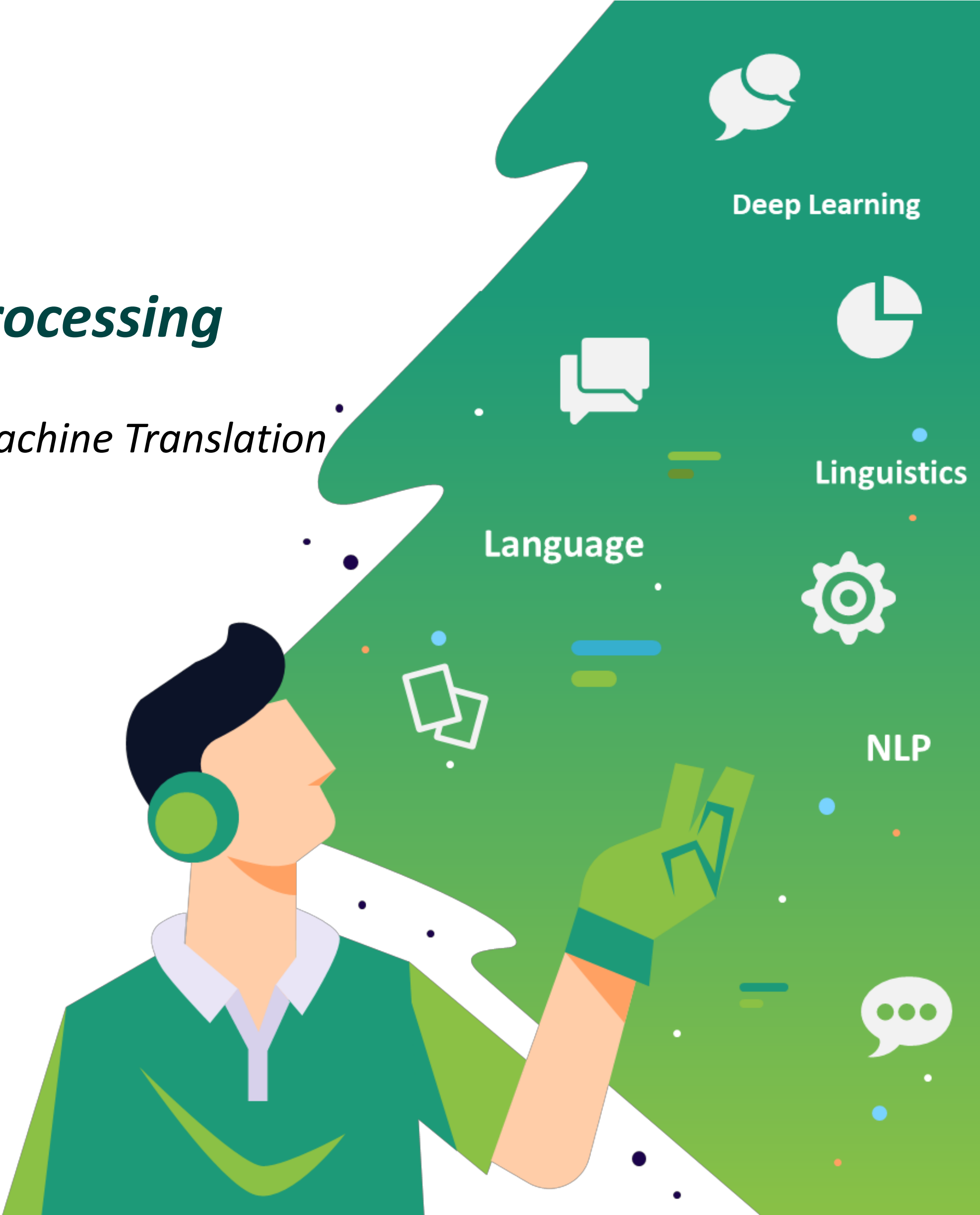
COMP5046
Natural Language Processing

Lecture 11: Advanced NLP: Machine Translation and Transformer



Dr. Caren Han

Semester 1, 2022
School of Computer Science,
University of Sydney

0 LECTURE PLAN

Lecture 11: Machine Translation and Transformer

1. Machine Translation
2. Statistical Machine Translation
3. Neural Machine Translation
4. Attention and Transformer for MT
5. The Rise of the Pre-trained Model

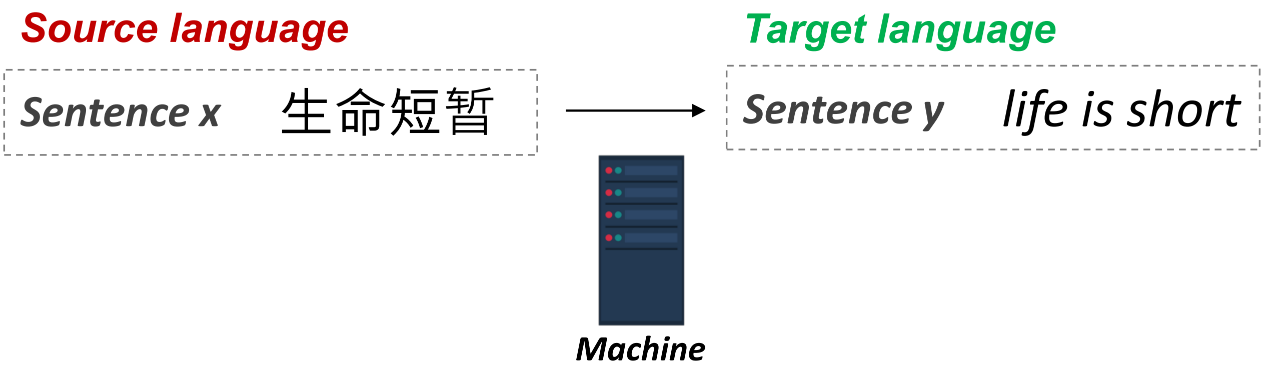
0 Assignment 2 Specification

1 What is Machine Translation?

1 Machine Translation

Machine Translation

*“translate a sentence x from one language (**the source language**) to a sentence y in another language (**the target language**).”*



Source language

Sentence x 生命短暂

Target language

Sentence y life is short

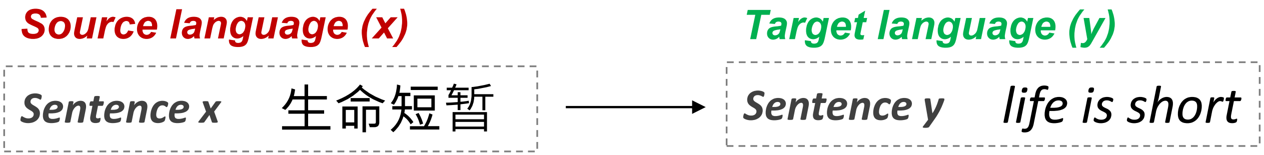
Machine

2 Statistical Machine Translation

2 Statistical Machine Translation

Statistical Machine Translation

*"Learning a **probabilistic model** from data"*



Source language (x)

Sentence x 生命短暂

Target language (y)

Sentence y *life is short*

Best translation?

$$\text{argmax}_y P(y|x)$$

How to learn translation model $P(x|y)$ *?*

2 Statistical Machine Translation

Statistical Machine Translation

“Learning a ***probabilistic model*** *from data”*

Source language (x)

Sentence x 生命短暂

→

Target language (y)

Sentence y *life is short*

Best translation?

$$\operatorname{argmax}_y P(y|x)$$

Bayes Rule

$$\operatorname{argmax}_y \underbrace{P(x|y)}_{\text{Translation Model (fidelity)}} \underbrace{P(y)}_{\text{Language Model (fluency)}}$$

Translation Model (fidelity)

Models how words and phrases should be translated

Language Model (fluency)

Models to write good English

2 Statistical Machine Translation

Statistical Machine Translation

*"Learning a **probabilistic model** from data"*

Source language (x)

Sentence x 生命短暂

→

Target language (y)

Sentence y *life is short*

Best translation?

$$\operatorname{argmax}_y \underline{P(y|x)}$$

Bayes Rule

$$\operatorname{argmax}_y \underline{P(x|y)}\,\underline{P(y)}$$

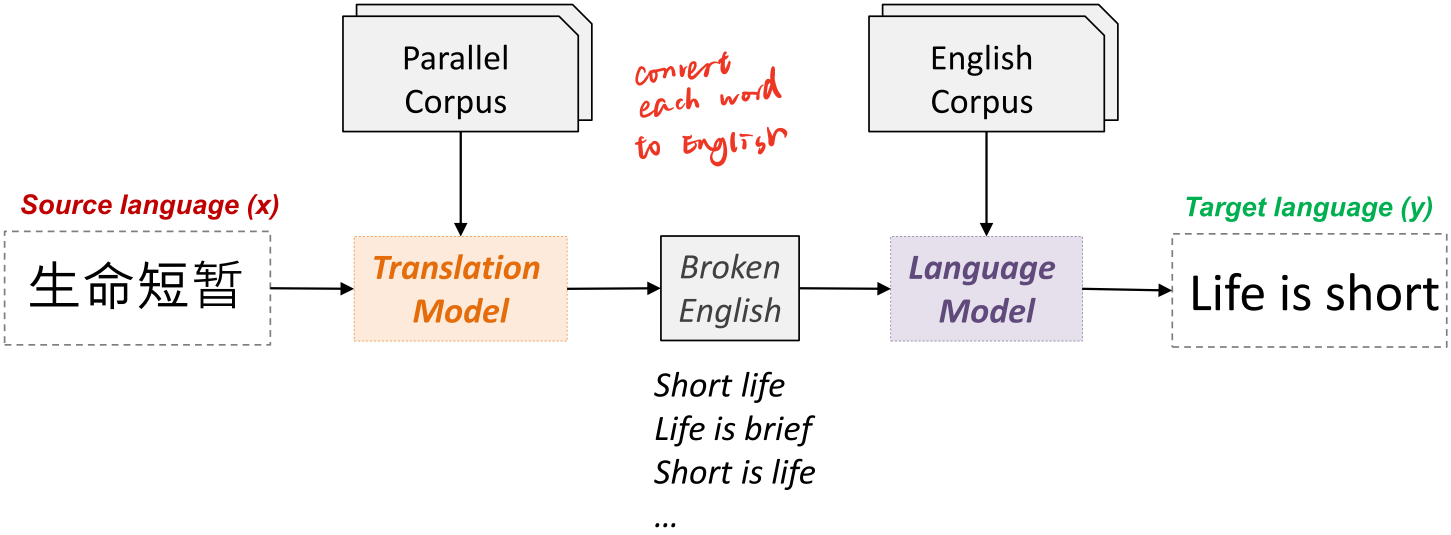
Translation Model (fidelity)
*Learnt from **parallel data***

x → y

Language Model (fluency)
*Learnt from **monolingual data***

2 Statistical Machine Translation

How to learn translation model with *parallel corpus*?



convert each word to English

Short life
Life is brief
Short is life
…

Bayes Rule

$$\operatorname{argmax}_y \underline{P(x|y)}\,\underline{P(y)}$$

Translation Model (fidelity)
Learnt from ***parallel data***

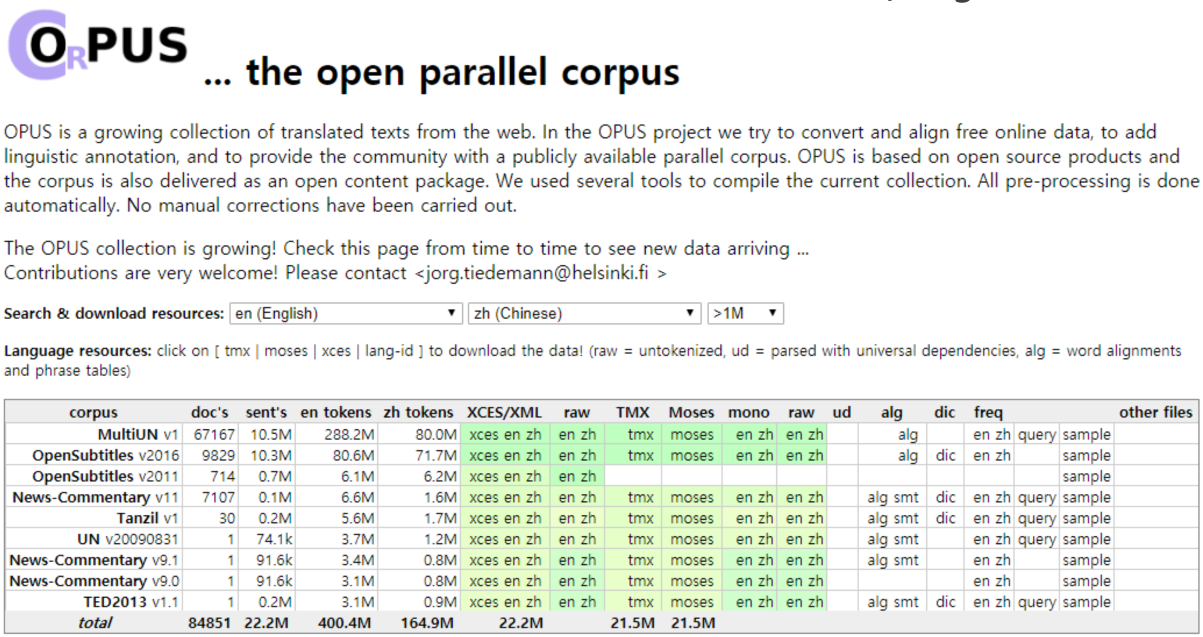
Language Model (fluency)
Learnt from ***monolingual data***

2 Statistical Machine Translation

Parallel corpus and Alignment

How to learn translation model ***from the*** ***parallel corpus****?*

i.e. pairs of human-translated Chinese/English sentences



OPUS ... the open parallel corpus

OPUS is a growing collection of translated texts from the web. In the OPUS project we try to convert and align free online data, to add linguistic annotation, and to provide the community with a publicly available parallel corpus. OPUS is based on open source products and the corpus is also delivered as an open content package. We used several tools to compile the current collection. All pre-processing is done automatically. No manual corrections have been carried out.

The OPUS collection is growing! Check this page from time to time to see new data arriving ...
Contributions are very welcome! Please contact <jorg.tiedemann@helsinki.fi >

Search & download resources: en (English) ▾ zh (Chinese) ▾ >1M ▾

Language resources: click on [tmx | moses | xces | lang-id] to download the data! (raw = untokenized, ud = parsed with universal dependencies, alg = word alignments and phrase tables)

corpus	doc's	sent's	en tokens	zh tokens	XCES/XML	raw	TMX	Moses	mono	raw	ud	alg	dic	freq			other files
MultiUN v1	67167	10.5M	288.2M	80.0M	xces en zh	en zh	tmx	moses	en zh	en zh		alg		en zh	query	sample	
OpenSubtitles v2016	9829	10.3M	80.6M	71.7M	xces en zh	en zh	tmx	moses	en zh	en zh		alg	dic	en zh		sample	
OpenSubtitles v2011	714	0.7M	6.1M	6.2M	xces en zh	en zh										sample	
News-Commentary v11	7107	0.1M	6.6M	1.6M	xces en zh	en zh	tmx	moses	en zh	en zh		alg smt	dic	en zh	query	sample	
Tanzil v1	30	0.2M	5.6M	1.7M	xces en zh	en zh	tmx	moses	en zh	en zh		alg smt	dic	en zh	query	sample	
UN v20090831	1	74.1k	3.7M	1.2M	xces en zh	en zh	tmx	moses	en zh	en zh		alg smt		en zh	query	sample	
News-Commentary v9.1	1	91.6k	3.4M	0.8M	xces en zh	en zh	tmx	moses	en zh	en zh		alg smt		en zh		sample	
News-Commentary v9.0	1	91.6k	3.1M	0.8M	xces en zh	en zh	tmx	moses	en zh	en zh				en zh		sample	
TED2013 v1.1	1	0.2M	3.1M	0.9M	xces en zh	en zh	tmx	moses	en zh	en zh		alg smt	dic	en zh	query	sample	
total	**84851**	**22.2M**	**400.4M**	**164.9M**	**22.2M**		**21.5M**	**21.5M**									

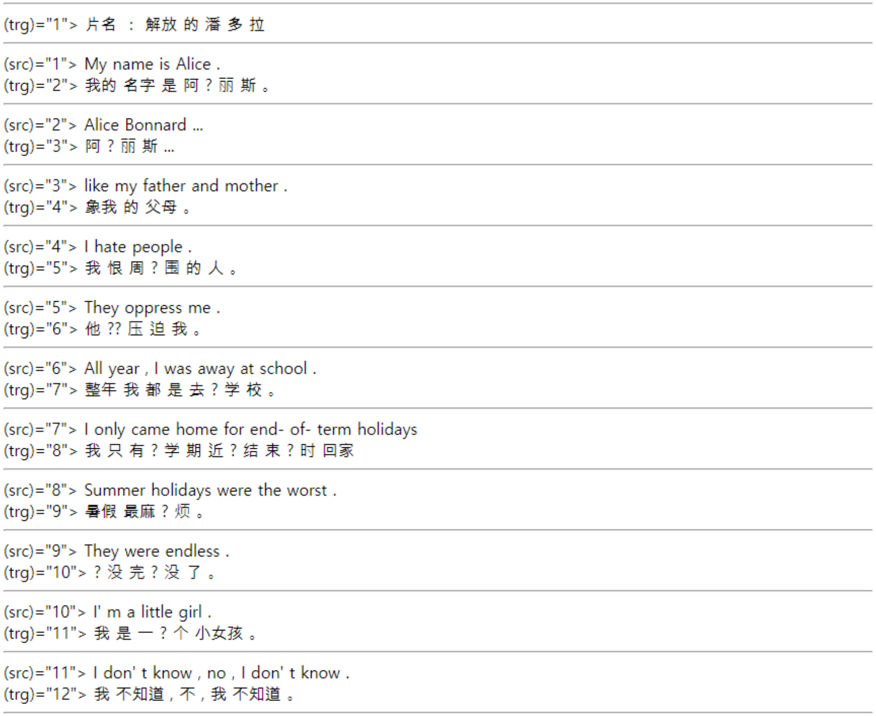
http://opus.nlpl.eu/

2 Statistical Machine Translation

↗ same scene but only different in language
same context

Parallel corpus and Alignment

How to align these sentence (Open subtitles)



(trg)="1"> 片名 ： 解放 的 潘 多 拉

(src)="1"> My name is Alice .
(trg)="2"> 我的 名字 是 阿 ? 丽 斯 。

(src)="2"> Alice Bonnard ...
(trg)="3"> 阿 ? 丽 斯 ...

