[course site]

#DLUPC

Day 2 Lecture 2

Neural Machine Translation

Marta R. Costa-jussà
marta.ruiz@upc.edu

Ramón y Cajal Researcher
Universitat Politecnica de Catalunya
Technical University of Catalonia

Acknowledgments

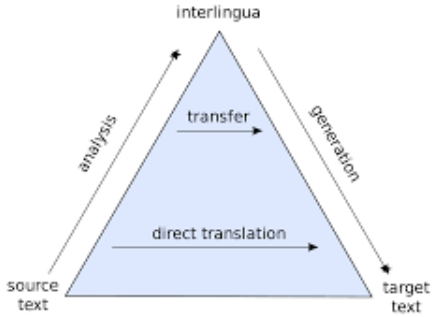
Kyunghyun Cho, NVIDIA BLOGS:
https://devblogs.nvidia.com/parallelforall/introduction-neural-machine-translation-with-gpus/

Graham NEUBIG's lectures

Previous concepts

- Recurrent neural network (LSTM and GRU) (handle variable-length sequences)
- Word embeddings
- Language Modeling (assign a probability to a sentence)

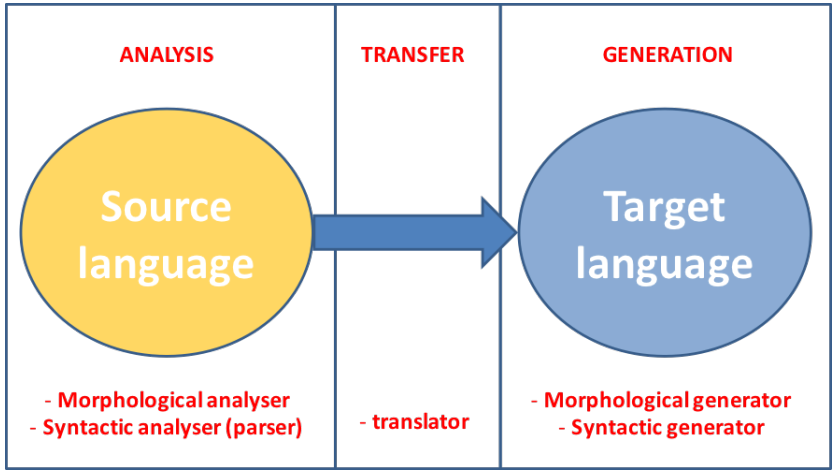
Machine Translation background



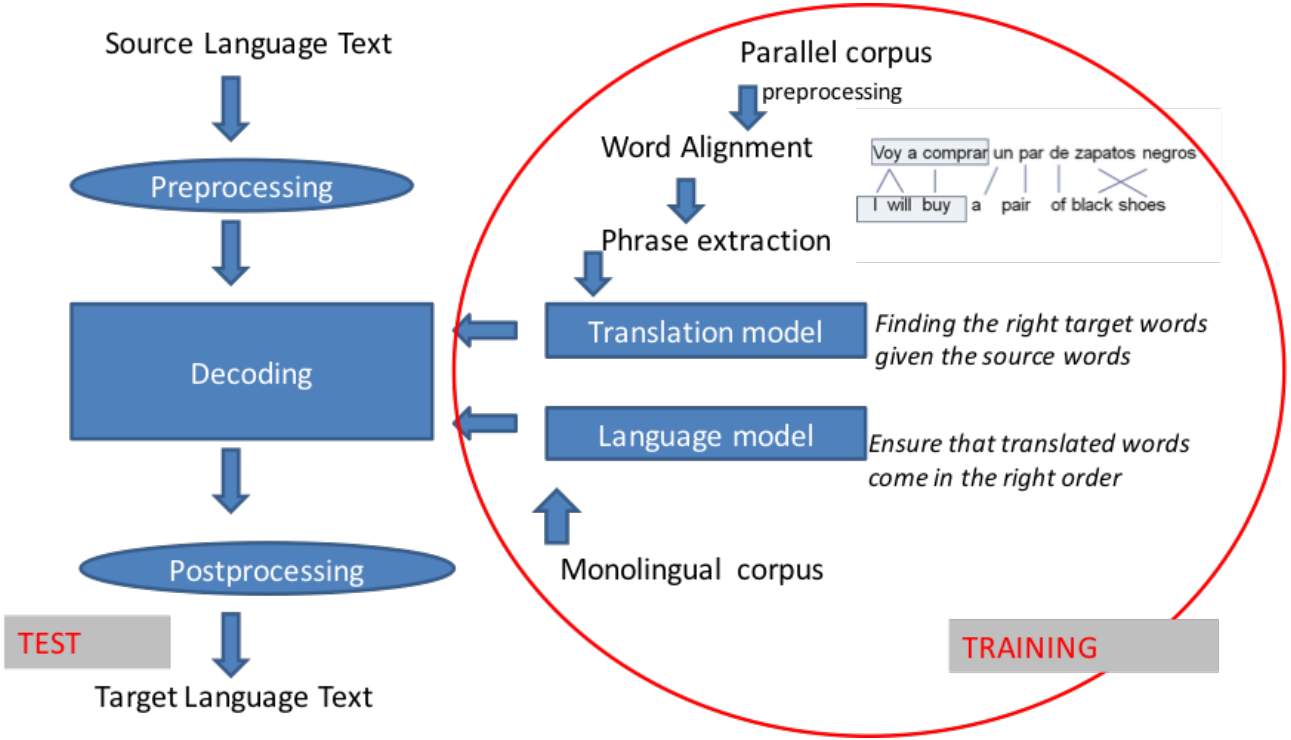
Machine Translation is the application that is able to automatically translate from source (S) to target (T).

Main approaches have been either rule-based or statistical-based

Rule-based approach



Statistical-based approach



Why a new approach?

We need years to develop a nice rule-based approach

Regarding statistical systems:

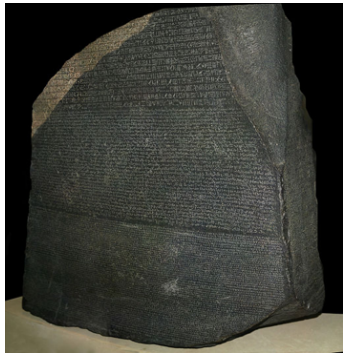
(1) Word alignment and Translation are optimized separately
(2) Translation at the level of words, but difficulties with high variations in morphology (e.g. translation English-to-Finnish)
(3) Translation by language pairs
    (a) difficult to think of an automatic interlingua
    (b) bad performance with low resourced-languages

Why Neural Machine Translation?

- Integrated MT paradigm
- Trainable at the subword/character level
- Multilingual advantages

What do we need?

- Parallel Corpus



English	Russian
This course is a thorough introduction to machine translation technology	Этот курс представляет собой интенсивное введение в технологию машинного перевода
We will describe all aspects of building a statistical machine translation system, from both formal and practical perspectives	Мы рассмотрим все аспекты построения системы статистического машинного перевода с теоретической и практической точки зрения

Same requirement than phrase-based systems

Sources of parallel corpus

- European Plenary Parlament Speeches (EPPS) transcriptions
- Canadian Handsards
- United Nations
- CommonCrawl
- ...

International evaluation campaigns:
Conference on Machine Translation (WMT)
International Workshop on Spoken Language Translation (IWSLT)

Same requirements than phrase-based systems

What else do we need?

