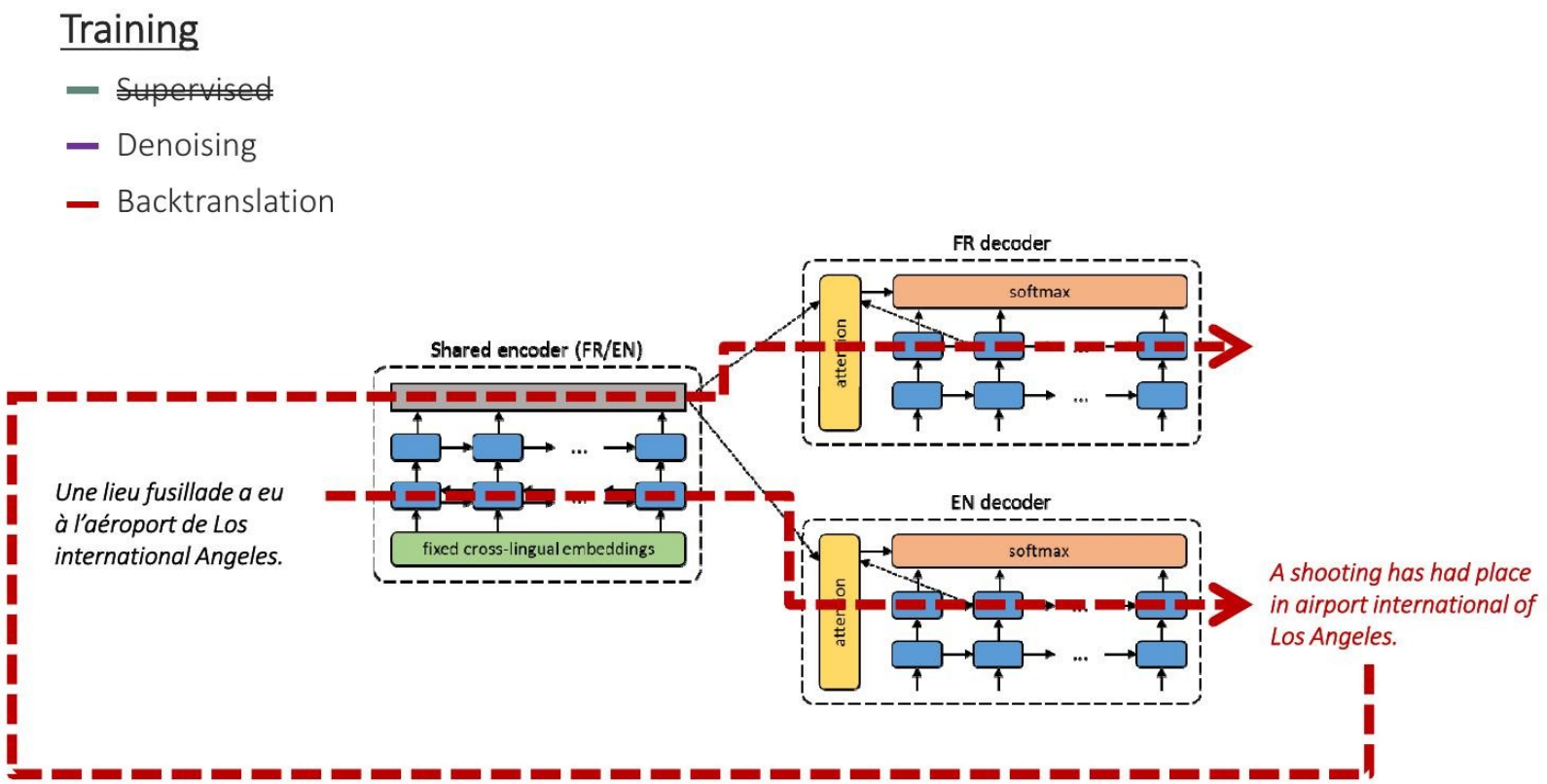
Unsupervised NMT

Training