(src)="3"> like my father and mother .
(trg)="4"> 象我 的 父母 。

(src)="4"> I hate people .
(trg)="5"> 我 恨 周 ? 围 的 人 。

(src)="5"> They oppress me .
(trg)="6"> 他 ?? 压 迫 我 。

(src)="6"> All year , I was away at school .
(trg)="7"> 整年 我 都 是 去 ? 学 校 。

(src)="7"> I only came home for end- of- term holidays
(trg)="8"> 我 只 有 ? 学 期 近 ? 结 束 ? 时 回家

(src)="8"> Summer holidays were the worst .
(trg)="9"> 暑假 最麻 ? 烦 。

(src)="9"> They were endless .
(trg)="10"> ? 没 完 ? 没 了 。

(src)="10"> I' m a little girl .
(trg)="11"> 我 是 一 ? 个 小女孩 。

(src)="11"> I don' t know , no , I don' t know .
(trg)="12"> 我 不知道 , 不 , 我 不知道 。

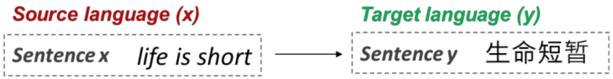
2 Statistical Machine Translation

How to learn translation model?

How to learn translation model ***from the*** ***parallel corpus****?*

$$\begin{tikzcd} \underline{P(x|y)} \arrow[d, shift left] \arrow[d, shift right] \\ P(x, a|y) \end{tikzcd}$$

i.e. pairs of human-translated Chinese/English sentences



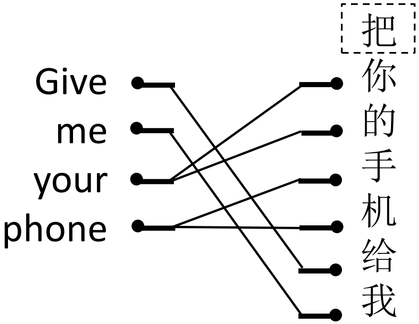
a is the ***alignment***

Alignment is the correspondence between particular words in the translated sentence pair. (i.e. word-level correspondence between *source sentence x* and *target sentence y*)

2 Statistical Machine Translation

What is Alignment *a*?

“The correspondence between particular words in the translated sentence pair”



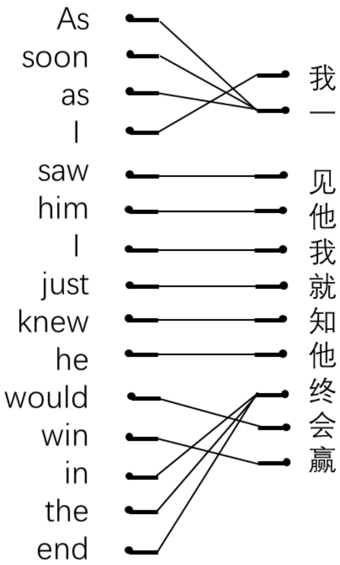
Spurious word

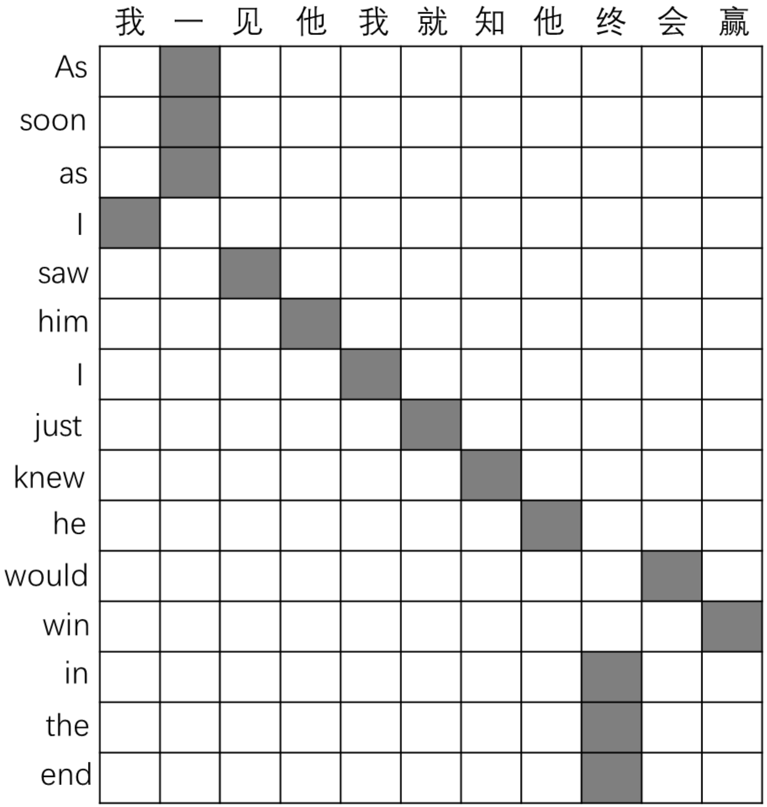
	把	你	的	手	机	给	我
Give						■	
me							■
your		■	■				
phone				■	■		

Statistical Machine Translation

What is Alignment *a*?

Many-to-One Alignment



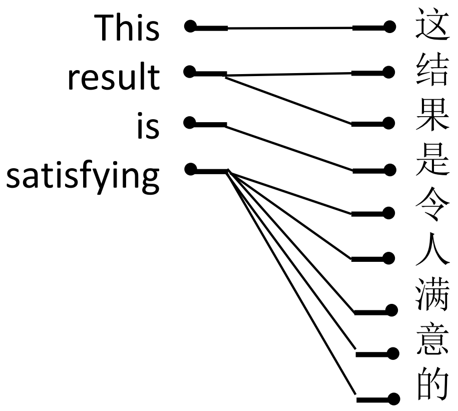


	我	一	见	他	我	就	知	他	终	会	赢
As		■									
soon		■									
as		■									
I	■										
saw			■								
him				■							
I					■						
just						■					
knew							■				
he								■			
would										■	
win											■
in									■		
the									■		
end									■		

2 Statistical Machine Translation

What is Alignment *a*?

One-to-Many Alignment

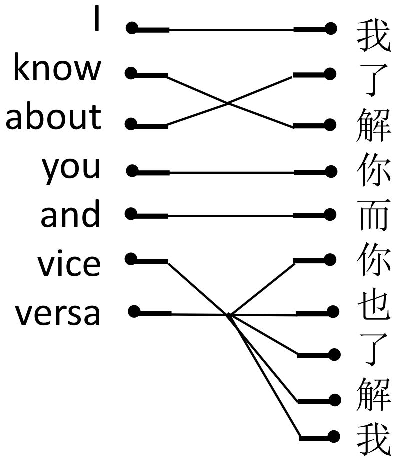


	这	结	果	是	令	人	满	意	的
This	■								
result		■	■						
is				■					
satisfying					■	■	■	■	■

2 Statistical Machine Translation

What is Alignment *a*?

Many-to-many Alignment



	我	了	解	你	而	你	也	了	解	我
I	■									
know		■	■							
about		■	■							
you				■						
and					■					
vice						■	■	■	■	■
versa						■	■	■	■	■

2 Statistical Machine Translation

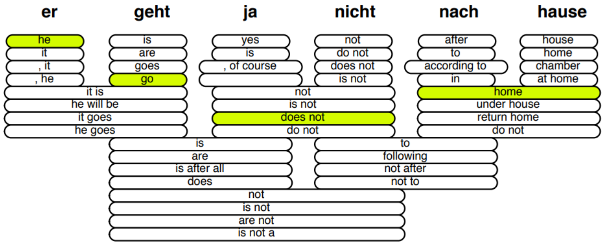
Decoding for SMT

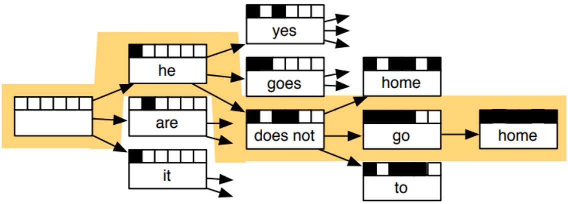
$$\text{argmax}_y \underline{P(x|y)}\,\underline{P(y)}$$

Translation Model (fidelity)
*Learnt from **parallel data***

Language Model (fluency)
*Learnt from **monolingual data***

- We could enumerate every possible y and calculate the probability? Too expensive!
- Answer: Use a ***heuristic search algorithm*** to ***search for the best*** translation, discarding hypotheses that are too low-probability





backtrack from highest scoring complete hypothesis

2 Statistical Machine Translation

Statistical Machine Translation

The Best System

*SMT was a **huge research field** and **Extremely complex System***

Hundreds of important details (haven't mentioned here)

- *Systems had many separately-designed subcomponents*
- *Lots of feature engineering*
  - *Need to design features to capture particular language phenomena*
- *Require compiling and maintaining extra resources*
  - *Like tables of equivalent phrases*
- *Lots of human effort to maintain*
  - *Repeated effort for each language pair!*

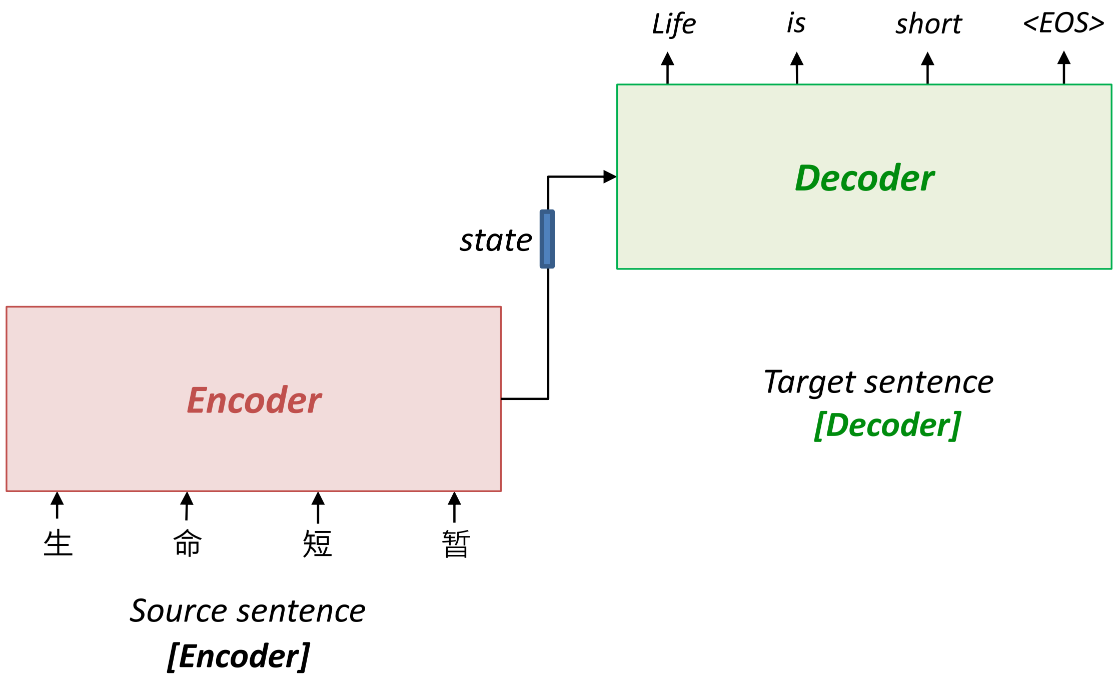
3 Neural Machine Translation

3 Neural Machine Translation

Neural Machine Translation with Seq2Seq

“a way to do Machine Translation with a single neural network (NN)”

- *The NN architecture is called **seq2seq** and involves **two RNNs.***



Target sentence
[Decoder]

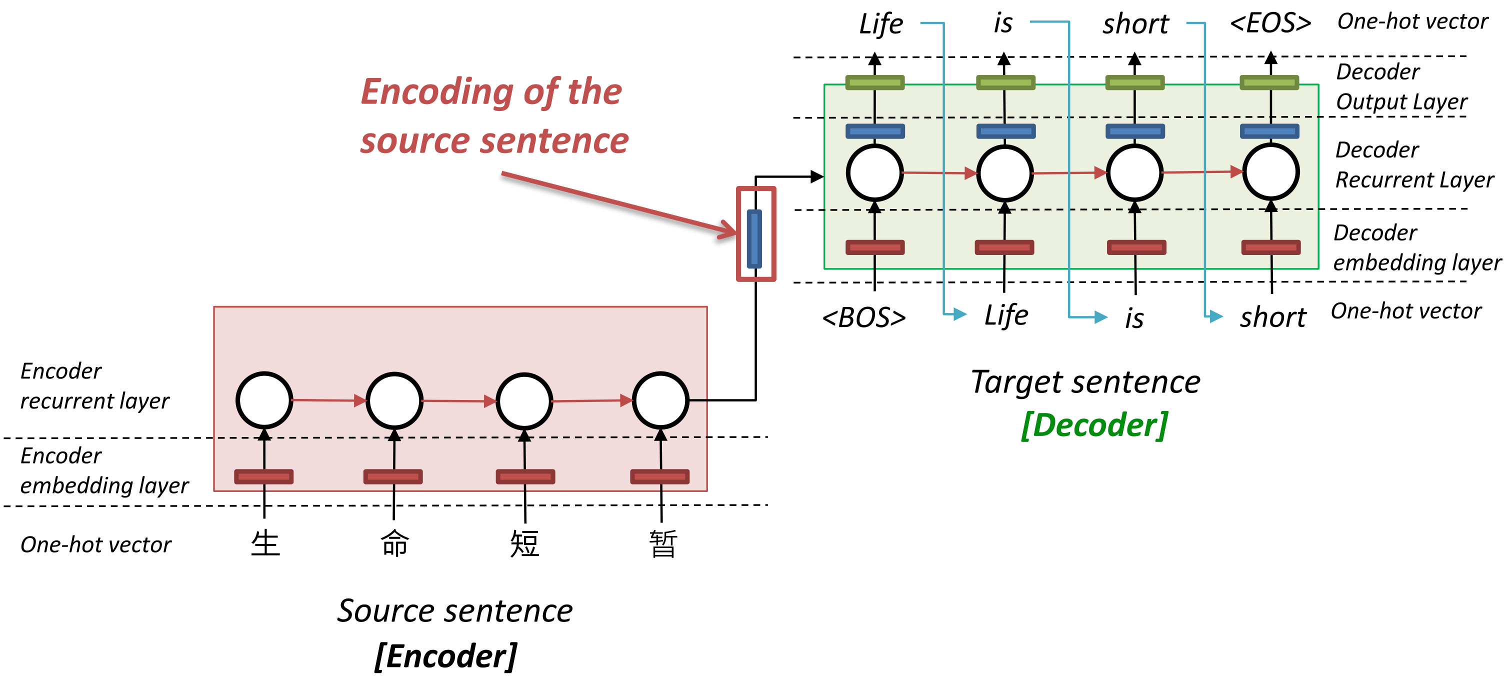
Source sentence
[Encoder]

3 Neural Machine Translation

Neural Machine Translation with Seq2Seq

“a way to do Machine Translation with a single neural network (NN)”