Automatic measure

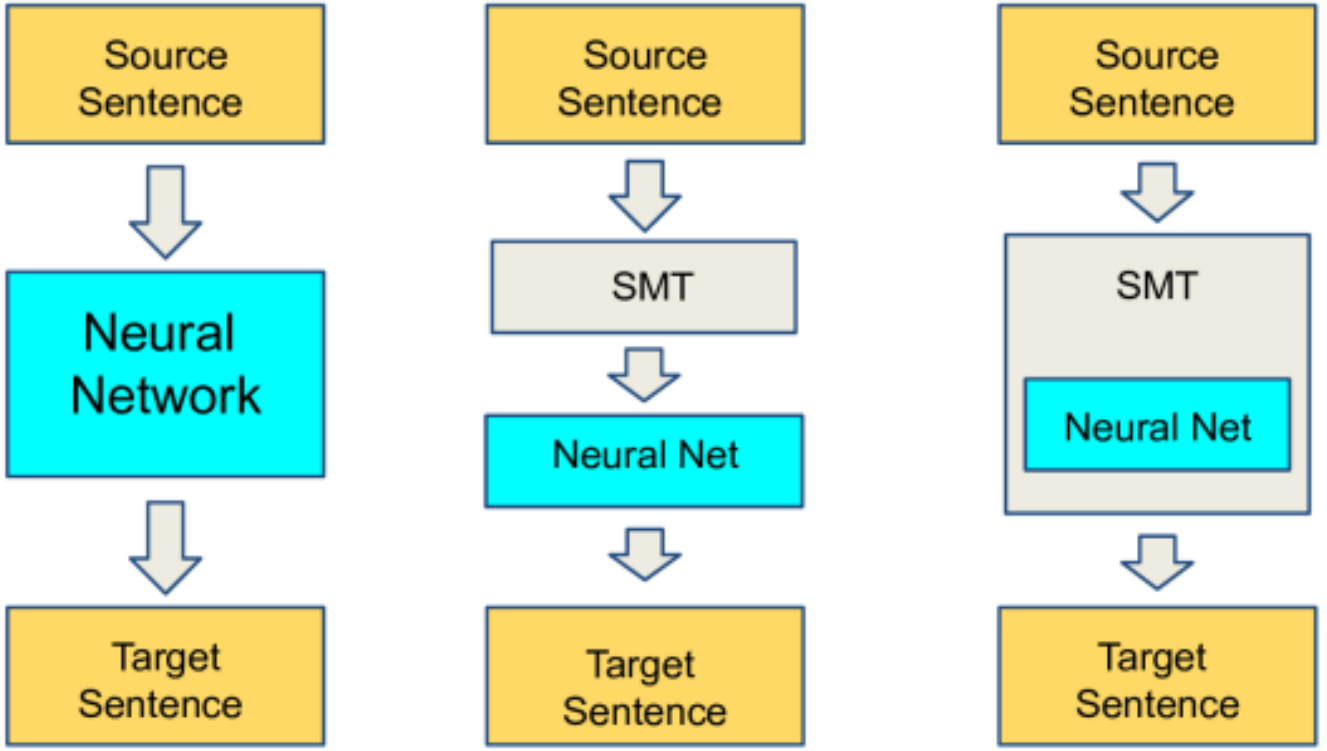
SYSTEM A: Israeli officials (2-GRAM MATCH) responsibility of airport (1-GRAM MATCH) safety

REFERENCE: Israeli officials are responsible for airport security

SYSTEM B: airport security (2-GRAM MATCH) Israeli officials are responsible (4-GRAM MATCH)

Metric	System A	System B
precision (1gram)	3/6	6/6
precision (2gram)	1/5	4/5
precision (3gram)	0/4	2/4
precision (4gram)	0/3	1/3
brevity penalty	6/7	6/7
BLEU	0%	52%

Towards Neural Machine Translation

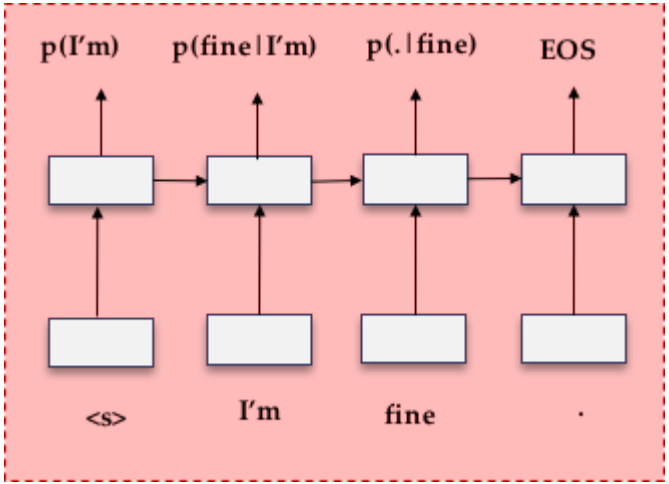


Kyunghyun Cho, "Introduction to Neural Machine Translation with GPUs" (2015)

Sequence modeling

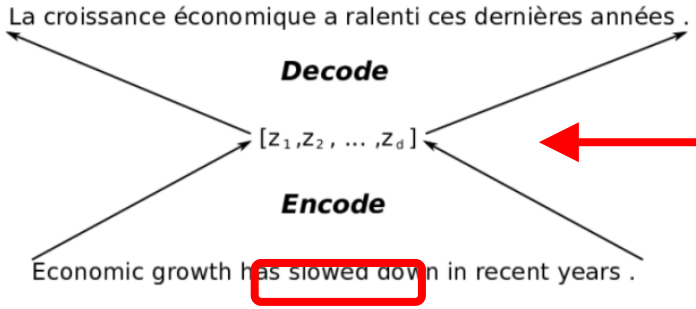
Model the probability of sequences of words