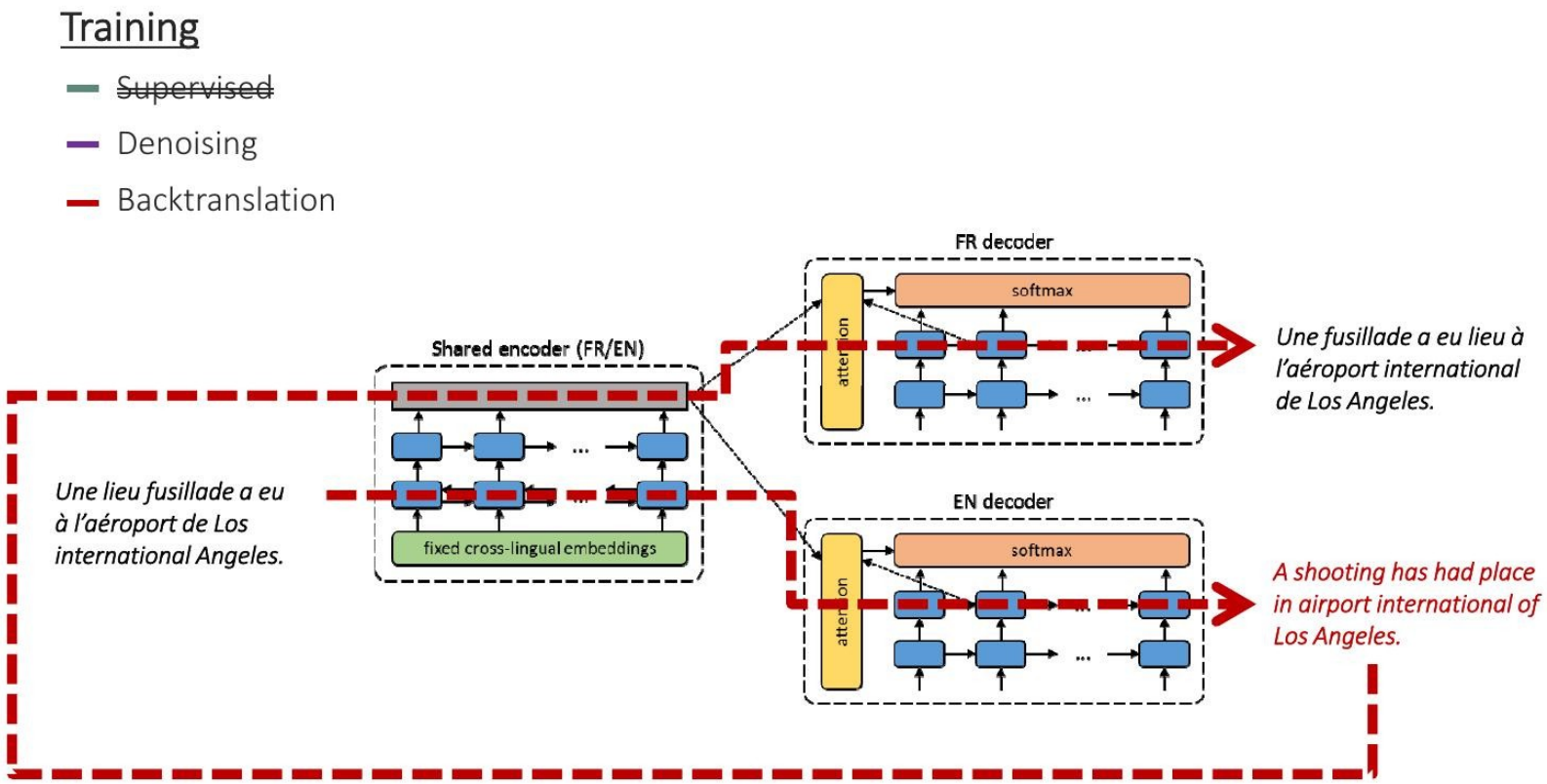
- ~~Supervised~~
- Denoising
- Backtranslation