- *The NN architecture is called **seq2seq** and involves **two RNNs.***

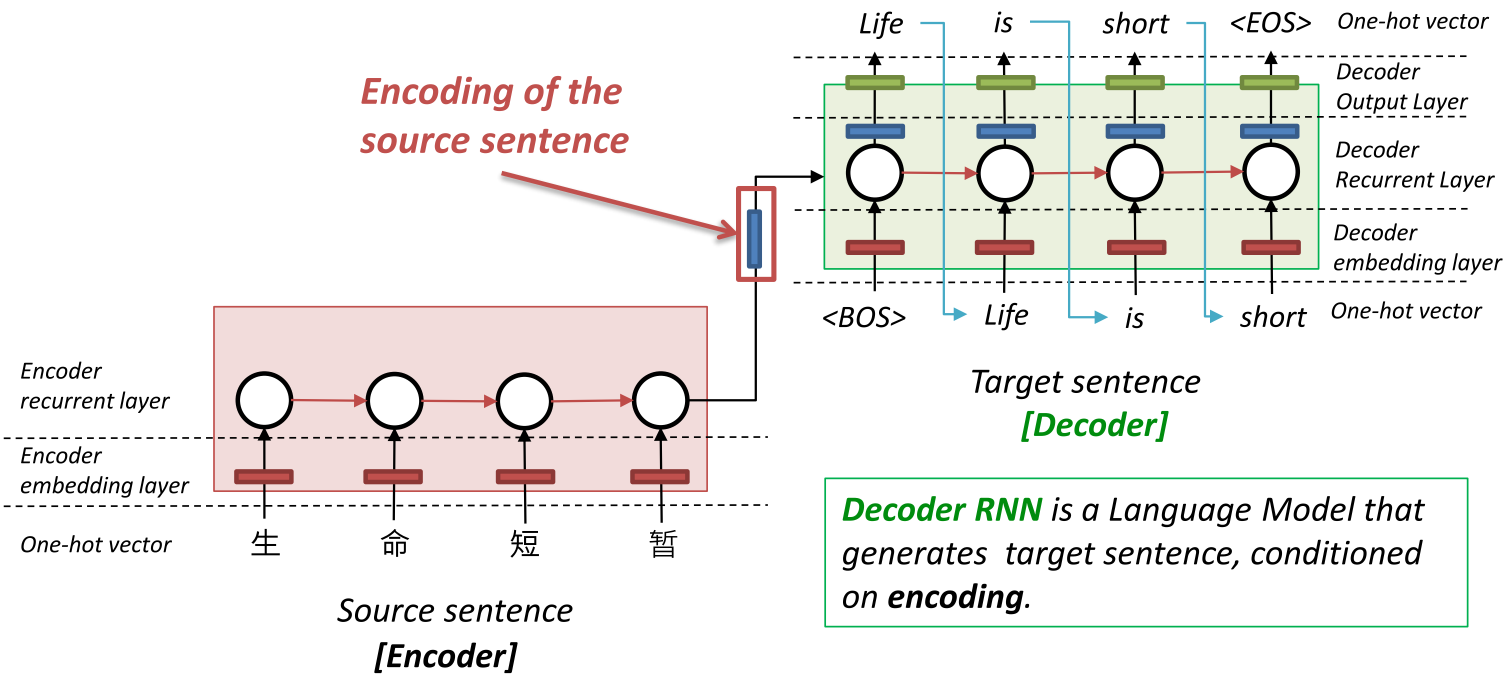


3 Neural Machine Translation

Neural Machine Translation with Seq2Seq

“a way to do Machine Translation with a single neural network (NN)”

- *The NN architecture is called **seq2seq** and involves **two RNNs.***



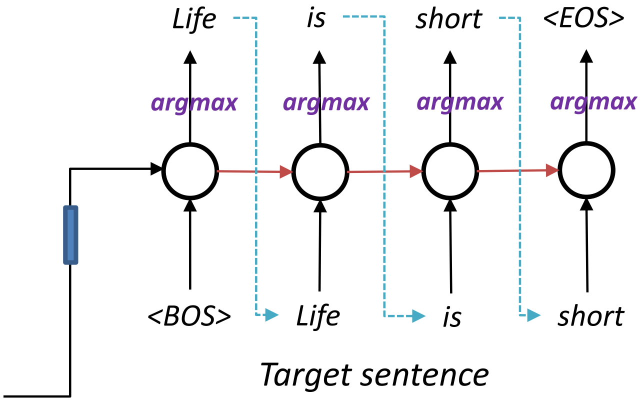
Decoder RNN *is a Language Model that generates target sentence, conditioned on **encoding**.*

Neural Machine Translation: Greedy Decoding *[Recap]*

Language Model Decoding: Recap

- Generate the sentence by taking ***argmax*** (the most probable word) on each step
- Use that as the next word, and feed it as input on the next step
- Keep going until you produce *<EOS>*

training: teacher forcing



Target sentence

[Decoder]

Greedy decoding has no way to undo decisions!!
(Ungrammatical, unnatural)

Solution..? try computing all possible sequences

Neural Machine Translation: Beam Search Decoding *[Recap]*

Language Model Decoding: Recap

- A search algorithm which aims to find a high-probability sequence (not necessarily the optimal sequence, though) by tracking multiple possible sequences at once.

- On each step of decoder, keep track of the k most probable partial sequences (which we call hypotheses)
- K is the beam size (in practice around 5 to 10)

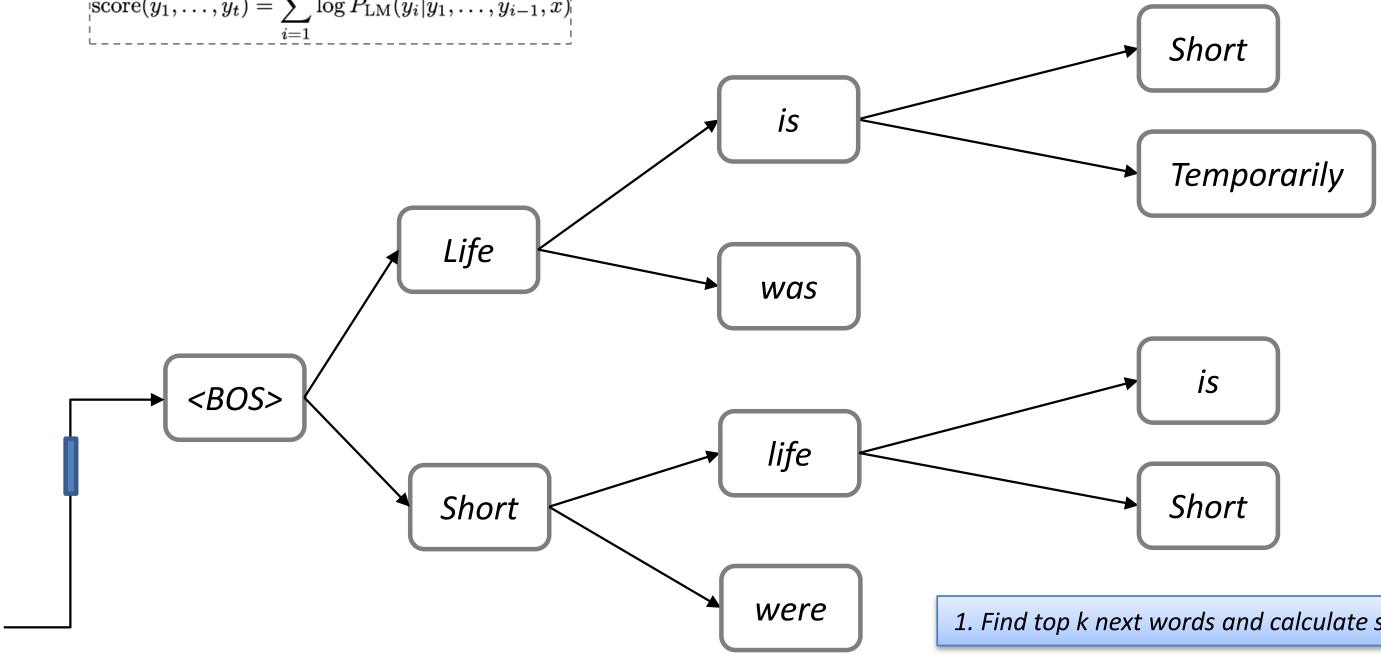
- After you reach some stopping criterion, choose the sequence with the highest probability (factoring in some adjustment for length)

Neural Machine Translation: Beam Search Decoding

Language Model Decoding: Recap

Assume that k(beam size)=2

$$\text{score}(y_1, \ldots, y_t) = \sum_{i=1}^{t} \log P_{\text{LM}}(y_i | y_1, \ldots, y_{i-1}, x)$$



1. Find top k next words and calculate scores

2. Of these k^2 hypotheses, keep only highest k

3 Neural Machine Translation

Evaluate Machine Translation

BLEU *(Bilingual Evaluation Understudy)*

“Compares the ***machine-written translation*** *to one or several* ***human-written translation****(s), and computes a similarity score based on”*

- *n-gram precision (usually for 1 to 4-grams)*
- *Plus a penalty for too-short system translations*

BLEU *is useful but imperfect*

- *Many valid ways to translate a sentence*
- *So a good translation can get a poor BLEU score because it has low n-gram overlap with the human translation*

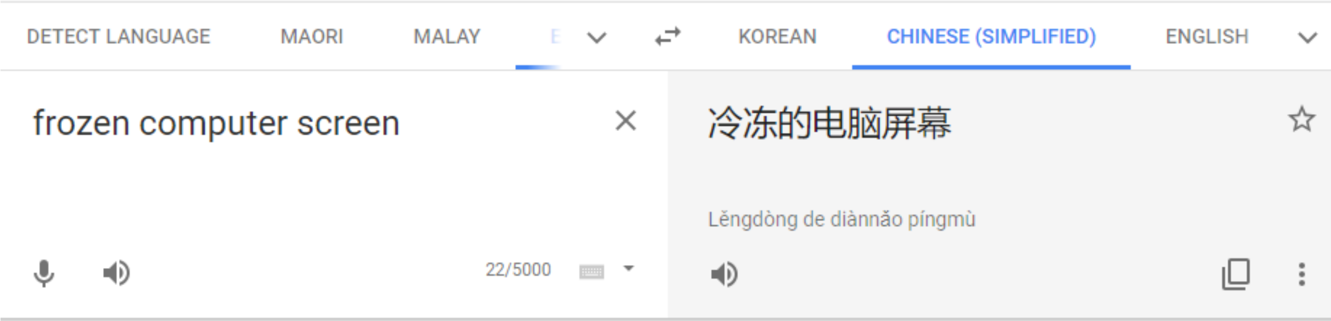
3 Neural Machine Translation

However, there are still several difficulties…

- ***Out-of-vocabulary (OOV)*** *words*
- *Domain* ***mismatch*** *between train and test data*
- *Maintaining context over* ***longer text***
- ***Low-resource*** *language pairs*

3 Neural Machine Translation

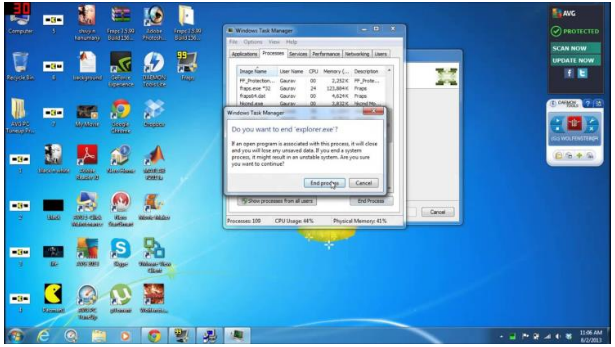
Machine Translation is not PERFECT…

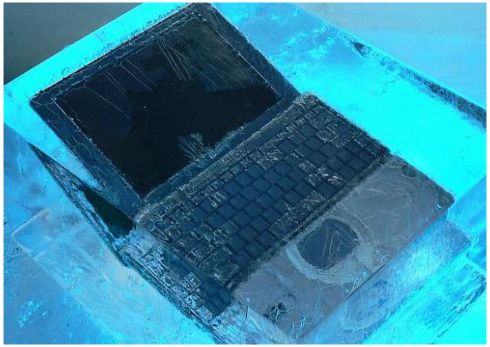


frozen computer screen

冷冻的电脑屏幕

Lěngdòng de diànnǎo píngmù

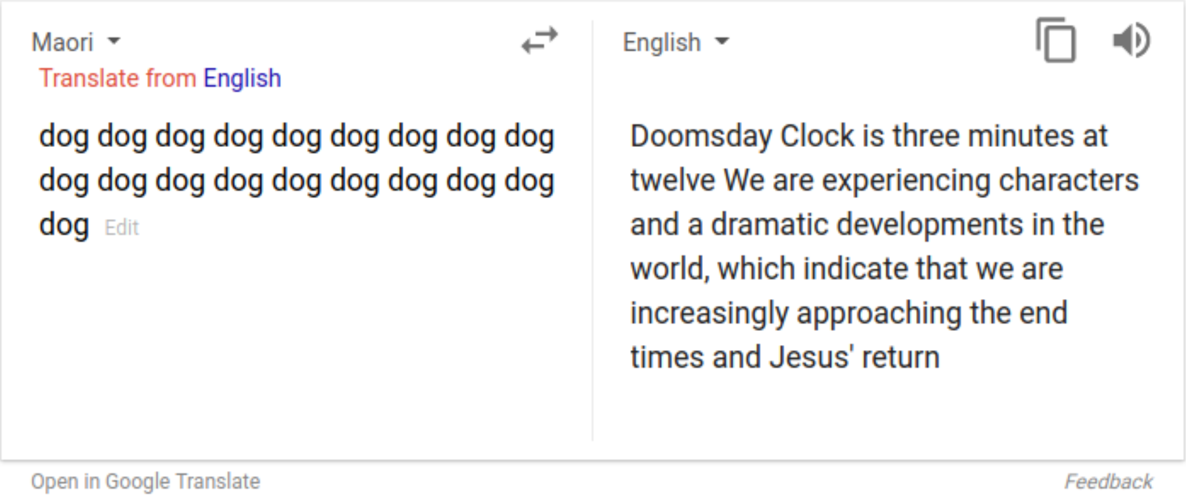




Using common sense is still hard and **NMT** picks up biases in training data

3 Neural Machine Translation

Machine Translation is not PERFECT…



dog dog dog dog dog dog dog dog dog dog dog dog dog dog dog dog dog dog dog

Doomsday Clock is three minutes at twelve We are experiencing characters and a dramatic developments in the world, which indicate that we are increasingly approaching the end times and Jesus' return

Uninterpretable systems do strange things

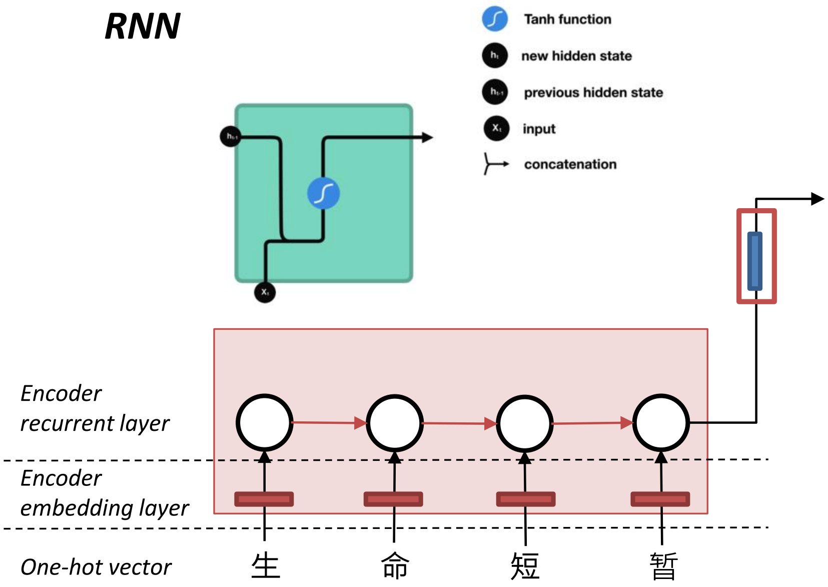
https://www.vice.com/en_us/article/j5npeg/why-is-google-translate-spitting-out-sinister-religious-prophecies