From previous lecture… we model sequences with RNNs

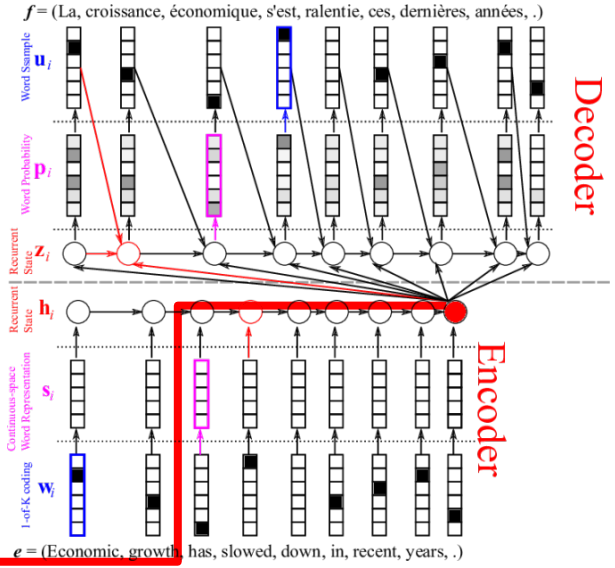


Encoder-Decoder

Front View



Side View

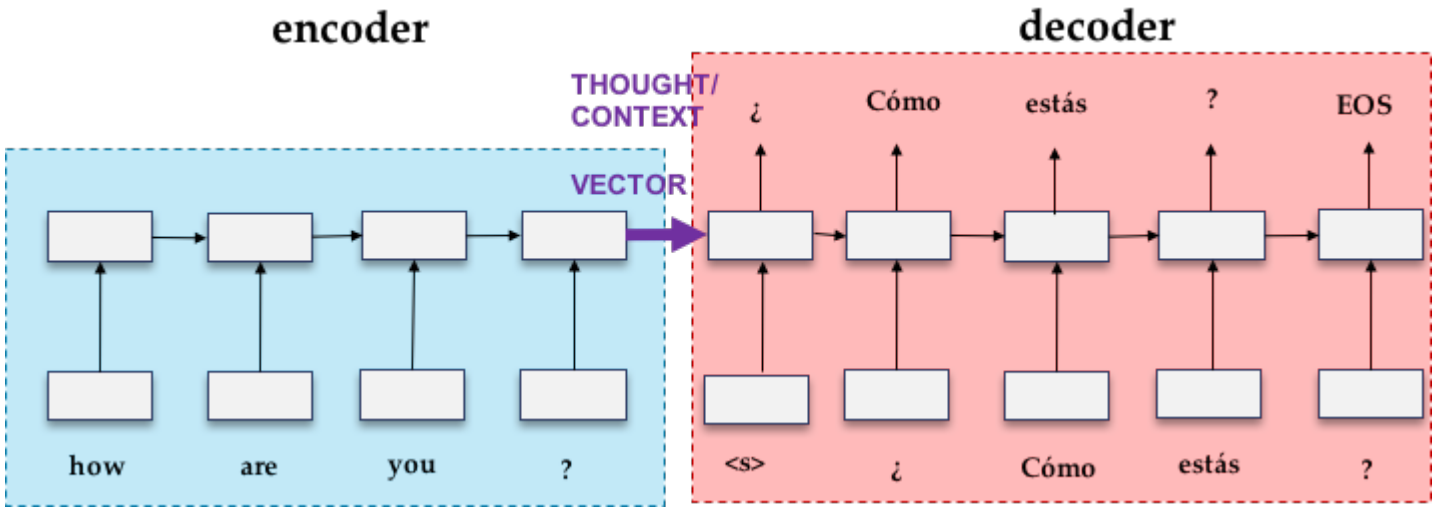


Representation of the sentence

Kyunghyun Cho, "Introduction to Neural Machine Translation with GPUs" (2015)

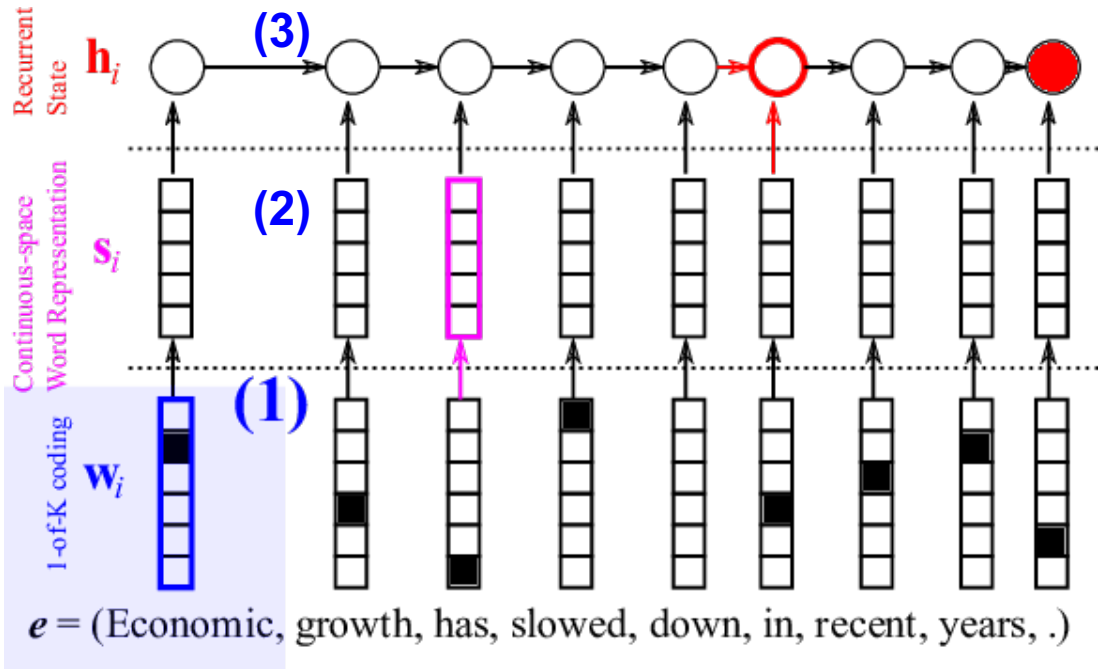
Cho, Kyunghyun, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. "Learning phrase representations using RNN encoder-decoder for statistical machine translation." arXiv preprint arXiv:1406.1078 (2014).

Sequence-to-sequence



Encoder

Encoder in three steps



(1) One hot encoding
(2) Continuous space representation
(3) Sequence summarization

Step 1: One-hot encoding

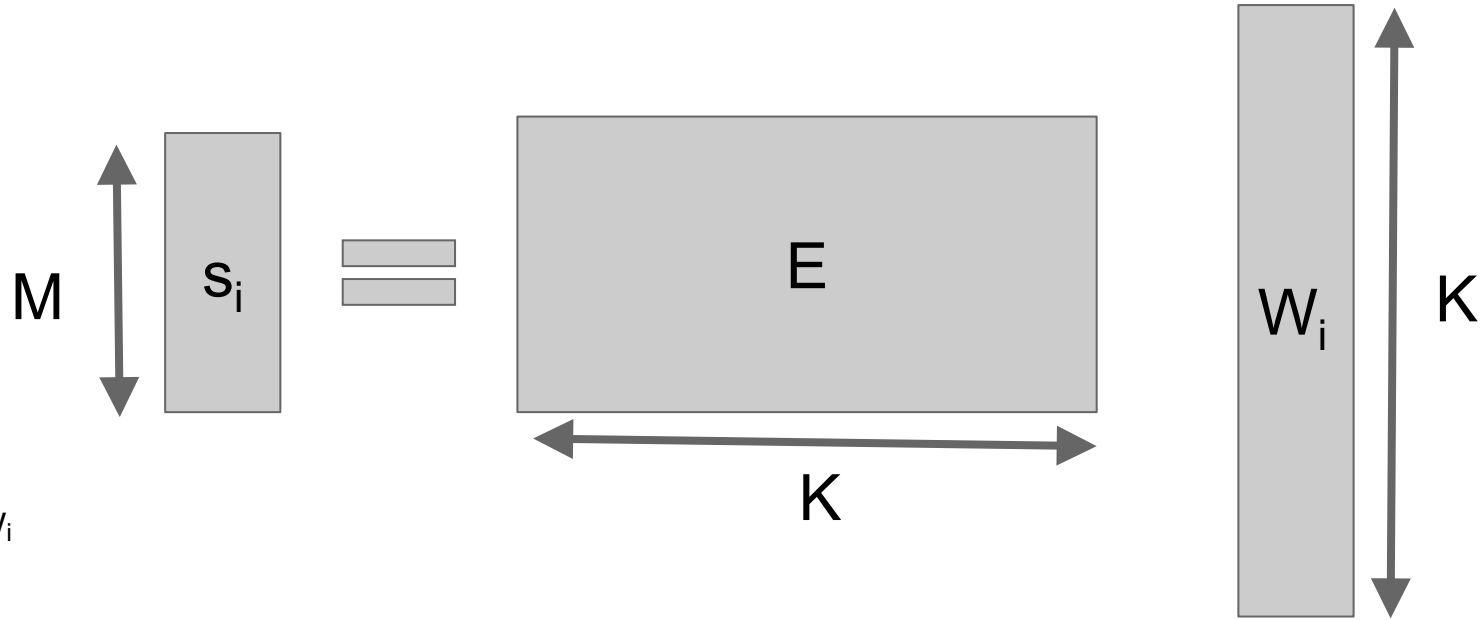
From previous lecture on language modeling

Natural language words can also be one-hot encoded on a vector of dimensionality equal to the size of the dictionary (K).