Unsupervised NMT



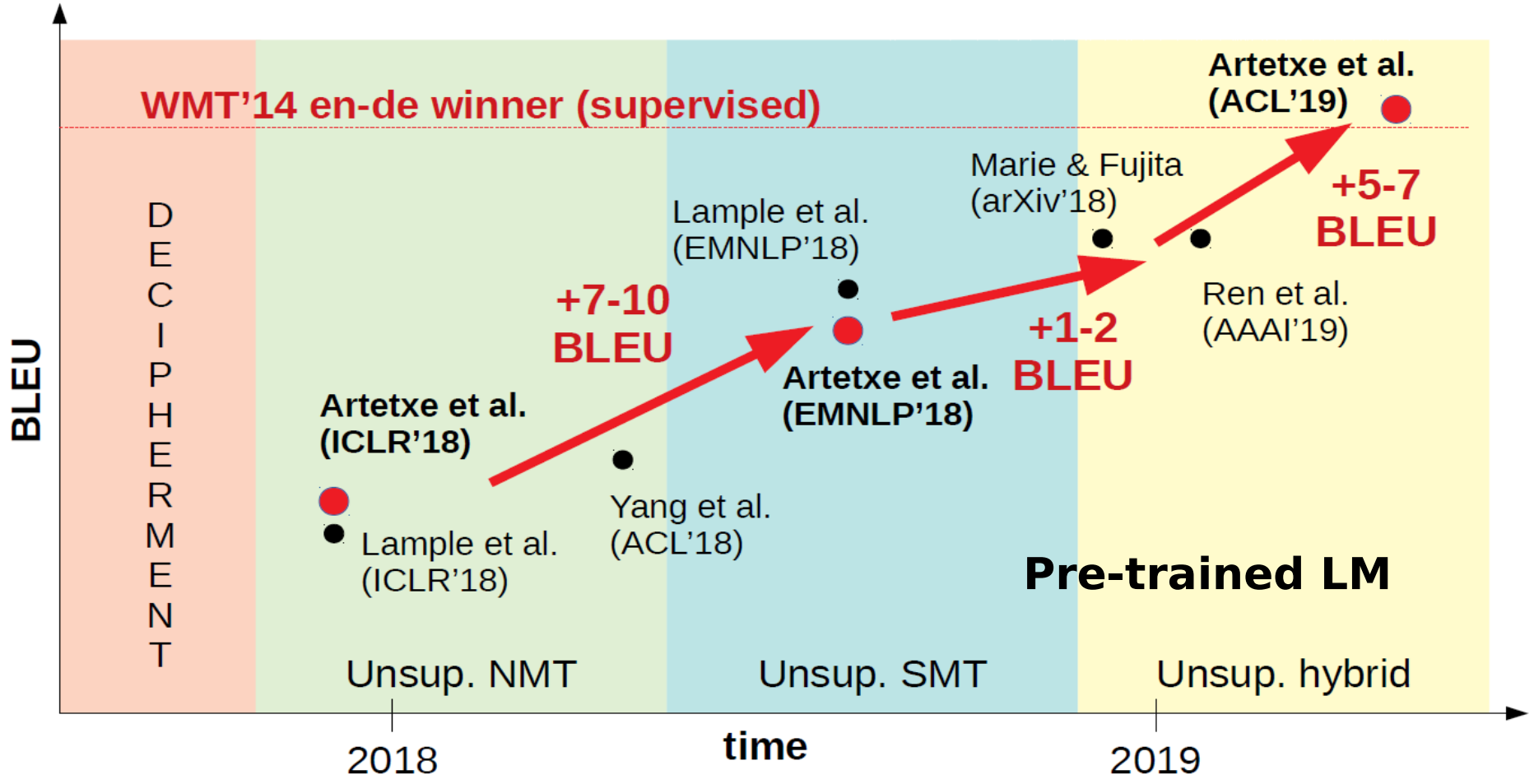
Unsupervised NMT



Unsupervised NMT

- More informative tasks:
  - Denoising autoencoder: introduce noise in input
  - Backtranslation: Given sentence in L1 translate to L2, and train shared encoder with L1 decoder to recover original sentence
- It works!
  - Artetxe et al. (2017): The first paper (by 1 day!) showing that it is possible to translate from one language to the other without bilingual resources
  - Further improvements (Artetxe et al. 2019)

Unsupervised NMT



UMT is at the level MT was at 2014

Plan for this session

- Re-thinking seq2seq:
  - Attention and memory
  - State of the art NMT: self-attention (transformers)
  - Amazing things:
    - Multilingual MT
    - MT without any bilingual data
- **Evaluating sentence representations**

Natural Language Inference

A soccer game with multiple males playing	Some men are playing sports
A man inspects the uniform of a figure in some East Asian country	The man is sleeping
A smiling customed woman is holding an umbrella	A happy woman in a fairy costume holds an umbrella

- What is the relation of the leftmost sentence with respect to the one in the right
  - Entailment
  - Contradiction
  - Neutral

Natural Language Inference

A soccer game with multiple males playing

Some men are playing sports

Natural Language Inference

A soccer game with multiple males playing

Some men are playing sports

Entailment!

Natural Language Inference

A man inspects the uniform of a figure in some East Asian country

The man is sleeping

Natural Language Inference

A man inspects the uniform of a figure in some East Asian country

The man is sleeping

Contradiction!