3 Neural Machine Translation

Neural Machine Translation with Seq2Seq

RNN-based neural MT was sort of successful! But…

RNN



Source sentence

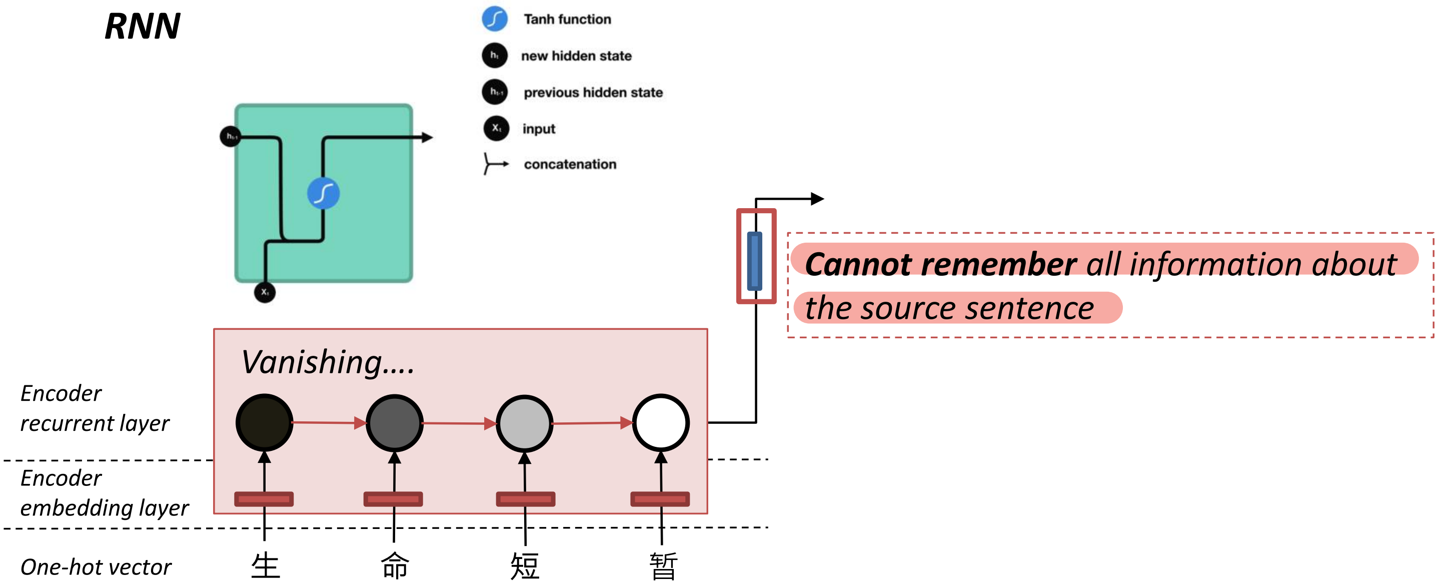
[Encoder]

3 Neural Machine Translation

Neural Machine Translation with Seq2Seq

RNN-based neural MT was sort of successful! But…

RNN



***Cannot remember** all information about the source sentence*

Source sentence

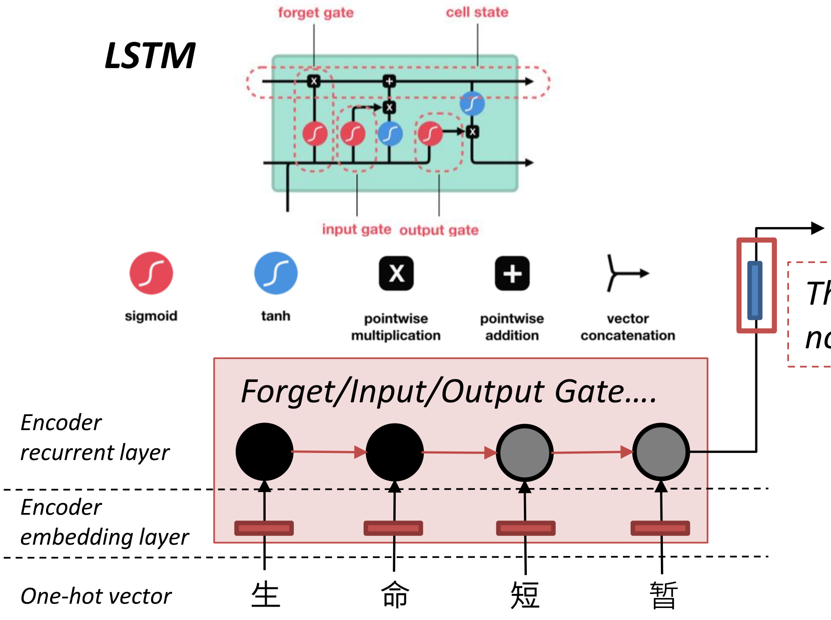
[Encoder]

3 Neural Machine Translation

Neural Machine Translation with Seq2Seq

RNN-based neural MT was successful! But…

LSTM



They ***can remember sequences of 100s****, not 1000s or 10,000s or more.*

Source sentence

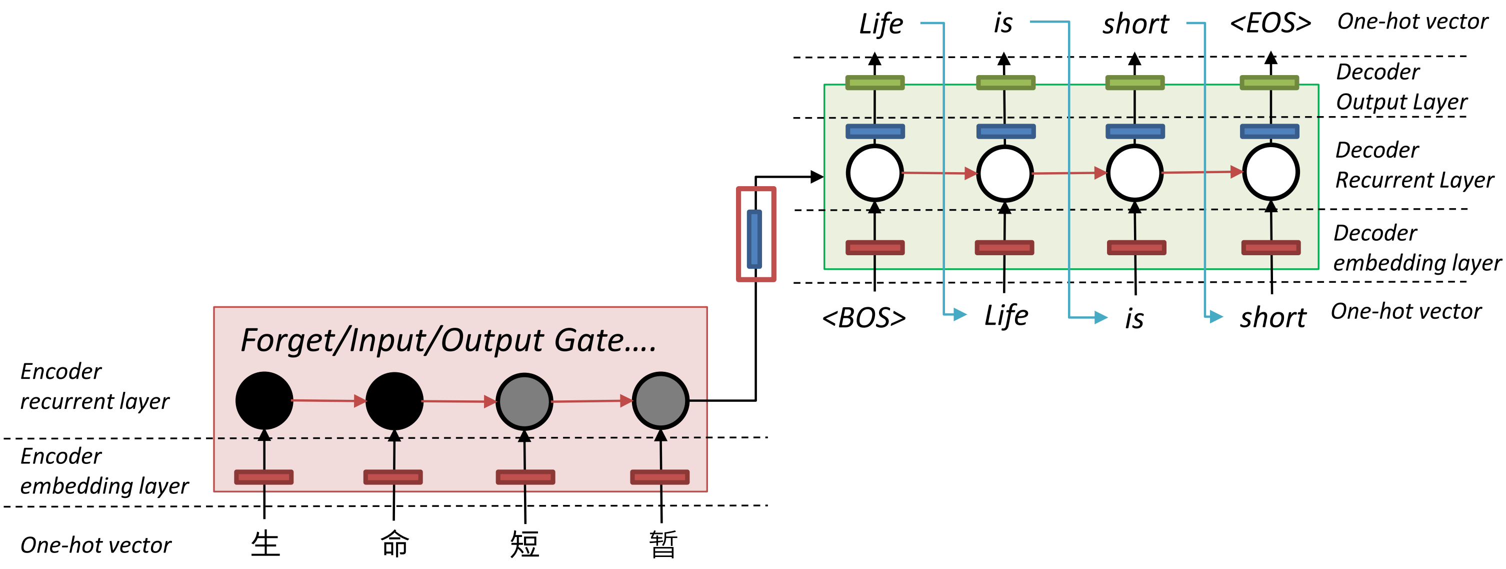
[Encoder]

3 Neural Machine Translation

Neural Machine Translation with Seq2Seq

Then, how to solve the information bottleneck issue?

Attention!



Source sentence

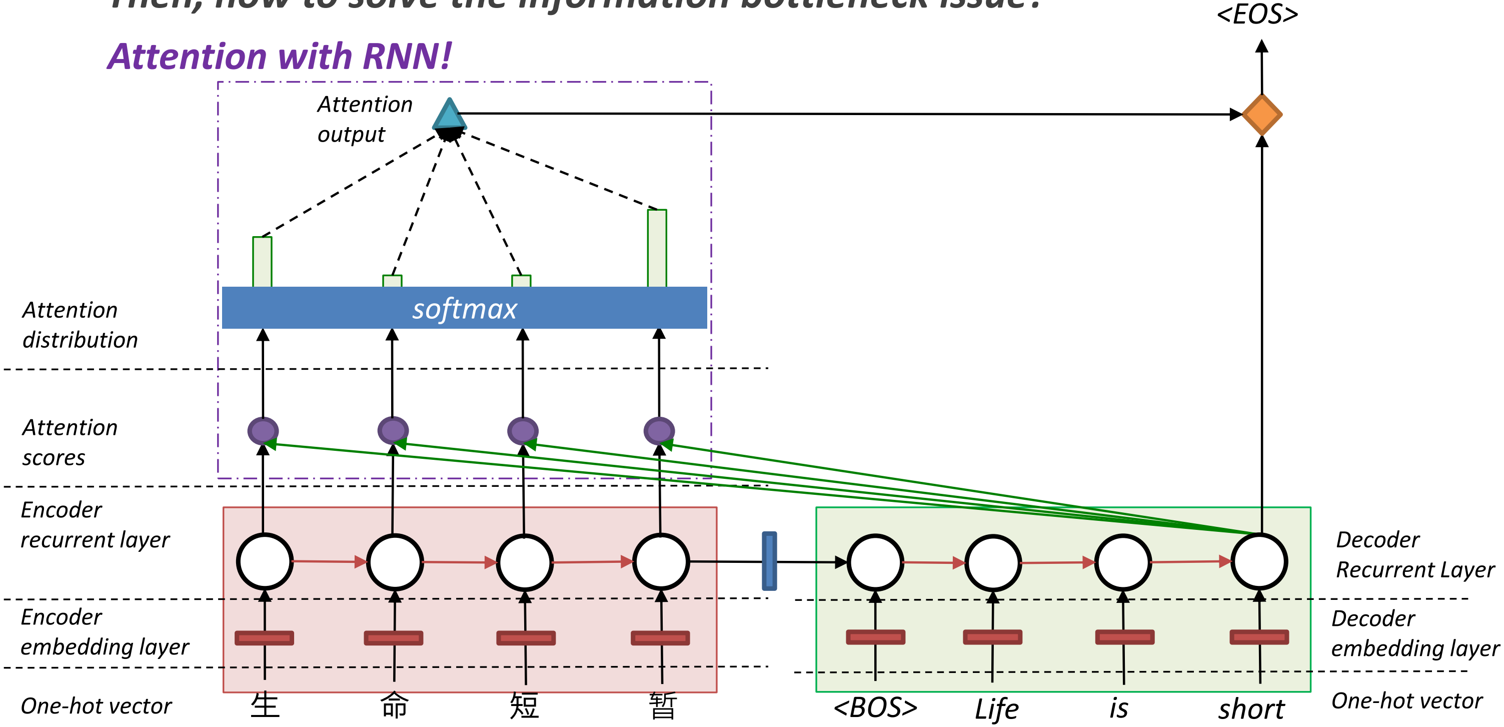
[Encoder]

3 Neural Machine Translation

Neural Machine Translation with RNN and Attention

Then, how to solve the information bottleneck issue?

Attention with RNN!



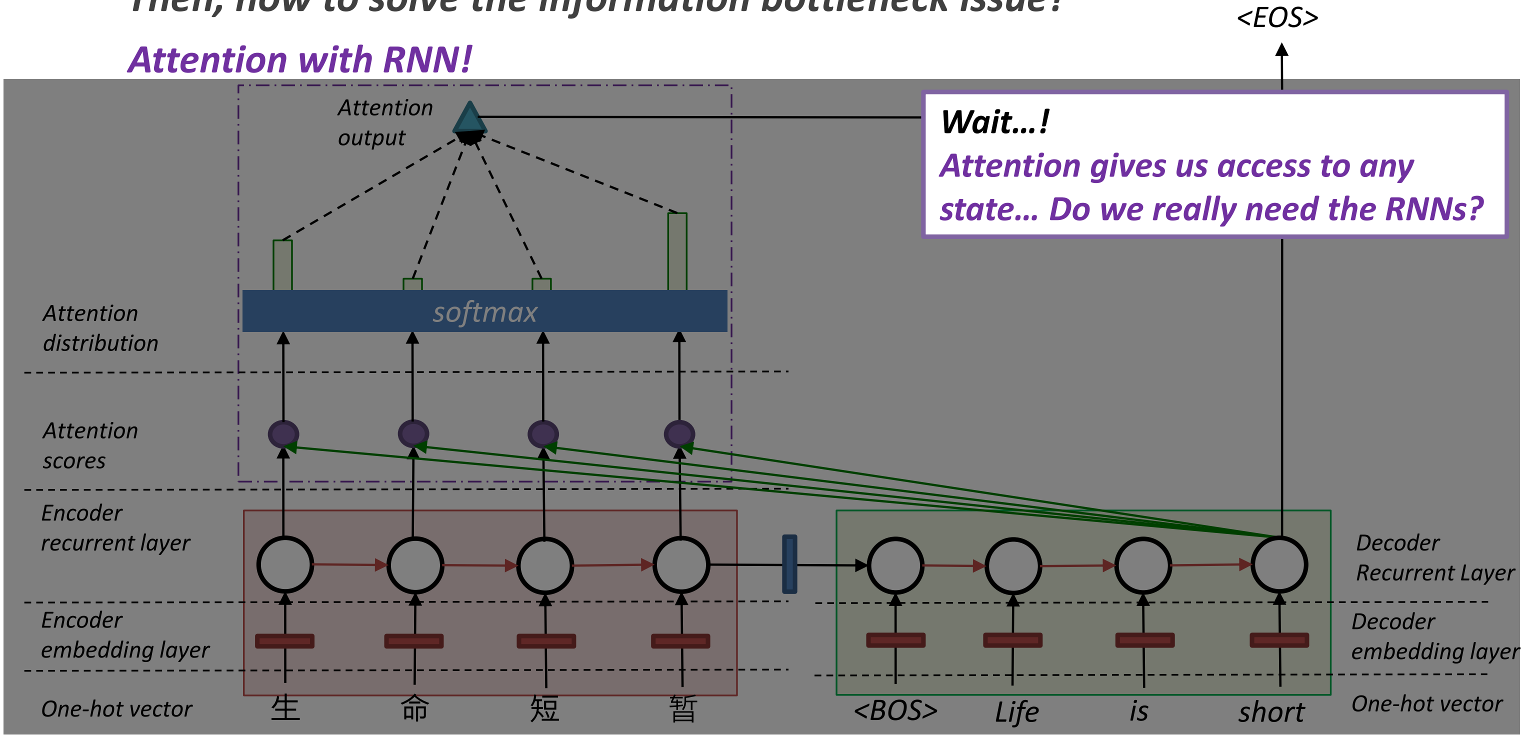
3 Neural Machine Translation

Attention has all info about any state.

Neural Machine Translation with RNN and Attention

Then, how to solve the information bottleneck issue?

Attention with RNN!



Wait…!
Attention gives us access to any state… Do we really need the RNNs?

4 Attention and Transformer for MT

Early 2018 ~

4 Attention and Transformer for MT

Attention is All You Need (Vaswani et al., 2017)

Attention Is All You Need

Ashish Vaswani*
Google Brain
avaswani@google.com

Noam Shazeer*
Google Brain
noam@google.com

Niki Parmar*
Google Research
nikip@google.com

Jakob Uszkoreit*
Google Research
usz@google.com

Llion Jones*
Google Research
llion@google.com

Aidan N. Gomez* †
University of Toronto
aidan@cs.toronto.edu

Łukasz Kaiser*
Google Brain
lukaszkaiser@google.com

Illia Polosukhin* ‡
illia.polosukhin@gmail.com

Abstract

The dominant sequence transduction models are based on complex recurrent or convolutional neural networks that include an encoder and a decoder. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to be superior in quality while being more parallelizable and requiring significantly less time to train. Our model achieves 28.4 BLEU on the WMT 2014 English-to-German translation task, improving over the existing best results, including ensembles, by over 2 BLEU. On the WMT 2014 English-to-French translation task, our model establishes a new single-model state-of-the-art BLEU score of 41.8 after training for 3.5 days on eight GPUs, a small fraction of the training costs of the best models from the literature. We show that the Transformer generalizes well to other tasks by applying it successfully to English constituency parsing both with large and limited training data.

Encoder-Decoder with only Attention

Core Task: Machine Translation with Parallel Corpus

- *Use self-attention in the encoder, instead of RNN or CNNs*
- *Predict each translated word*
- *Final cost/error function*

→ *standard cross-entropy error on top of a softmax classifier*

Attention is All You Need!

'The Transformer'!!

Output (Target Language)

Hello World

The Transformer!

こんにちは世界

Input (Source Language)

4 Attention and Transformer for MT

Attention is All You Need (Vaswani et al., 2017)

Attention Is All You Need

Ashish Vaswani* Google Brain avaswani@google.com
Noam Shazeer* Google Brain noam@google.com
Niki Parmar* Google Research nikip@google.com
Jakob Uszkoreit* Google Research usz@google.com
Llion Jones* Google Research llion@google.com
Aidan N. Gomez* † University of Toronto aidan@cs.toronto.edu
Łukasz Kaiser* Google Brain lukaszkaiser@google.com
Illia Polosukhin* ‡ illia.polosukhin@gmail.com

Abstract

The dominant sequence transduction models are based on complex recurrent or convolutional neural networks that include an encoder and a decoder. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to be superior in quality while being more parallelizable and requiring significantly less time to train. Our model achieves 28.4 BLEU on the WMT 2014 English-to-German translation task, improving over the existing best results, including ensembles, by over 2 BLEU. On the WMT 2014 English-to-French translation task, our model establishes a new single-model state-of-the-art BLEU score of 41.8 after training for 3.5 days on eight GPUs, a small fraction of the training costs of the best models from the literature. We show that the Transformer generalizes well to other tasks by applying it successfully to English constituency parsing both with large and limited training data.