Word	One-hot encoding
economic	000010...
growth	001000...
has	100000...
slowed	000001...

Step 2: Projection to continuous space

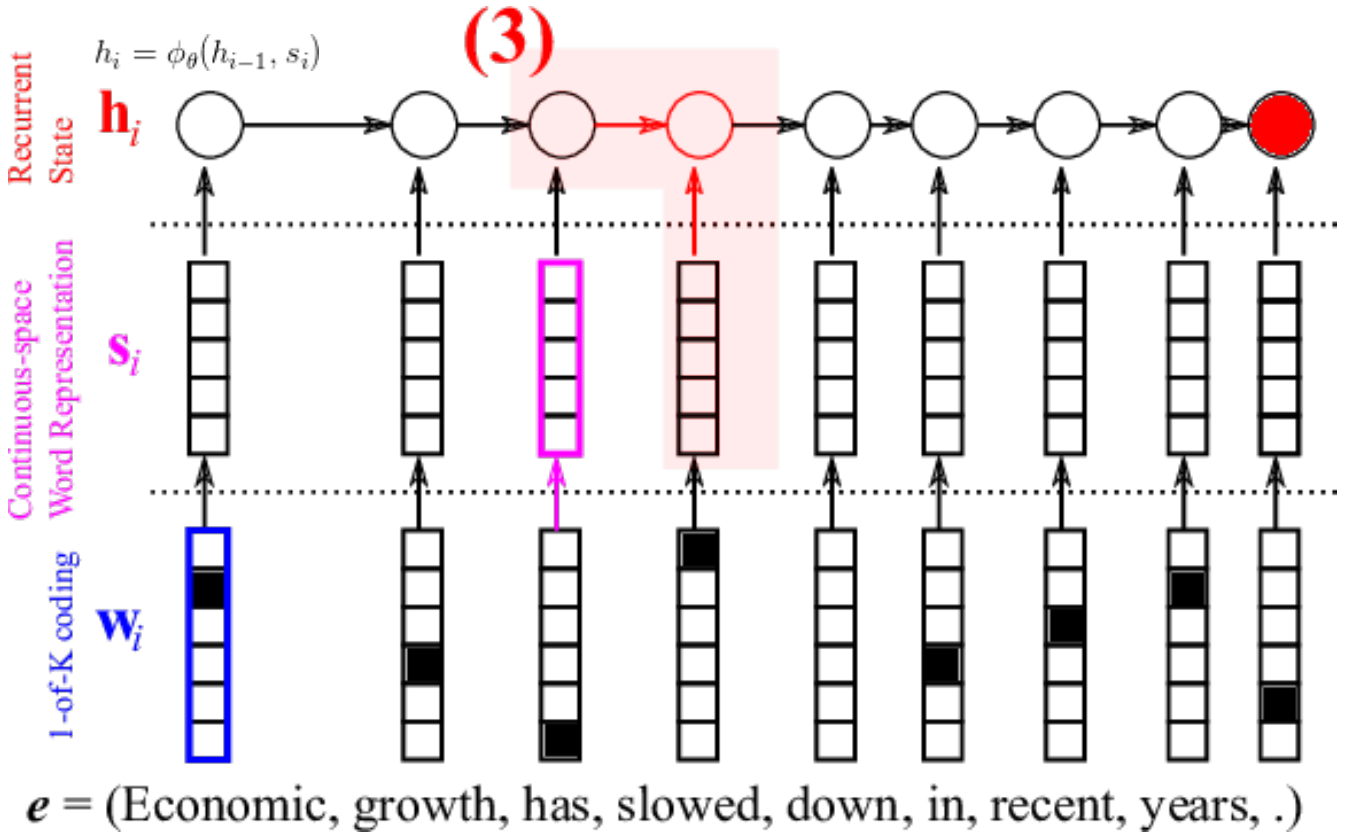
The one-hot is linearly projected to a space of lower dimension (typically 100-500) with matrix E for learned weights.



$s_i = Ew_i$

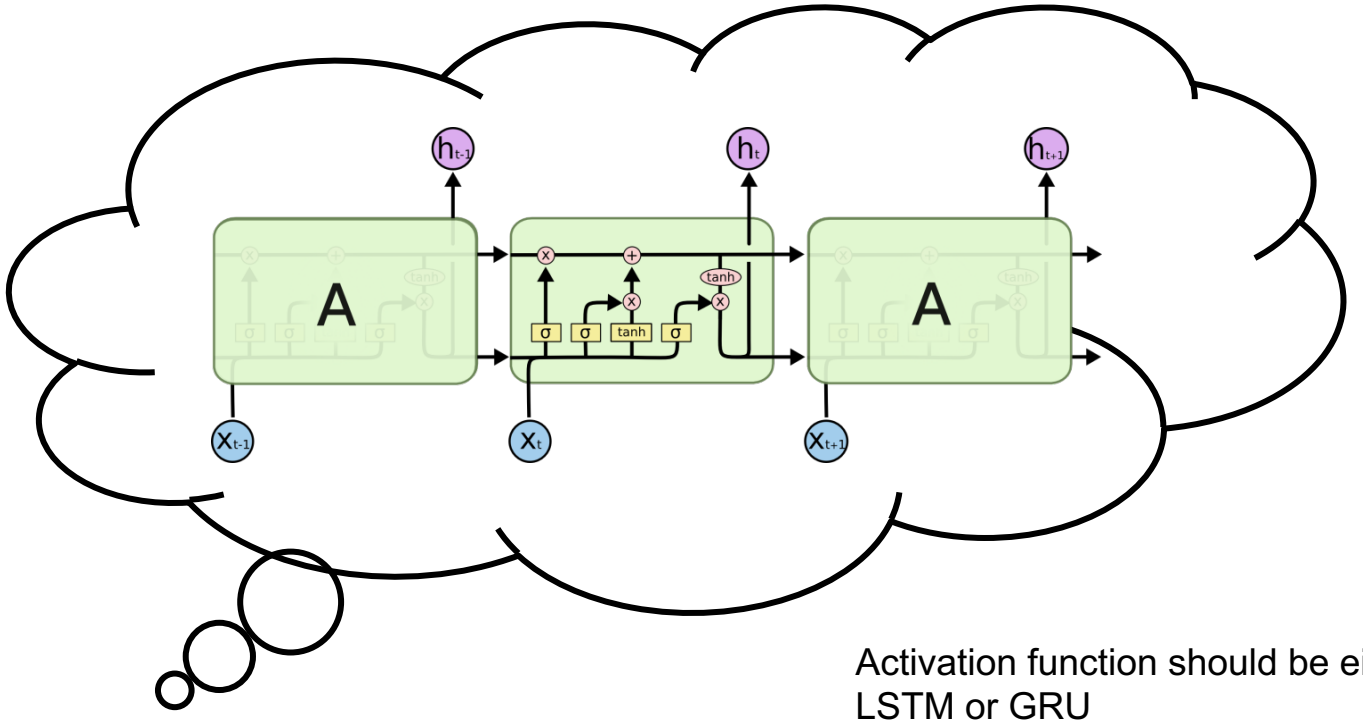
Kyunghyun Cho, "Introduction to Neural Machine Translation with GPUs" (2015)