Natural Language Inference

A smiling customed woman is holding an umbrella

A happy woman in a fairy costume holds an umbrella

Natural Language Inference

A smiling customed woman is holding an umbrella

A happy woman in a fairy costume holds an umbrella

Neutral!

NLI

- Annotating a huge dataset
- Get alternative descriptions using mechanical turk
- 570K pairs
  - Real caption
  - Alternative descriptions

We will show you the caption for a photo. We will not show you the photo. Using only the caption and what you know about the world:

- Write one alternate caption that is **definitely** a **true** description of the photo. *Example: For the caption "Two dogs are running through a field." you could write "There are animals outdoors."*
- Write one alternate caption that **might be** a **true** description of the photo. *Example: For the caption "Two dogs are running through a field." you could write "Some puppies are running to catch a stick."*
- Write one alternate caption that is **definitely** a **false** description of the photo. *Example: For the caption "Two dogs are running through a field." you could write "The pets are sitting on a couch." This is different from the* maybe correct *category because it's impossible for the dogs to be both running and sitting.*

Figure 1: The instructions used on Mechanical Turk for data collection.

Natural Language Inference

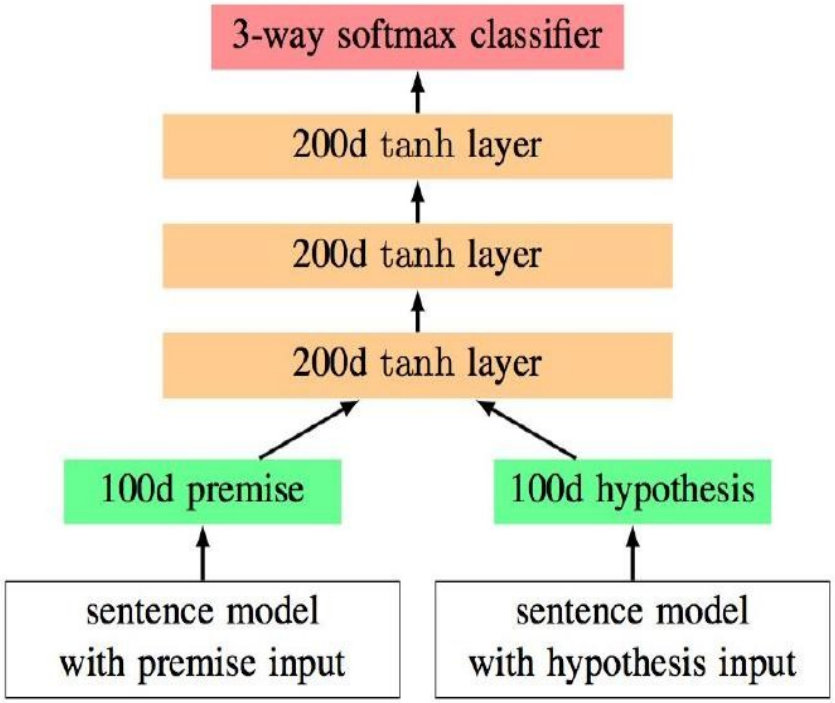
A man inspects the uniform of a figure in some East Asian country.	**contradiction** C C C C C	The man is sleeping
An older and younger man smiling.	**neutral** N N E N N	Two men are smiling and laughing at the cats playing on the floor.
A black race car starts up in front of a crowd of people.	**contradiction** C C C C C	A man is driving down a lonely road.
A soccer game with multiple males playing.	**entailment** E E E E E	Some men are playing a sport.
A smiling costumed woman is holding an umbrella.	**neutral** N N E C N	A happy woman in a fairy costume holds an umbrella.

Source: Bowman et al. 2015

Natural Language Inference

- Logistic regression with manual features
  - Overlap between two sentences
    - BLEU score
    - Length difference
    - Relative word overlap
  - Lexicalized features
    - One feature for unigrams and bigrams
    - One feature for each pair of unigrams across sentences
    - One feature for each pair of bigrams across sentences

Natural Language Inference



Source: Bowman et al. 2015

Test different sentence representations models e.g RNN, LSTM, ...