Encoder-Decoder with only Attention

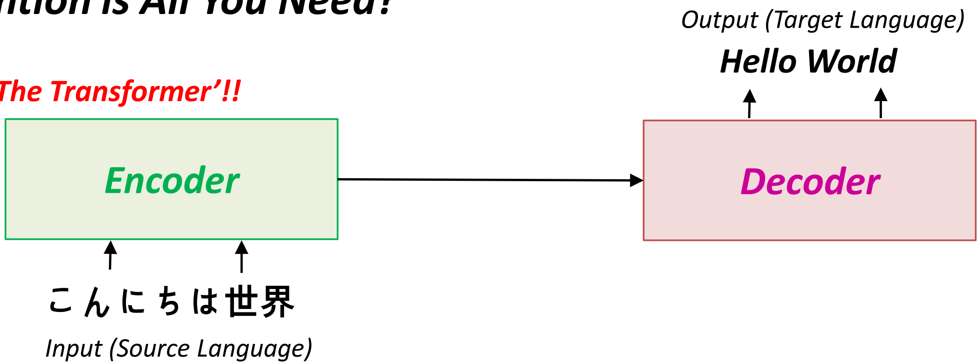
Core Task: Machine Translation with Parallel Corpus

- *Use self-attention in the encoder, instead of RNN or CNNs*
- *Predict each translated word*
- *Final cost/error function*

→ *standard cross-entropy error on top of a softmax classifier*

Attention is All You Need?

'The Transformer'!!



4 Attention and Transformer for MT

The Transformer

Encoder – Decoder Architecture

stacked 6 layers

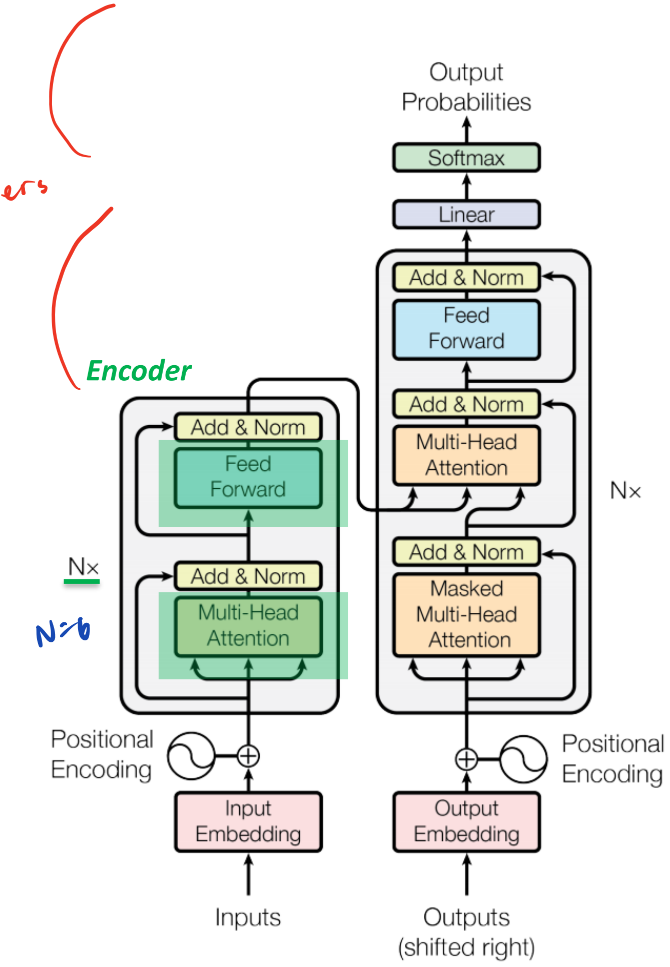
1. Encoder

A stack of N=6 identical layers.

Each layer with two sub-layers:

1. *Multi-head self-attention mechanism*
2. *Position-wise fully connected feed-forward network*

** Residual connection around each of the two sub-layers, followed by layer normalisation*



The transformer – model architecture

4 Attention and Transformer for MT

The Transformer

Encoder – Decoder Architecture

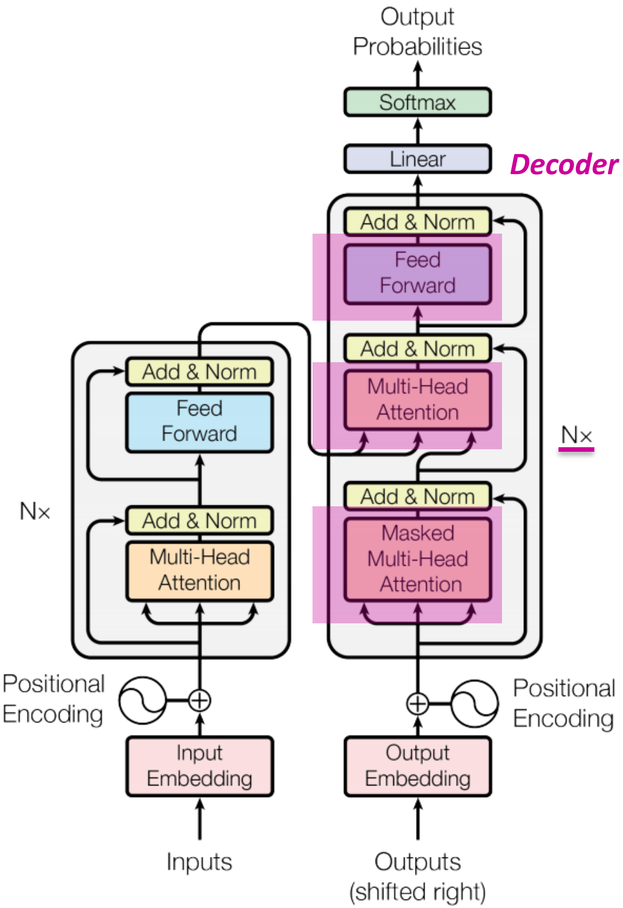
2. Decoder

A stack of N=6 identical layers.

Each layer with three sub-layers:

1. *Multi-head self-attention mechanism*
2. *Position-wise fully connected feed-forward network*
3. *Masked Multi-head self-attention*

* *Residual connection around each of the two sub-layers, followed by layer normalisation*



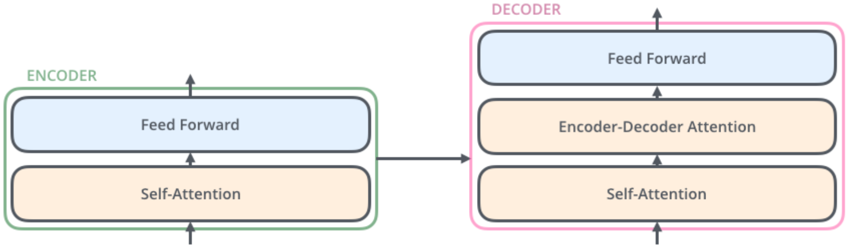
The transformer – model architecture

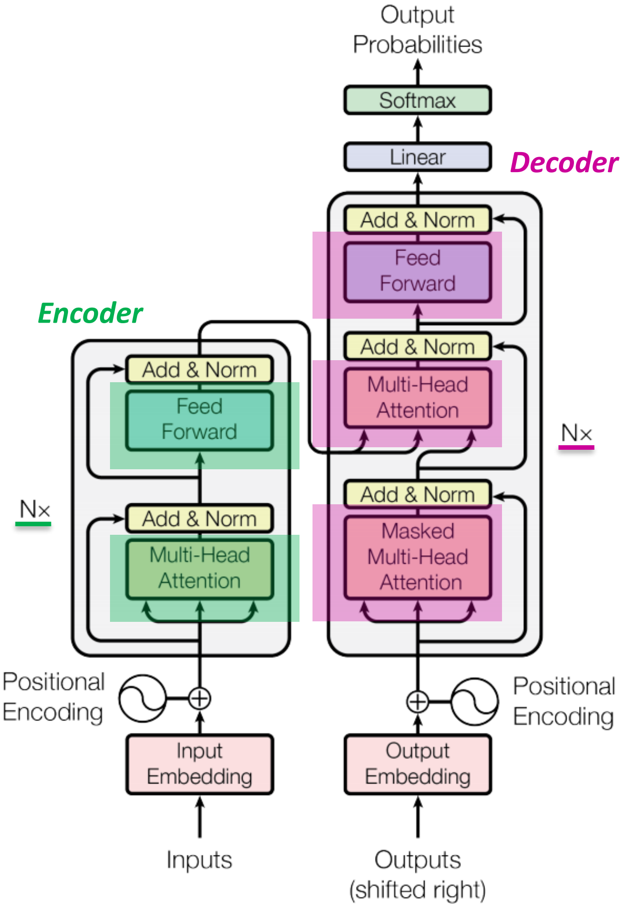
4 Attention and Transformer for MT

The Transformer

Encoder – Decoder Architecture

Brief Summary





The transformer – model architecture

4 Attention and Transformer for MT

The Transformer – Encoder (Stage1)

We are not using RNN anymore… No time step concept!

To make use of ***the order of the sequence***, ***inject*** *information about* ***the position of the tokens*** *in the sequence.*

periodic

even

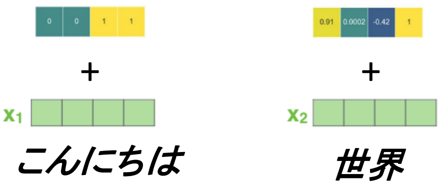
Positional Encoding

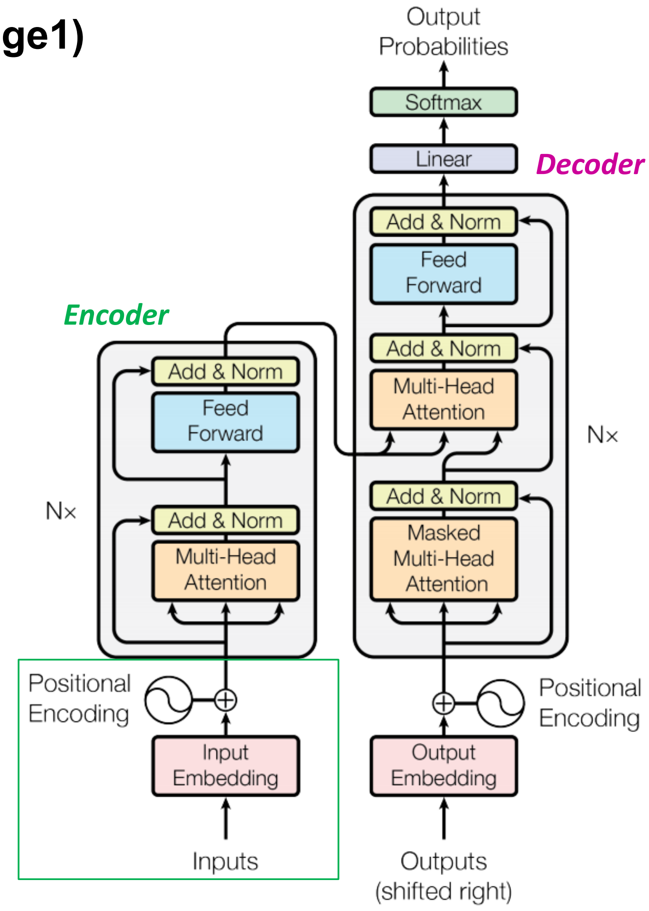
odd

(use sin and cos for position/dimension)

Input embedding

(a vector of size 512)

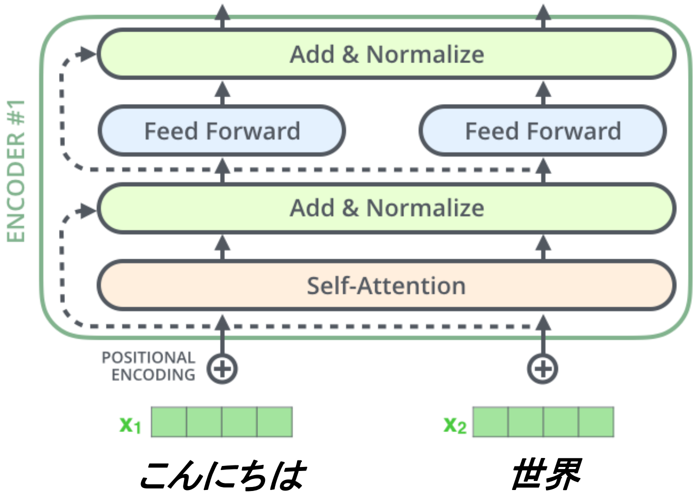


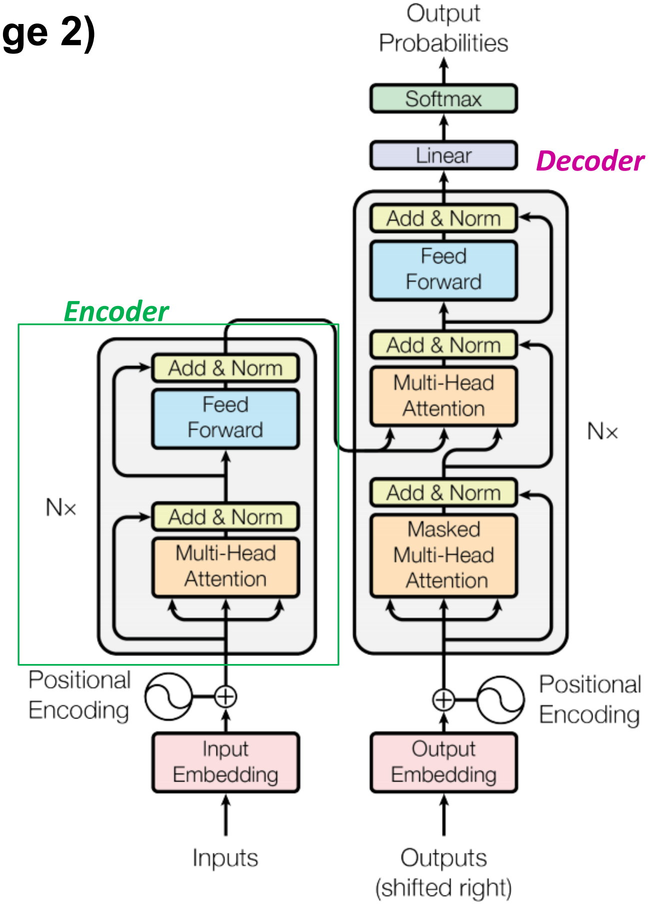


The transformer – model architecture

4 Attention and Transformer for MT

The Transformer – Encoder (Stage 2)

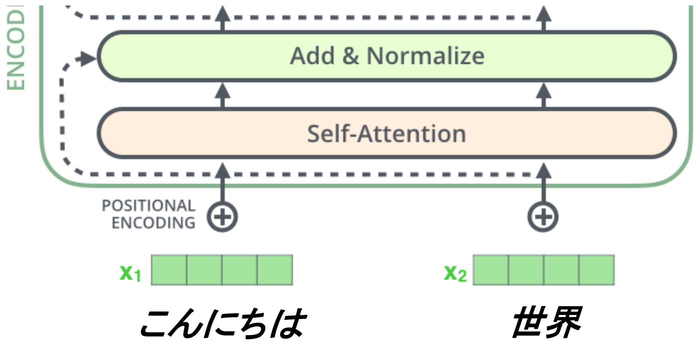


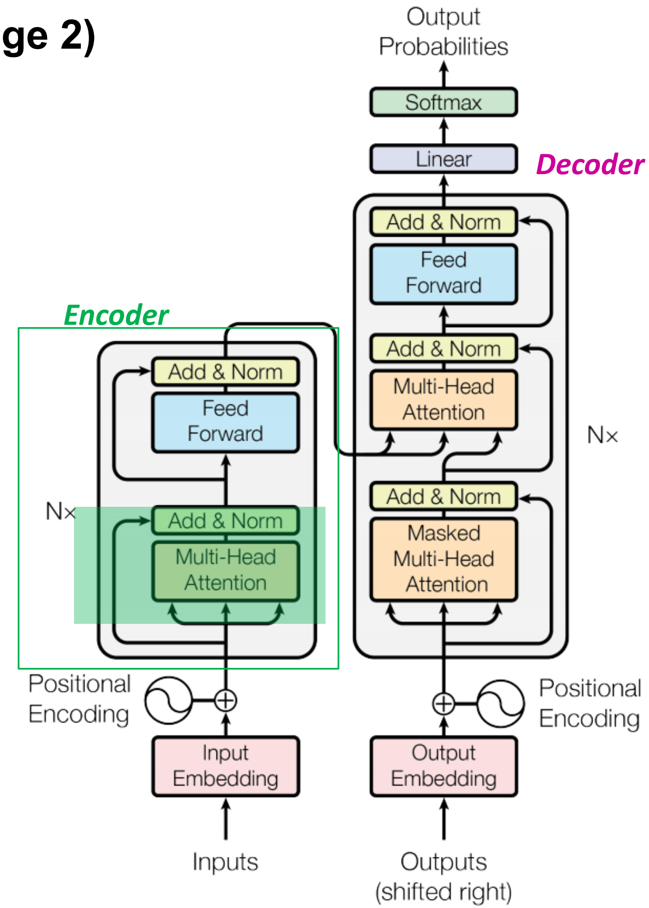


The transformer – model architecture

4 Attention and Transformer for MT

The Transformer – Encoder (Stage 2)