Step 3: Recurrence



Kyunghyun Cho, "Introduction to Neural Machine Translation with GPUs" (2015)

Step 3: Recurrence



Sequence

Activation function should be either LSTM or GRU

Figure: Cristopher Olah, "Understanding LSTM Networks" (2015)

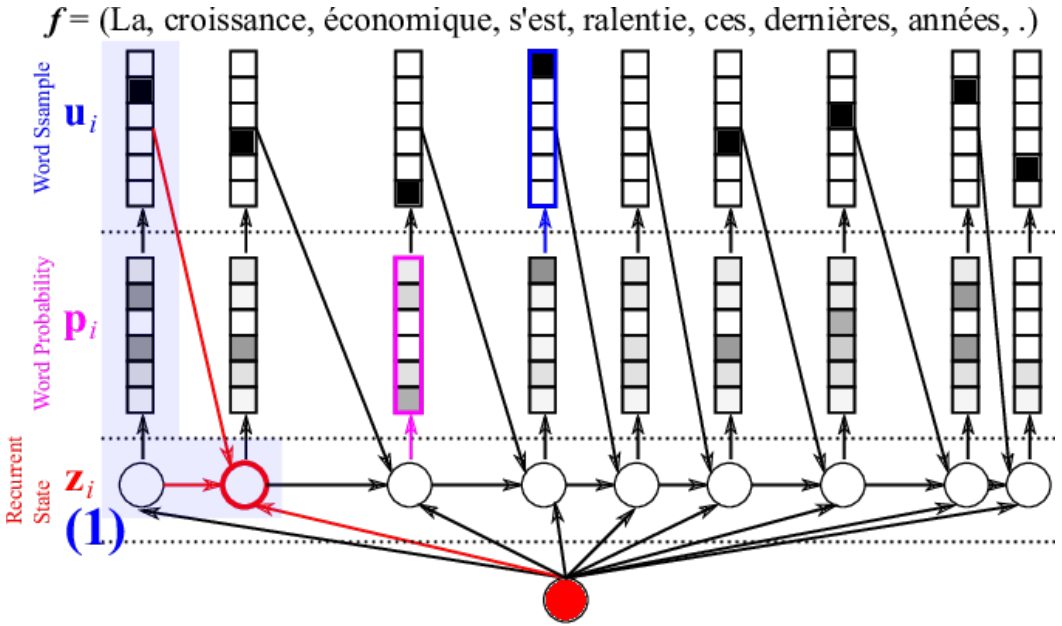
Decoder

Decoder

RNN's internal state z_i depends on: summary vector h_t, previous output word u_{i-1} and previous internal state z_{i-1}.

NEW INTERNAL STATE

$$z_i = \phi_{\theta'}(h_T, u_{i-1}, z_{i-1}).$$



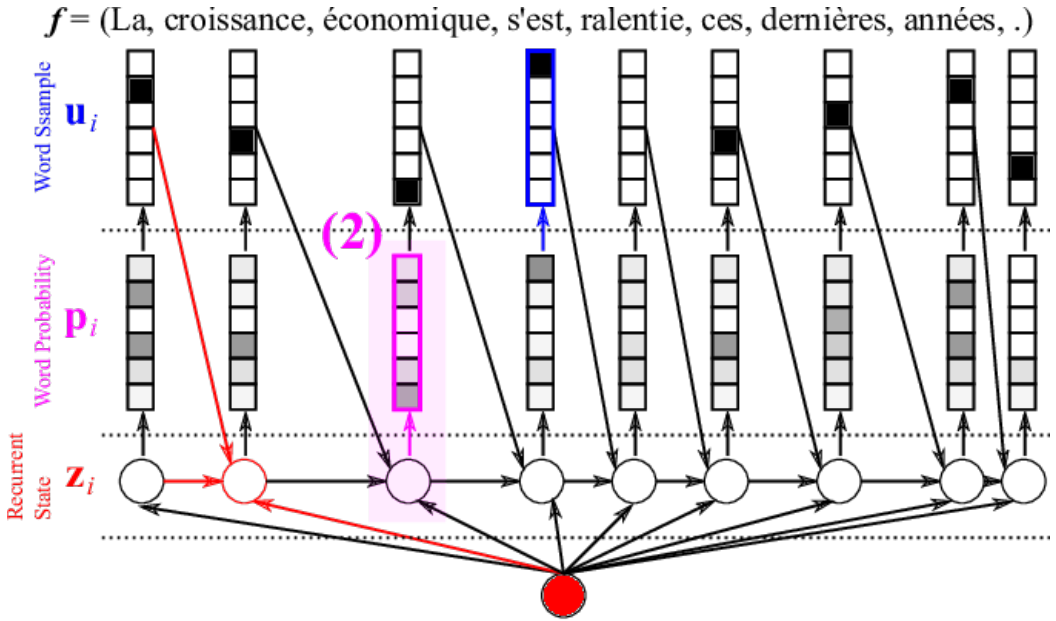
Decoder

With z_i ready, we can score each word *k* in the vocabulary with a dot product given this hidden state...

$$e(k) = w_k^\top z_i + b_k,$$

Neuron weights for word k

RNN internal state



Kyunghyun Cho, "Introduction to Neural Machine Translation with GPUs" (2015)

Decoder

A score is higher if word vector w_k and the decoder's internal state z_i are similar to each other.

$$e(k) = w_k^\top z_i + b_k,$$

Neuron weights for word k

RNN internal state

Remember:
a dot product gives the length of the projection of one vector onto another. Similar vectors (nearly parallel) the projection is longer than if they are very different (nearly perpendicular)

Kyunghyun Cho, "Introduction to Neural Machine Translation with GPUs" (2015)

Decoder

Given the score for word k

$$e(k) = w_k^\top z_i + b_k,$$

...we can finally normalize to word probabilities with a softmax.

Probability that the ith word is word k

$$p(w_i = k | w_1, w_2, \ldots, w_{i-1}, h_T) = \frac{\exp(e(k))}{\sum_j \exp(e(j))}.$$

Previous words Hidden state

Bridle, John S. "Training Stochastic Model Recognition Algorithms as Networks can Lead to Maximum Mutual Information Estimation of Parameters." NIPS 1989