Natural Language Inference

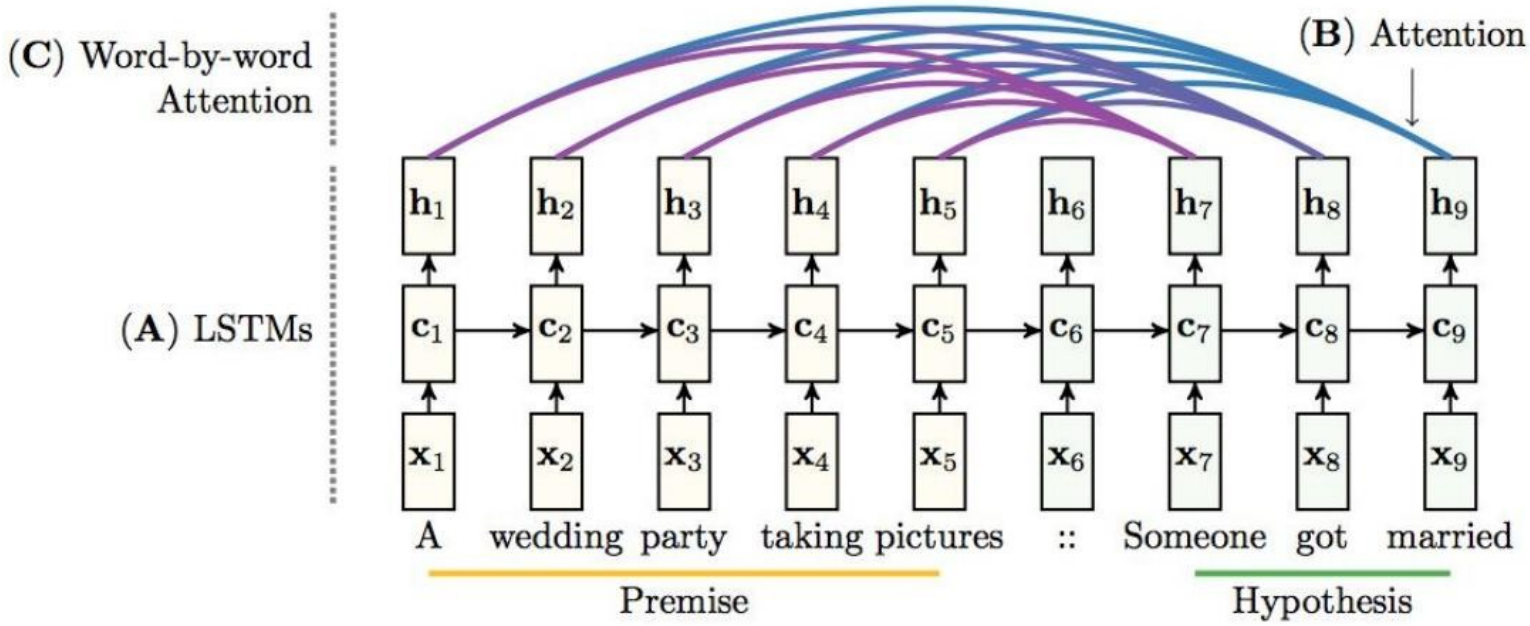
	accuracy
Logistic Regression with manual features	78.2
Bag of words	75.3
RNN	73.1
LSTM	80.6