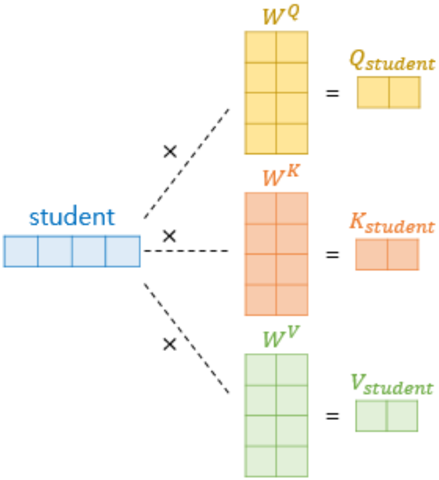


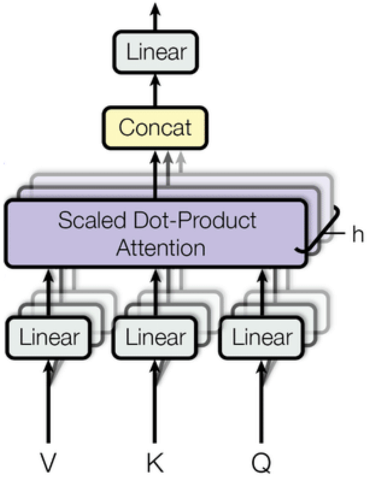
The transformer – model architecture

4 Attention and Transformer for MT

The Transformer – Encoder (Stage 2)

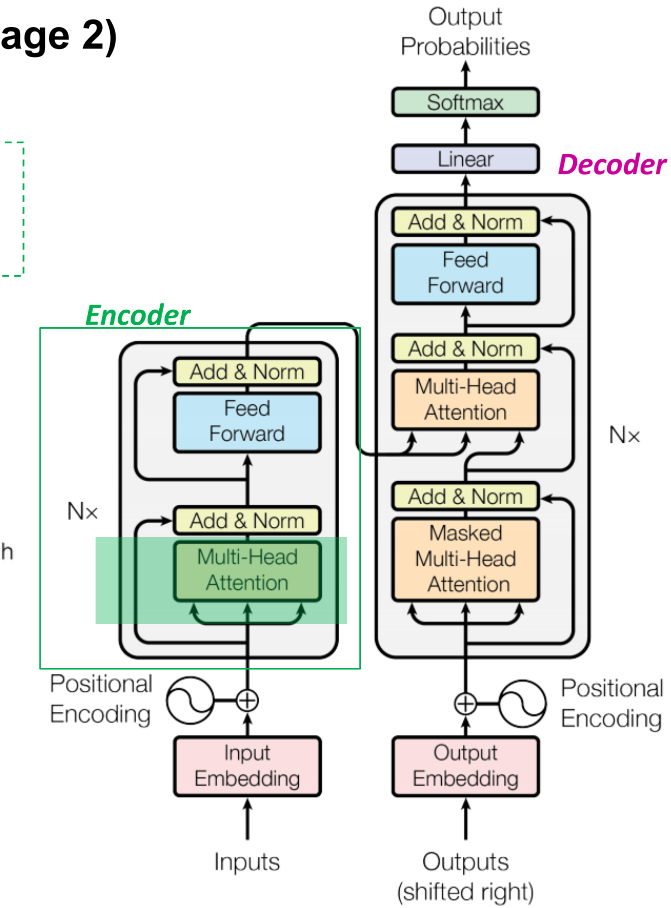
Multi-head attention** allows the model to jointly attend to **information from different representation subspaces at different positions





Multi-Head Attention
(With self attention)

Q=Query, K=Key, V=Value
(64 dimension)

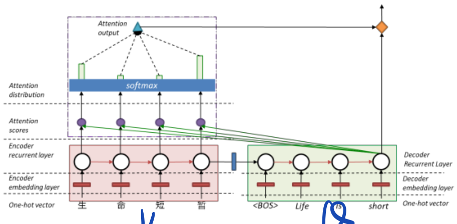


The transformer – model architecture

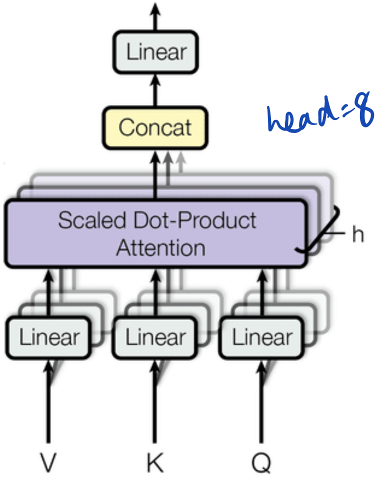
4 Attention and Transformer for MT

The Transformer – Encoder (Stage 2)

Multi-head attention** allows the model to jointly attend to **information from different representation subspaces at different positions



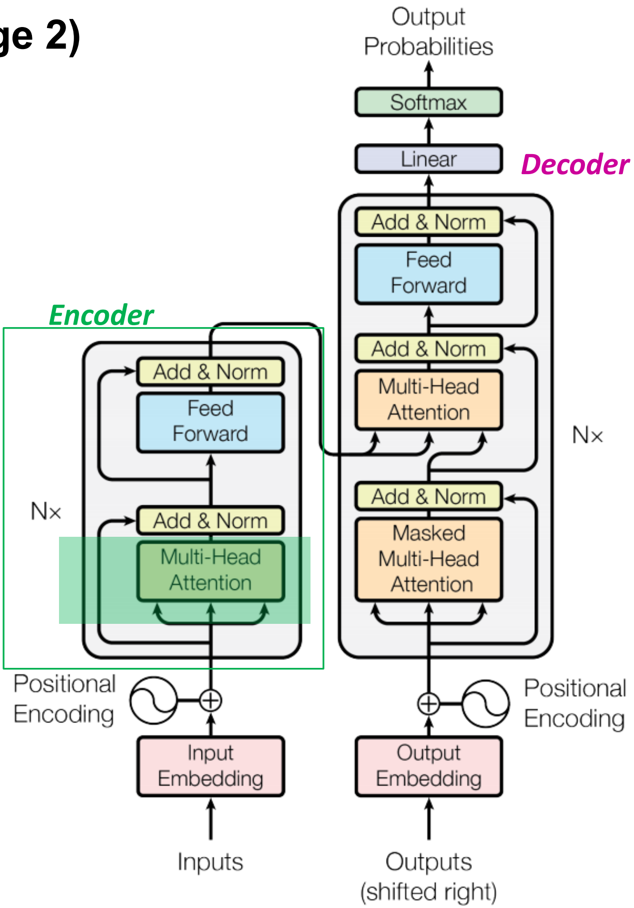
Source and Target attention



vector size 512
512 ÷ 8 = 64
↓
1 head

Multi-Head Attention
(With self attention)

Q=Query, K=Key, V=Value (64 dimension)

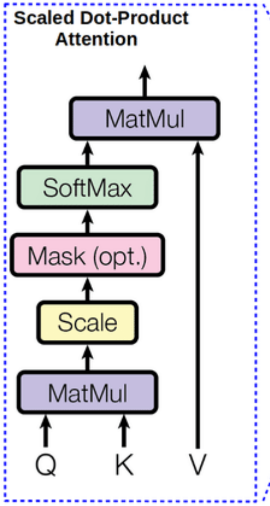


The transformer – model architecture

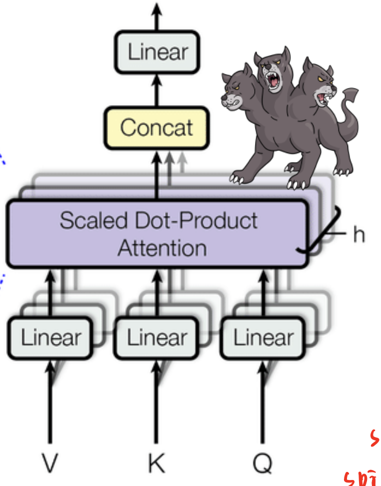
4 Attention and Transformer for MT

The Transformer – Encoder (Stage 2)

Multi-head attention** allows the model to jointly attend to **information from different representation subspaces at different positions



Self-attention

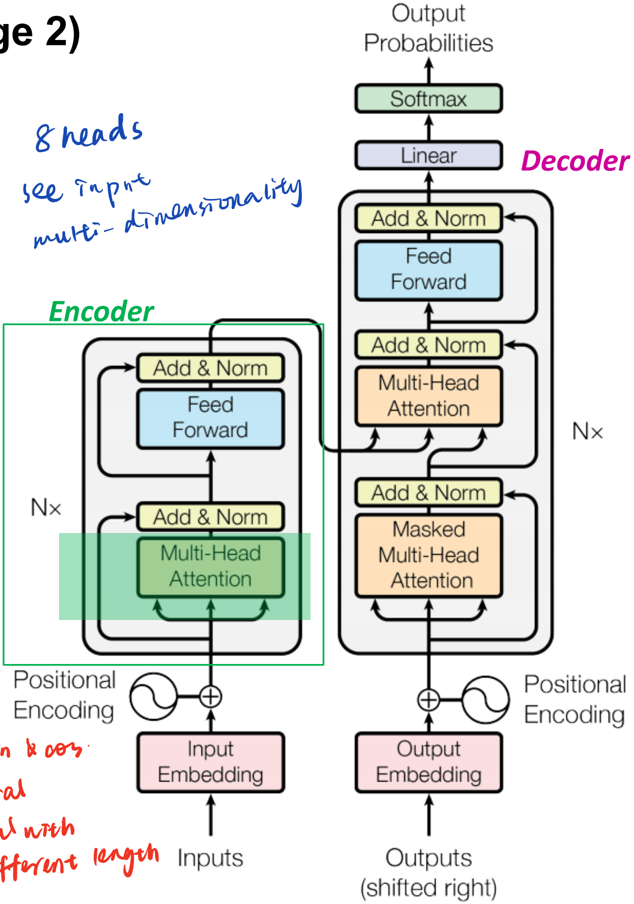


Multi-Head Attention
(With self attention)

Q=Query, K=Key, V=Value (64 dimension)

8 heads

see input multi-dimensionality



sin & cos.

spiral

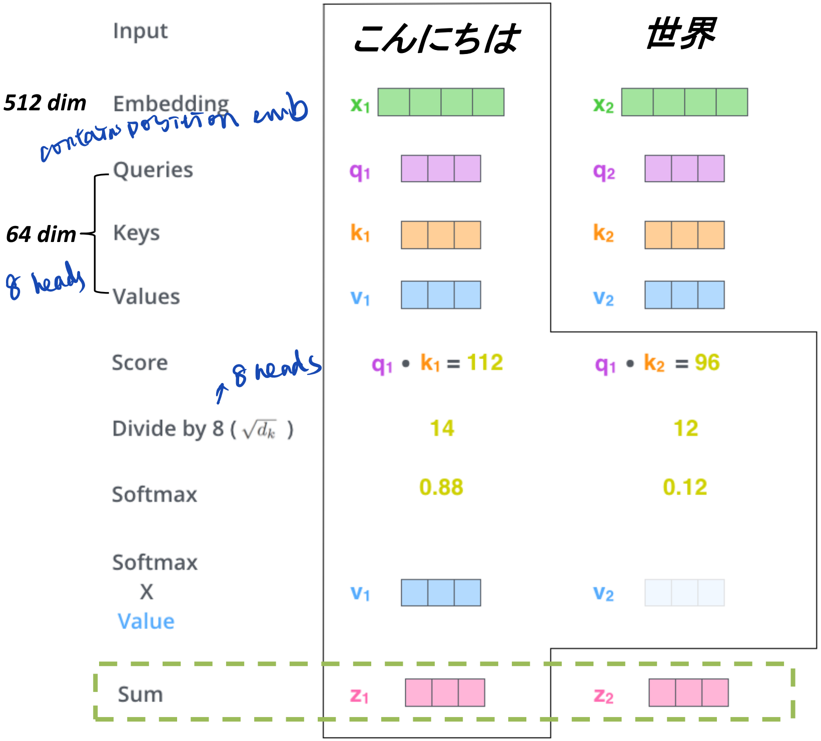
deal with different length

The transformer – model architecture

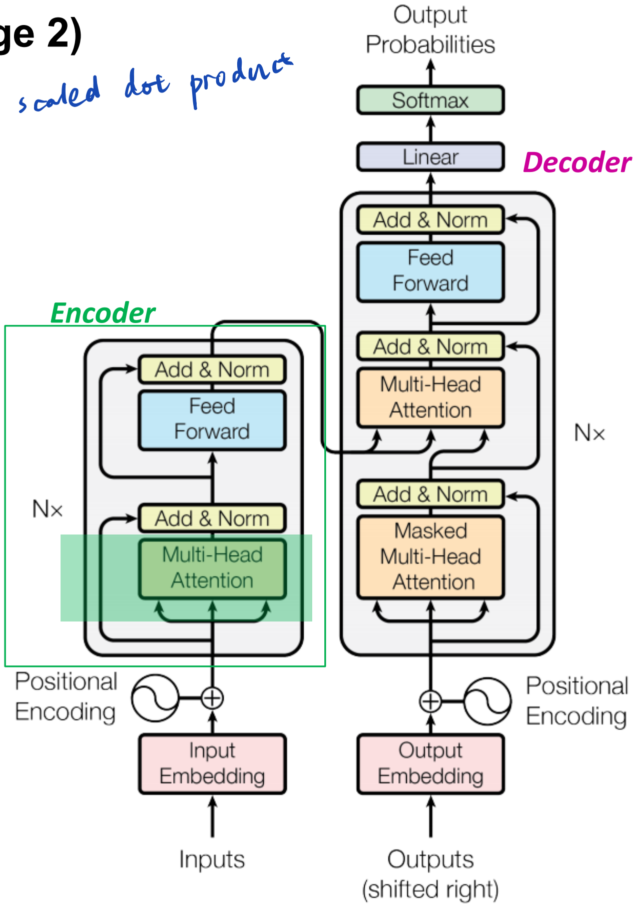
4 Attention and Transformer for MT

The Transformer – Encoder (Stage 2)

why scaled dot product



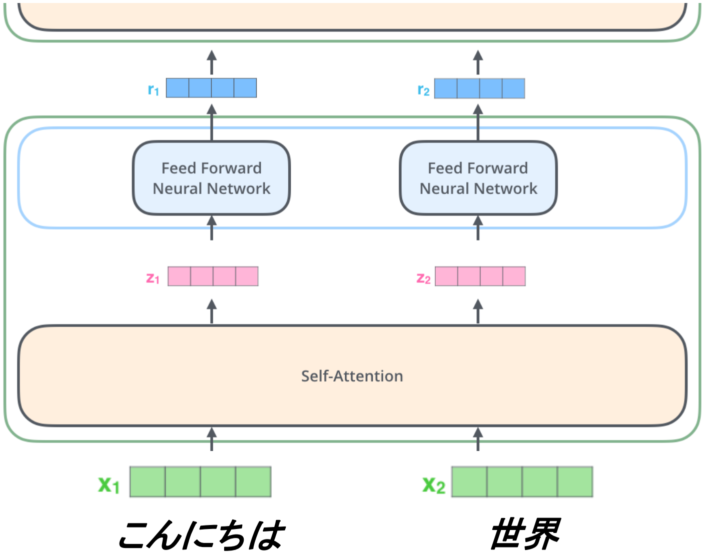
	Input	こんにちは	世界
512 dim	Embedding contain position emb	x_1 🟩🟩🟩🟩	x_2 🟩🟩🟩🟩
64 dim 8 heads	Queries	q_1 🟪🟪🟪	q_2 🟪🟪🟪
	Keys	k_1 🟧🟧🟧	k_2 🟧🟧🟧
	Values	v_1 🟦🟦🟦	v_2 🟦🟦🟦
	Score ↗ 8 heads	$q_1 \bullet k_1 = 112$	$q_1 \bullet k_2 = 96$
	Divide by 8 ($\sqrt{d_k}$)	14	12
	Softmax	0.88	0.12
	Softmax X Value	v_1 🟦🟦🟦	v_2 ▢▢▢
	Sum	z_1 🟪🟪🟪	z_2 🟪🟪🟪