Decoder

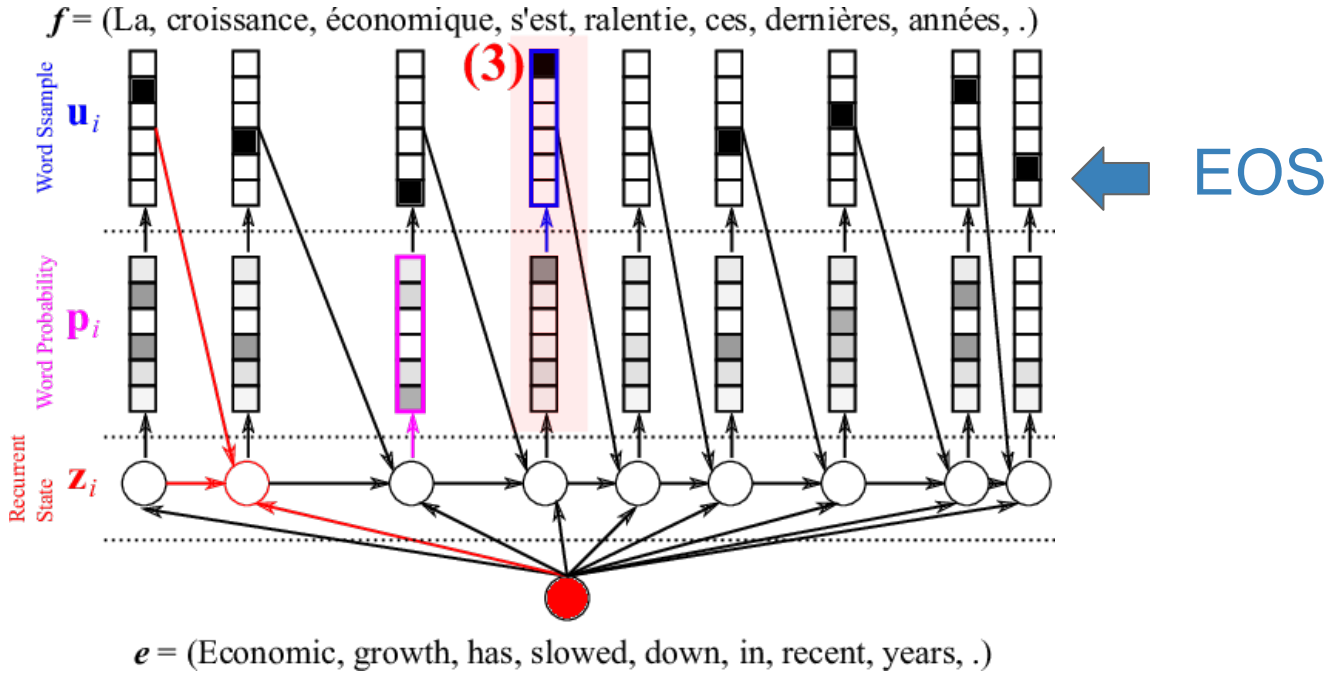
go back to the 1st step…

(1)computing the decoder's internal state
(2)score and normalize target words
(3)select the next word

Kyunghyun Cho, "Introduction to Neural Machine Translation with GPUs" (2015)

Decoder

More words for the decoded sentence are generated until a <EOS> (End Of Sentence) “word” is predicted.



Kyunghyun Cho, “Introduction to Neural Machine Translation with GPUs” (2015)

Training

Training: Maximum Likelihood Estimation

(1) Prepare the parallel corpus, each sample in the corpus is a pair (X^n, Y^n) of source and target

(2) Given any pair from the corpus, the NMT model can compute the conditional log-probability

$\log P(Y^n|X^n, \theta)$, and the log-likelihood of the whole training corpus:

$$\mathcal{L}(D, \theta) = \frac{1}{N}\sum_{n=1}^{N} \log P(Y^n|X^n, \theta)$$

(3) Maximize this log likelihood function, e.g. using stochastic gradient descent (SGD), Adam, Adadelta, Adagrad.. By using backpropagation

Kyunghyun Cho, "Introduction to Neural Machine Translation with GPUs" (2015)

Computational Complexity

1. Source word embeddings: $T \times |V|$ (T source words, $|V|$ unique words)
2. Source embeddings to the encoder: $T \times n_e \times (3 \times n_r)$ (n_e-dim embedding, n_r recurrent units; two gates and one unit for GRU)
3. h_{t-1} to h_t: $T \times n_r \times (3 \times n_r)$
4. Context vector to the decoder: $T \times n_r \times (3 \times n_r)$
5. z_{t-1} to z_t: $T \times n_r \times (3 \times n_r)$
6. The decoder to the target word embeddings: $T' \times n_r \times n_{e'}$ (T' target words, $n_{e'}$-dim target embedding)
7. Target embeddings to the output: $T' \times n_{e'} \times |V'|$ ($|V'|$ target words)
8. Softmax normalization of the output: $T' \times |V'|$

Kyunghyun Cho, "Introduction to Neural Machine Translation with GPUs" (2015)

Why this may not work?

Why this may not work?

We are encoding the entire source sentence into a single context vector

How to solve this?

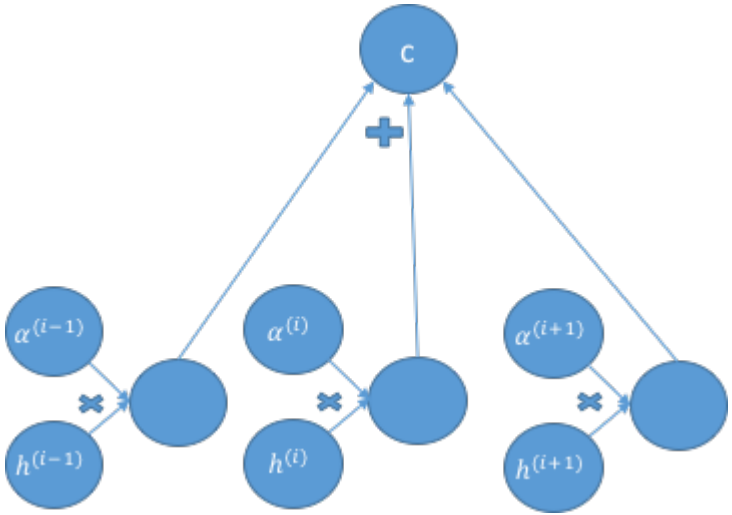
With the attention-based mechanism…

Attention-based mechanism

Read the whole sentence, then produce the translated words one at a time, each time focusing on a different part of the input sentence