Source: Bowman et al. 2015, SNLI website

Natural Language Inference

Soft Attention

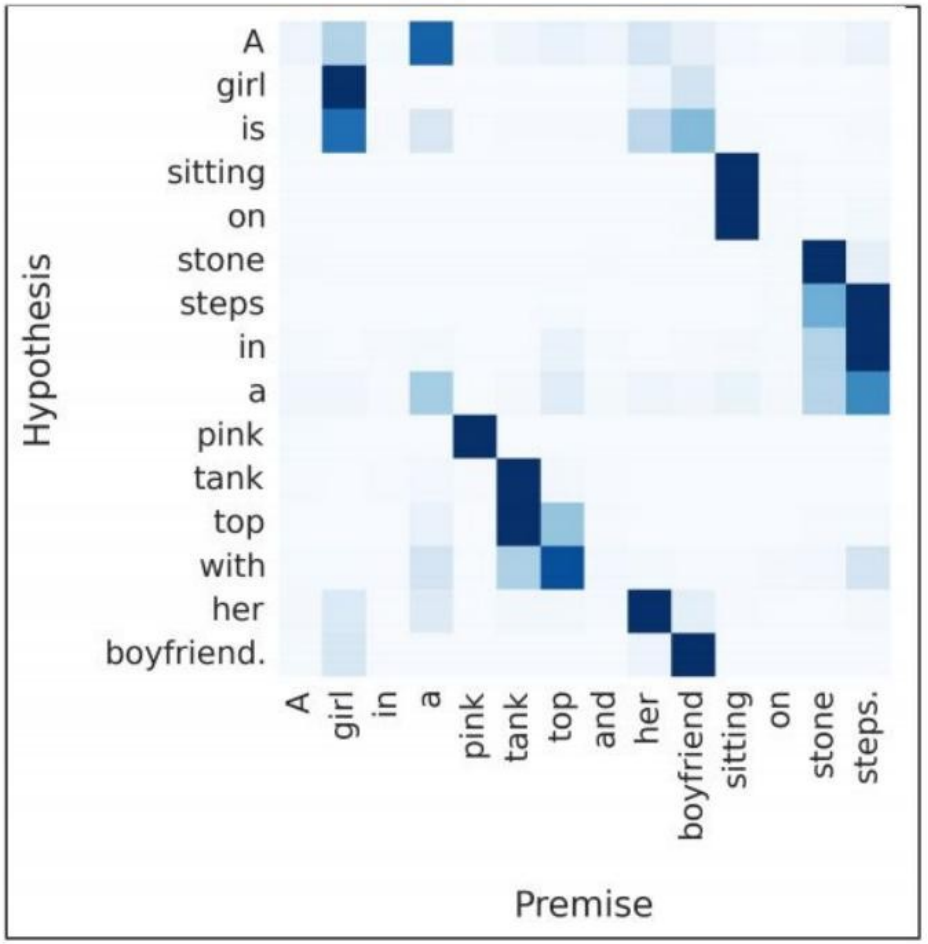
Rocktäschel, Grefenstette, Hermann, Kočiský, & Blunsom, ICLR '16



Natural Language Inference

Attention

makes sense



Source: Röcktaschel et al. 2016 - ICLR

Natural Language Inference

	accuracy
Logistic Regression with manual features	78.2
Bag of words	75.3
RNN	73.1
LSTM	80.6
LSTM+attention	83.5
Leader	92.1

Source: Bowman et al. 2015, SNLI website, Röcktaschel et al. 2016

Semantic Textual Similarity

Set of datasets 2012-2017 measuring the similarity of two sentences: a continous score between 0 and 5

Source: Agirre et al. 2017

Semantic T

Set of datasets 2012-2017 measuring the similarity of two sentences: a continous score between 0 and 5

Source: Agirre et al. 2017

5	*The two sentences are completely equivalent, as they mean the same thing.*
	The bird is bathing in the sink. Birdie is washing itself in the water basin.
4	*The two sentences are mostly equivalent, but some unimportant details differ.*
	Two boys on a couch are playing video games. Two boys are playing a video game.
3	*The two sentences are roughly equivalent, but some important information differs/missing.*
	John said he is considered a witness but not a suspect. "He is not a suspect anymore." John said.
2	*The two sentences are not equivalent, but share some details.*
	They flew out of the nest in groups. They flew into the nest together.
1	*The two sentences are not equivalent, but are on the same topic.*
	The woman is playing the violin. The young lady enjoys listening to the guitar.
0	*The two sentences are completely dissimilar.*
	The black dog is running through the snow. A race car driver is driving his car through the mud.

Evaluating (sentence) representations

- SNLI, MNLI, XNLI
  https://www.nyu.edu/projects/bowman/multinli/
- STS http://ixa2.si.ehu.es/stswiki/index.php/STSbenchmark
- Senteval https://research.fb.com/downloads/senteval/
- GLUE https://gluebenchmark.com/
  - Pre-trained transformer LM and seq2seq (e.g. BERT, T5)
  - State of the art (Jan 2021): 90.8 (Human 87)
- SuperGLUE https://super.gluebenchmark.com/
  - State of the art (Jan 2021): 90.3 (Human 89.8)

Attention and transformers in tf

- tf.keras.layers.Attention (dot product, Luong attention)
  - Same as our attention, but no W parameter matrix
- tf.keras.layers.AdditiveAttention (Bahdanu attention)
- tf.keras.layers.MultiHeadAttention (as in transformers)

- https://huggingface.co/transformers/ (code and pre-trained models)

Tutorials

- Tensorflow: neural machine translation with attention
- Tensorflow: transformer model for language understanding
- Keras: Text classification with transformer
- Huggingface tutorials for their transformer library (see above)

THANKS!

Acknowledgements:

- Overall slides: Sam Bowman, Kyunghyun Cho (NYU), Chris Manning and Richard Socher (Stanford)

Other resources:

- https://github.com/pytorch/fairseq (Pytorch)

All source url's listed in the slides.