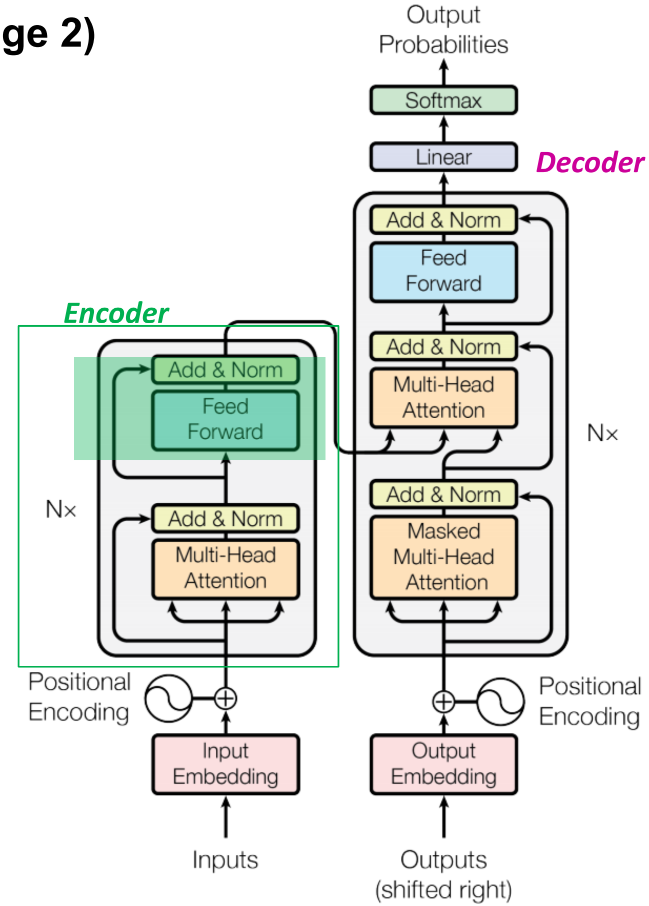


The transformer – model architecture

4 Attention and Transformer for MT

The Transformer – Encoder (Stage 2)

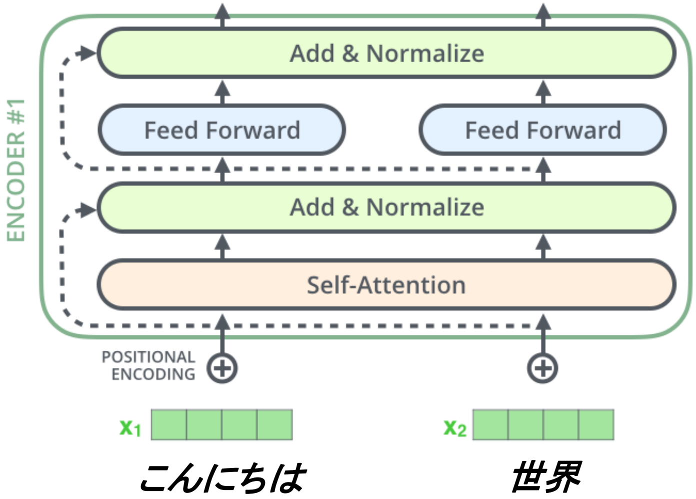


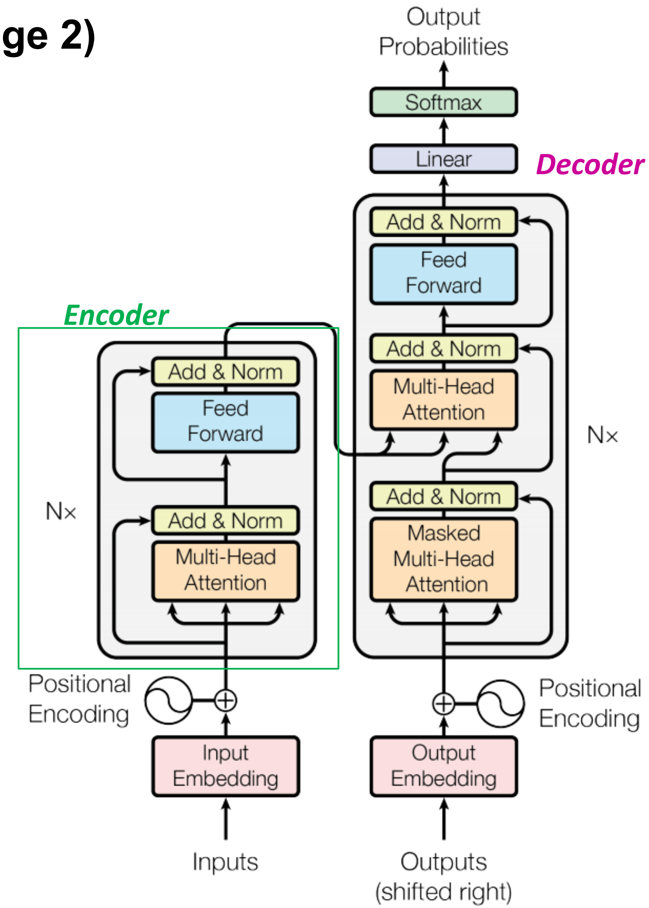


The transformer – model architecture

4 Attention and Transformer for MT

The Transformer – Encoder (Stage 2)

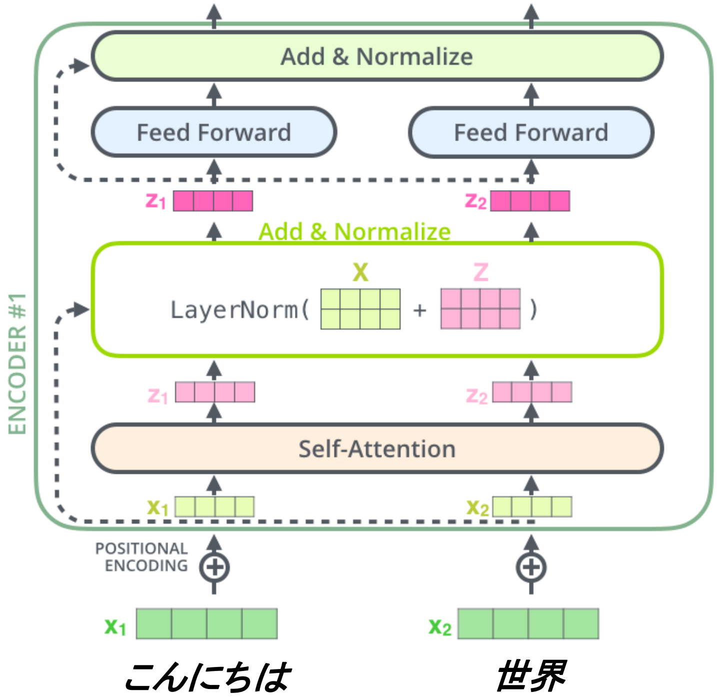


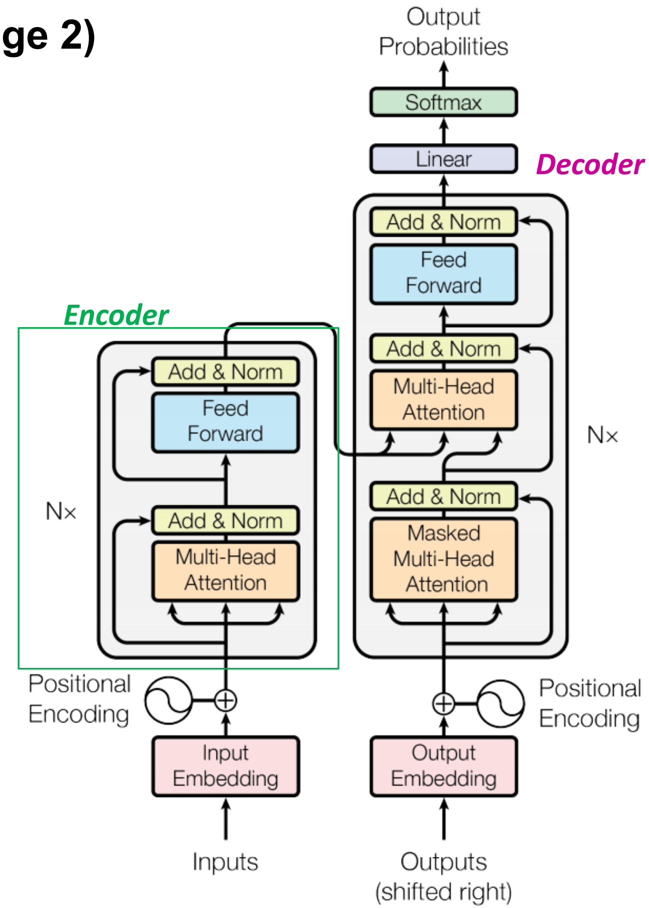


The transformer – model architecture

4 Attention and Transformer for MT

The Transformer – Encoder (Stage 2)

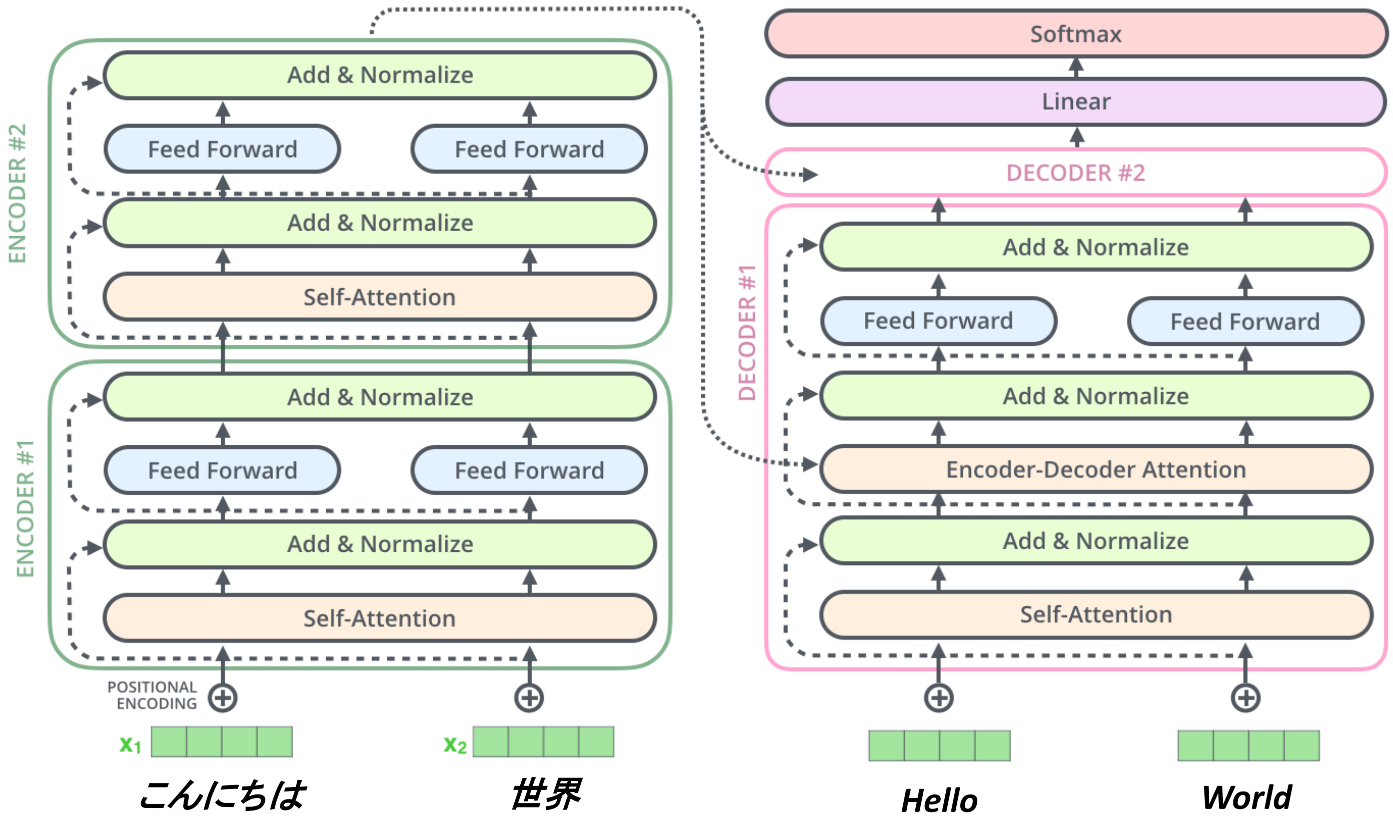




The transformer – model architecture

4 Attention and Transformer for MT

The Transformer – Encoder to Decoder



4 Attention and Transformer for MT

RNN input has state

Transformer all come together

need to hide part for prediction

Masked ? other part

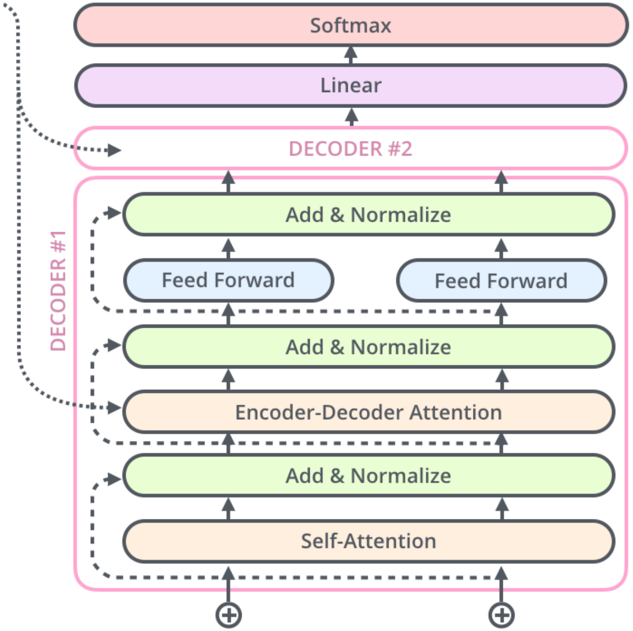
<BOS> xxx .

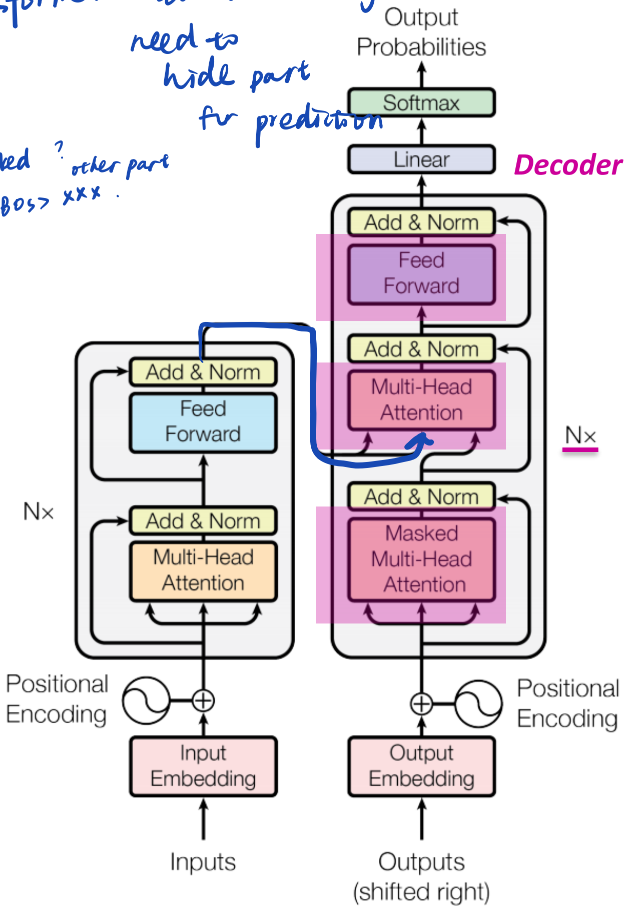
The Transformer - Decoder

Hello

Label [0, 0, ..., 1, 0]

Output[0.1, 0.01,........................., 0.8, 0.1]