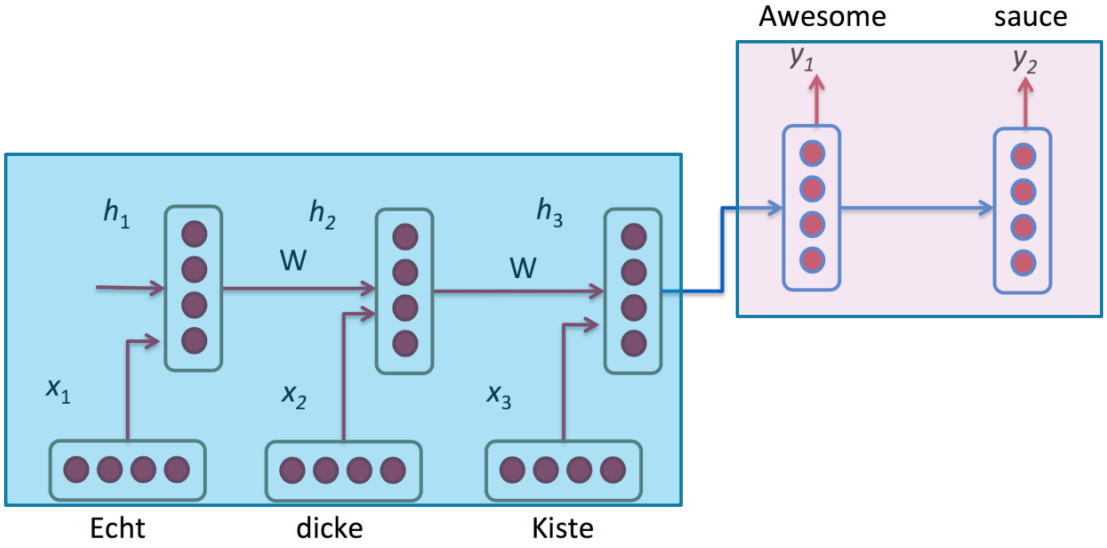
Encoder with attention: context vector

GOAL: Encode a source sentence into a set of context vectors

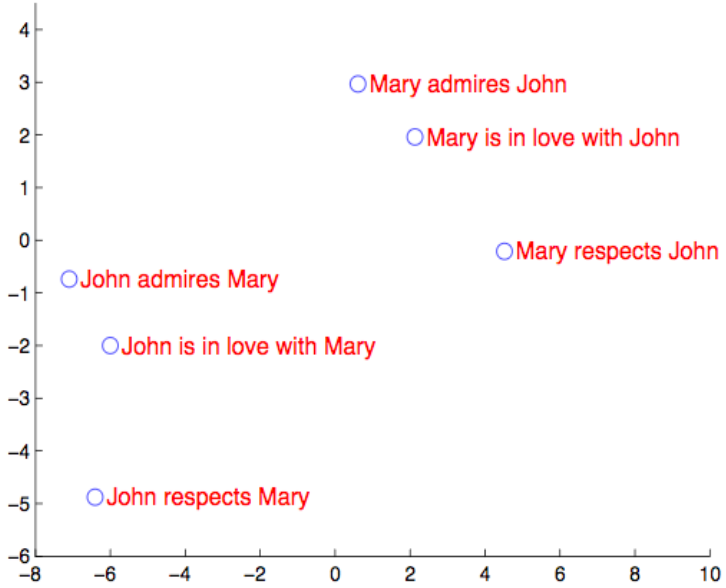


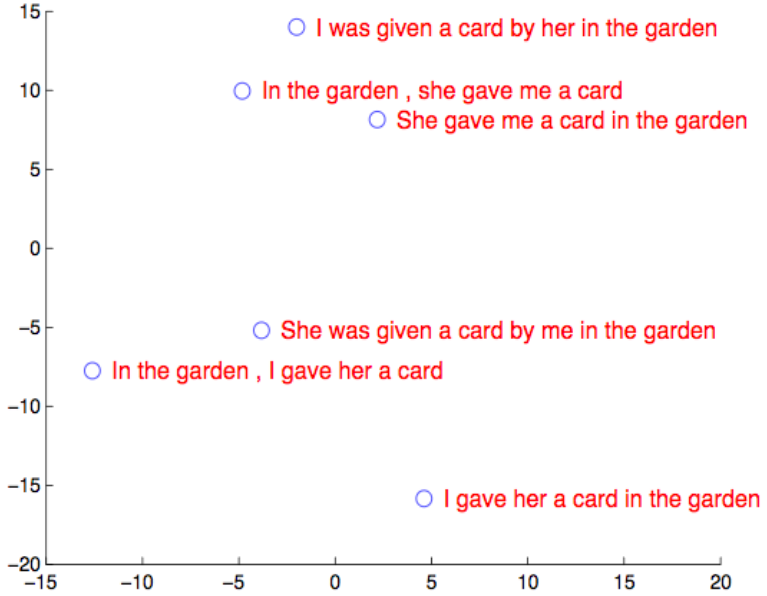
http://www.deeplearningbook.org/contents/applications.html