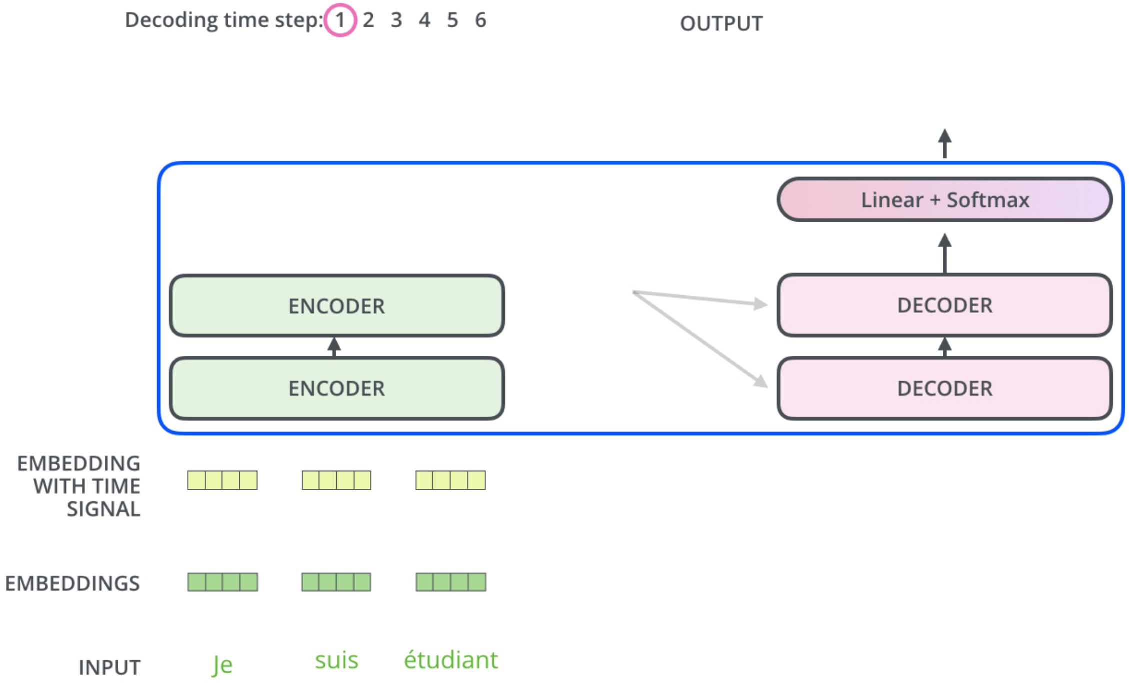




The transformer – model architecture

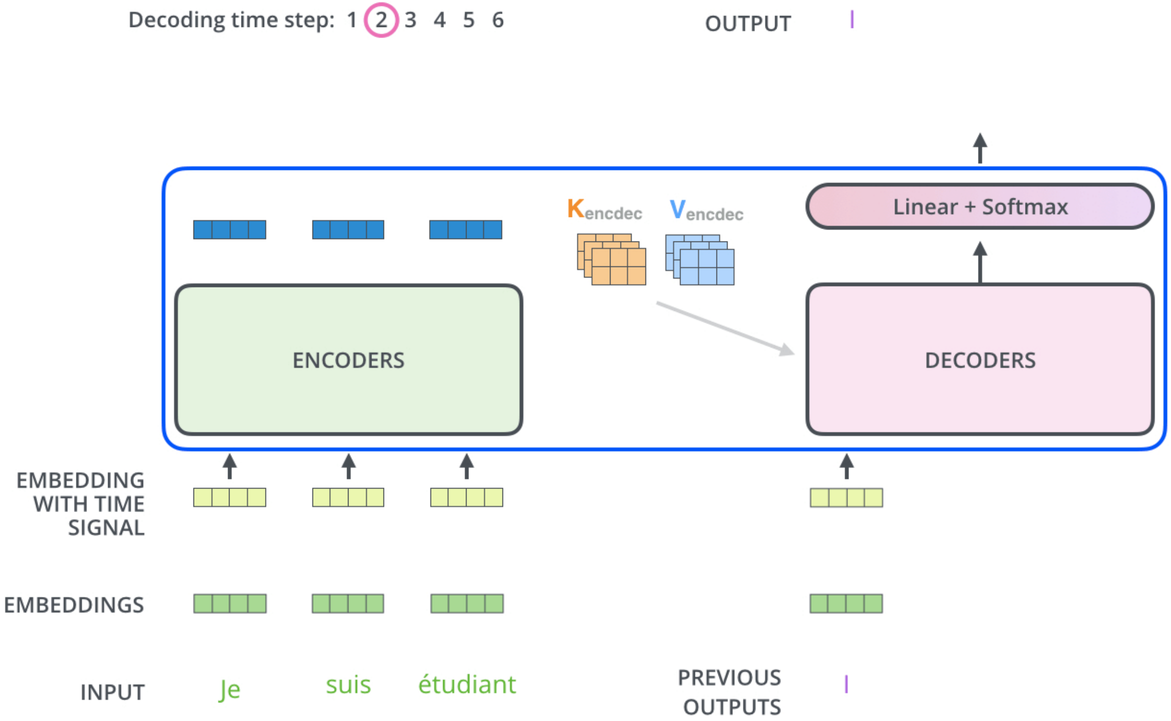
4 Attention and Transformer for MT

The Transformer with example – Encoder to Decoder



4 Attention and Transformer for MT

The Transformer with example – Decoding Phrases



5 The Rise of the Pre-trained Model

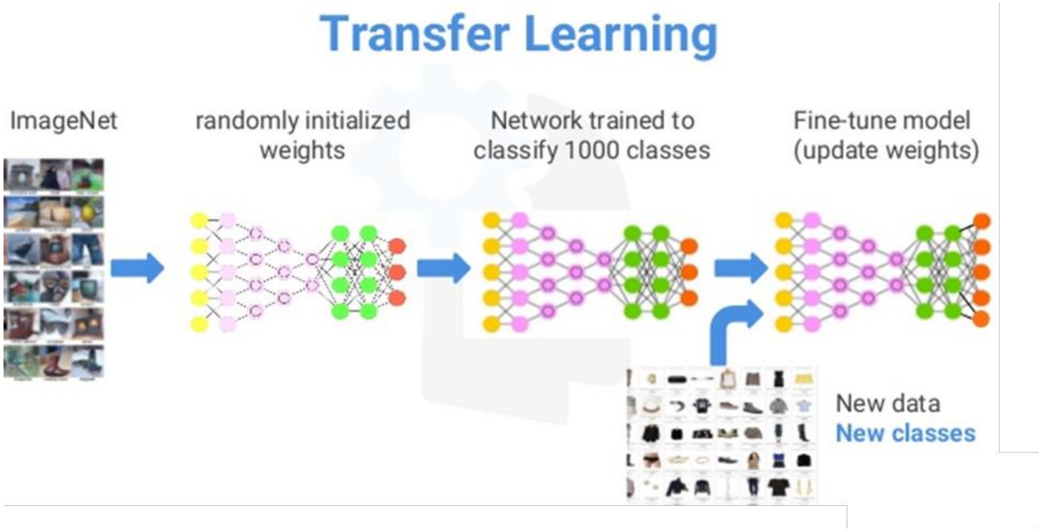
Early 2019 ~

5 The Rise of the Pre-trained Model

Pre-training and Transfer Learning

In computer vision, prove the value of transfer learning

- *pre-training a neural network on a known task (i.e. ImageNet)*
- *performing fine-tuning*
- *using the trained neural network as the basis of a new purpose-specific model.*


Transfer Learning
ImageNet
randomly initialized weights
Network trained to classify 1000 classes
Fine-tune model (update weights)
New data
New classes

5

The Rise of the Pre-trained Model

Pre-training and Transfer Learning in NLP

Popular Pre-trained Model in NLP

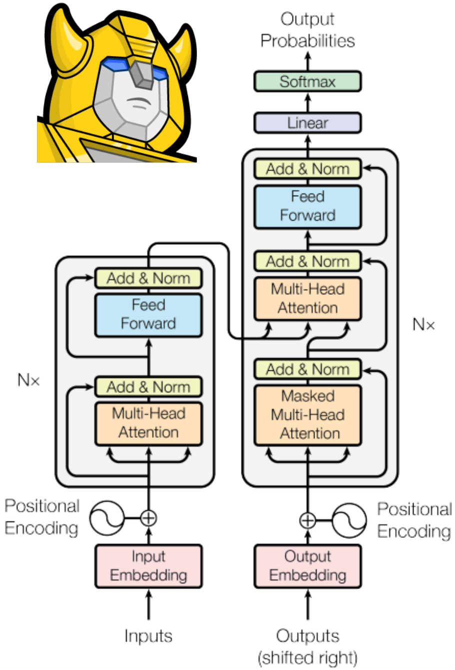
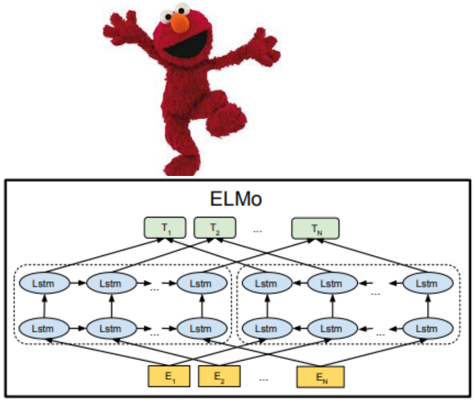
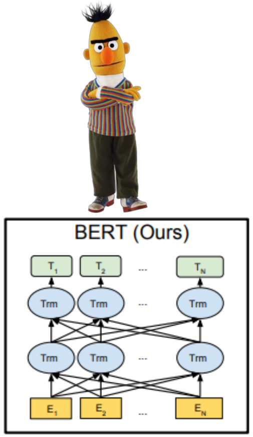


Figure 1: The Transformer - model architecture.



(Peters et al, 2018)



(Devlin et al, 2018)

Using Contextual word representations

5 The Rise of the Pre-trained Model

Pre-training and Transfer Learning in NLP

Popular Pre-trained Model: Contextual Representations

Word embeddings (i.e. word2vec, fastText, GloVe) are applied in a context free manner

Step up to the ***bat*** — ***bat*** *[0.7, 0.2, -0.5, 1.1, …]*

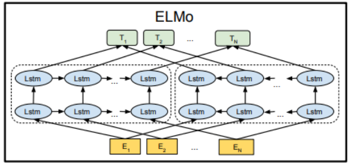
A vampire ***bat*** — ***bat*** *[0.7, 0.2, -0.5, 1.1, …]*

Need to train ***contextual representation*** *on text corpus*

Step up to the ***bat*** — ***bat*** *[1.1, -0.7, 0.8, 2.1, …]*

A vampire ***bat*** — ***bat*** *[0.3, 0.5, -0.9, 1.3, …]*

5 The Rise of the Pre-trained Model



Pre-training and Transfer Learning in NLP

ELMo: Deep Contextual Word Embeddings (2017)

Hey ELMo, what's the embedding of the word "stick"?
There are multiple possible embeddings! Use it in a sentence.
Oh, okay. Here: "Let's stick to improvisation in this skit"
Oh in that case, the embedding is: -0.02, -0.16, 0.12, -0.1 ....etc

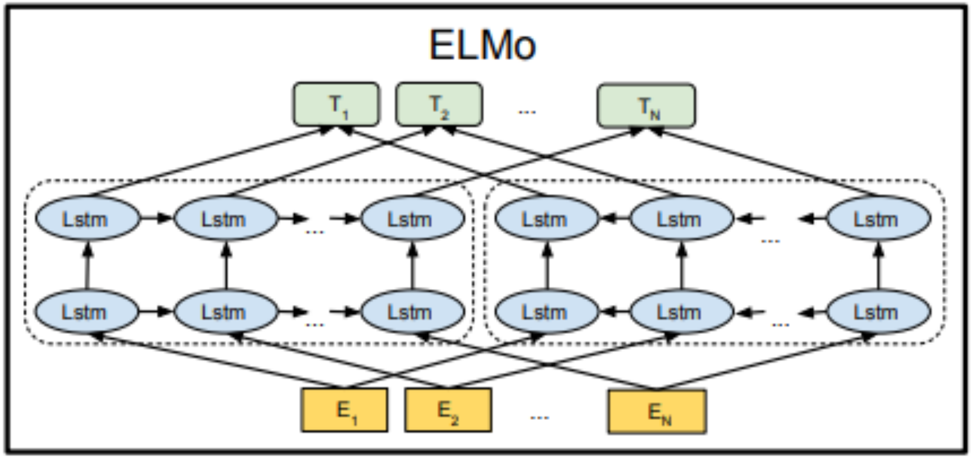
*ELMo provided a **significant step towards pre-training in the context of NLP**. Let's dig in what the ELMo's big secret is!*

5 The Rise of the Pre-trained Model

Pre-training and Transfer Learning in NLP

ELMo: Deep Contextual Word Embeddings (2017)

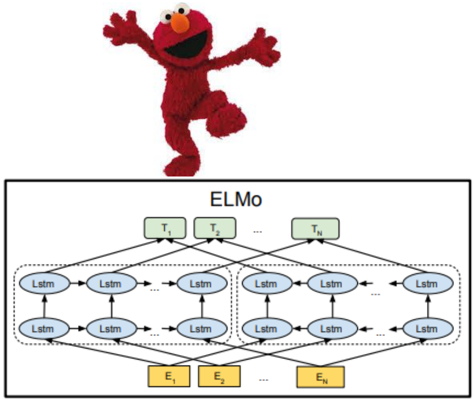
ELMo gained its language understanding from being trained to predict the next word in a sequence of words, Language Modeling Tasks. This is convenient because we have vast amounts of text data that such a model can learn from without needing labels.



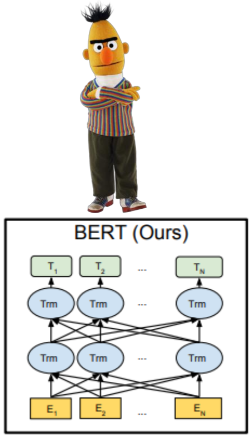
5 The Rise of the Pre-trained Model

Pre-training and Transfer Learning in NLP

ELMo and BERT



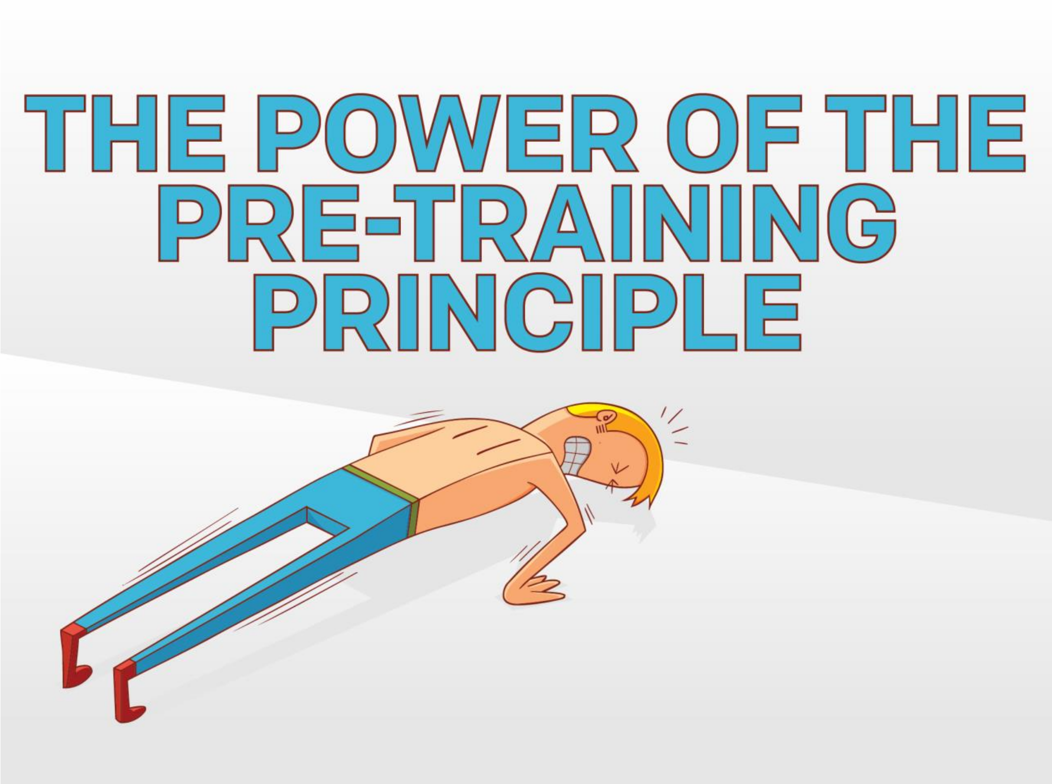
(Peters et al, 2018)



(Devlin et al, 2018)

5 The Rise of the Pre-trained Model

The future of NLP…



COMP5046 Natural Language Processing

What we learned in this course!

- Week 1: Introduction to Natural Language Processing (NLP)
- Week 2: Word Embeddings (Word Vector for Meaning)
- Week 3: Word Classification with Machine Learning I
- Week 4: Word Classification with Machine Learning II

NLP and Machine Learning

- Week 5: Language Fundamental
- Week 6: Part of Speech Tagging
- Week 7: Dependency Parsing
- Week 8: Language Model

NLP Techniques

- Week 9: Information Extraction: Named Entity Recognition
- Week 10: Advanced NLP: Attention and Reading Comprehension
- Week 11: Advanced NLP: Transformer and Machine Translation
- Week 12: Advanced NLP: Pretrained Model

Advanced Topic

- Week 13: Future of NLP and Exam Review

/ Reference

Reference for this lecture

- Deng, L., & Liu, Y. (Eds.). (2018). Deep Learning in Natural Language Processing. Springer.
- Rao, D., & McMahan, B. (2019). Natural Language Processing with PyTorch: Build Intelligent Language Applications Using Deep Learning. " O'Reilly Media, Inc.".
- Manning, C. D., Manning, C. D., & Schütze, H. (1999). Foundations of statistical natural language processing. MIT press.
- Manning, C 2018, Natural Language Processing with Deep Learning, lecture notes, Stanford University

- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. In Advances in neural information processing systems (pp. 5998-6008).
- Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.
- Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep contextualized word representations. arXiv preprint arXiv:1802.05365.
- Miller, A., Fisch, A., Dodge, J., Karimi, A. H., Bordes, A., & Weston, J. (2016). Key-value memory networks for directly reading documents. arXiv preprint arXiv:1606.03126.

- Drawings
- http://jalammar.github.io/illustrated-bert/
- http://jalammar.github.io/illustrated-transformer/