Motivation in the case of MT

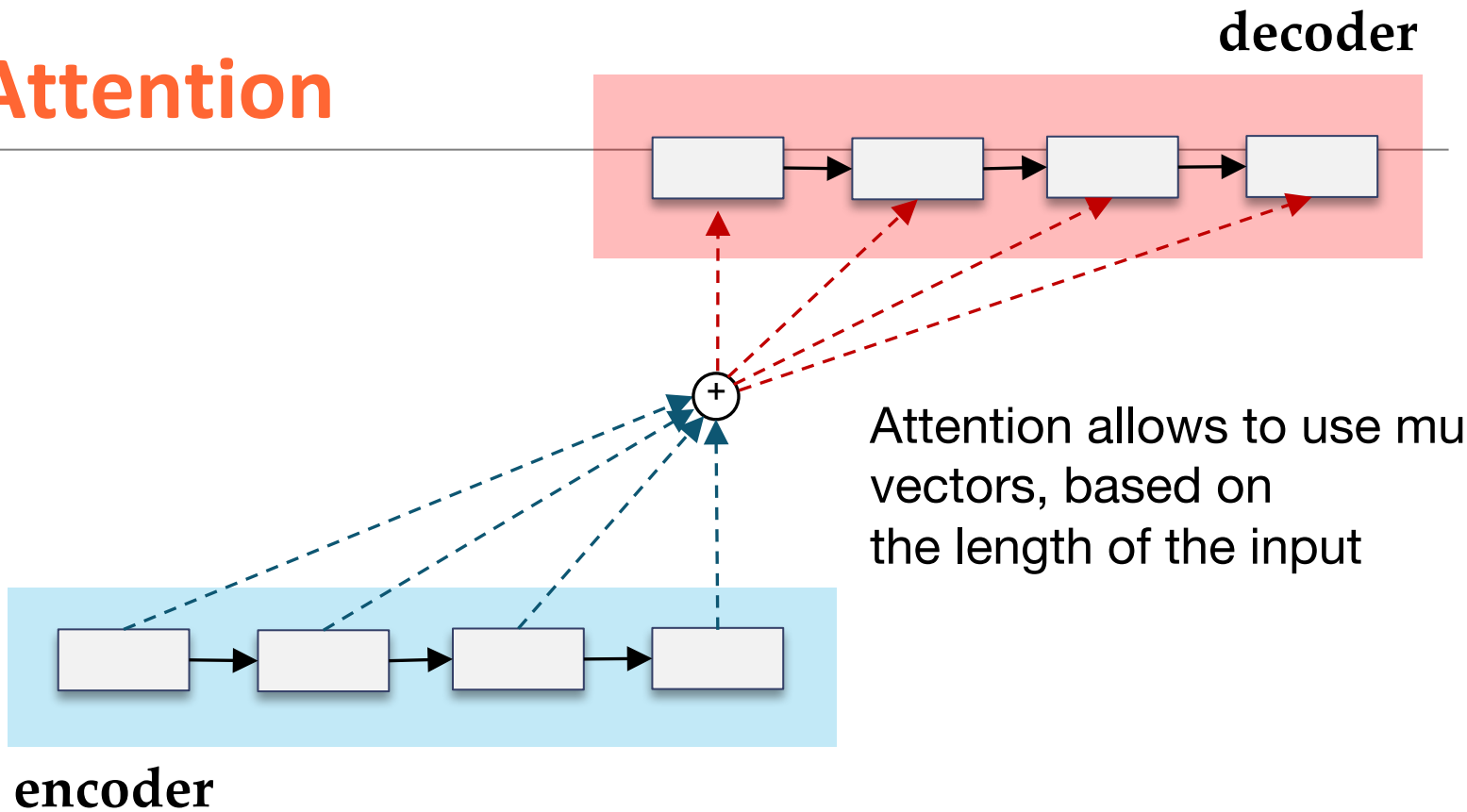


Motivation in the case of MT





Attention



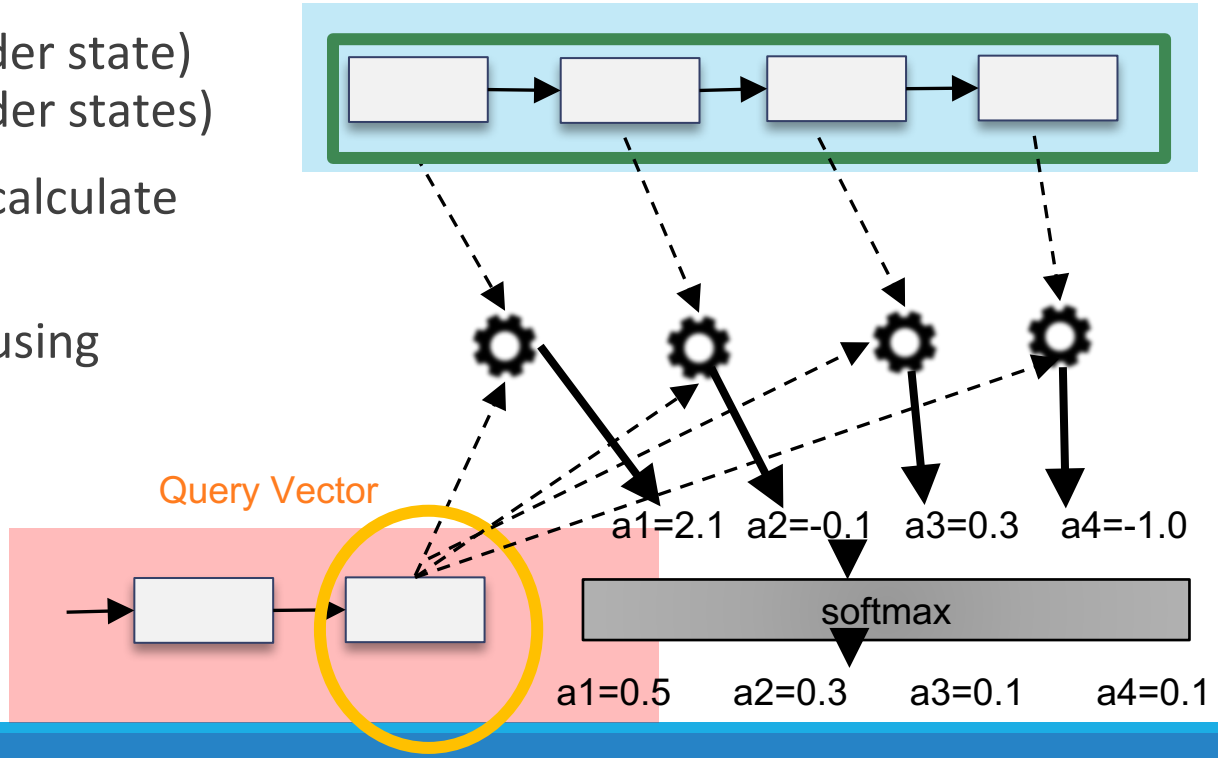
Attention allows to use multiple vectors, based on the length of the input

Attention Key Ideas

- Encode each word in the input and output sentence into a vector
- When decoding, perform a linear combination of these vectors, weighted by “attention weights”
- Use this combination in picking the next word

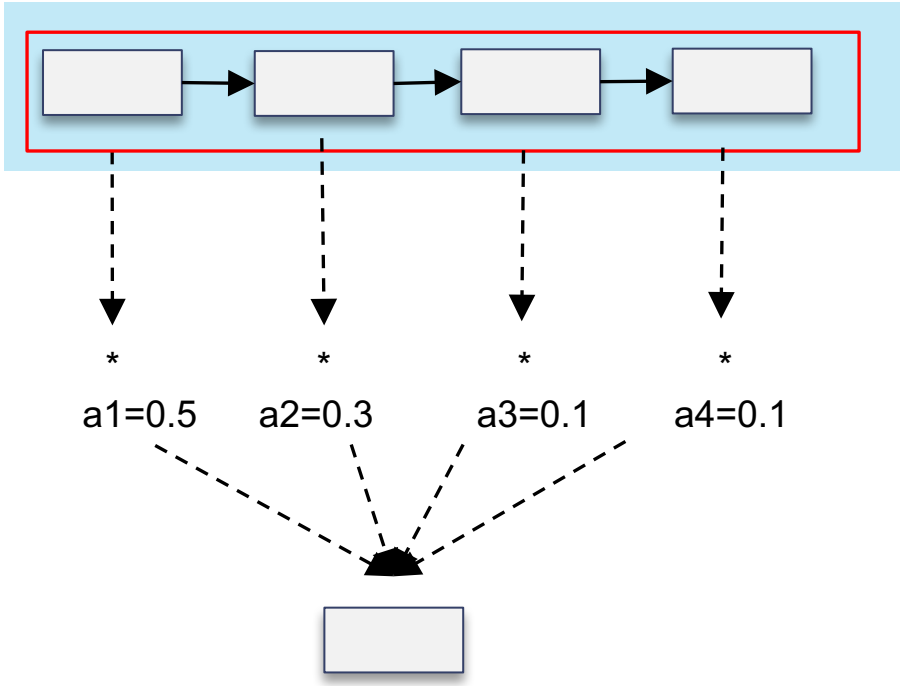
Attention computation I

- Use “query” vector (decoder state) and “key” vectors (all encoder states)
- For each query-key pair, calculate weight
- Normalize to add to one using softmax

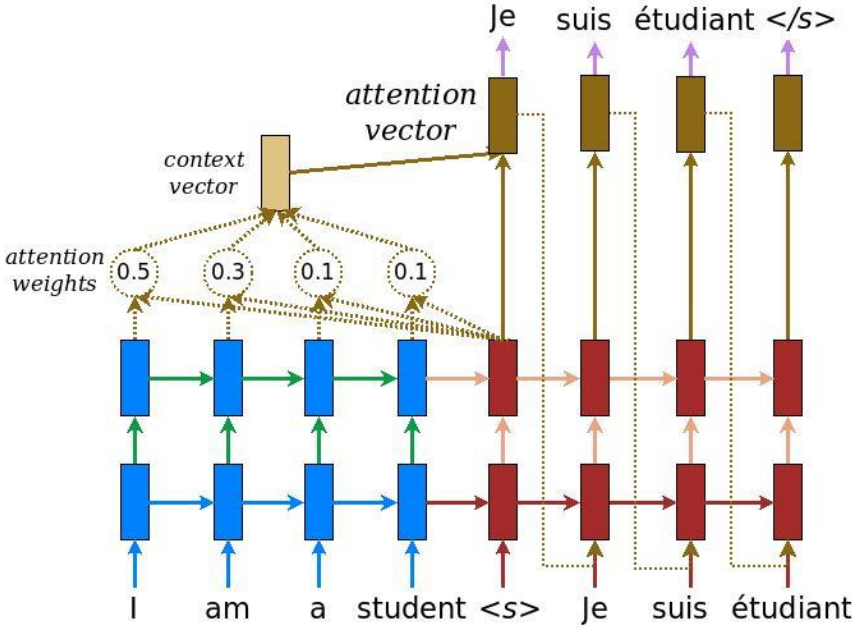


Attention computation II

- Combine together value vectors (usually encoder states, like key vectors) by taking the weighted sum



Attention Integration

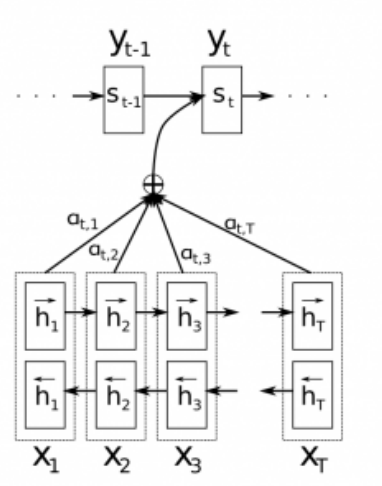


Attention Score Functions

q is the query and k is the key

			Reference
Multi-layer Perceptron	a(q,k)=tanh(W_1 [q,k])	Flexible, often very good with large data	Bahdanau et al., 2015
Bilinear	a(q,k)=q^T Wk		Luong et al 2015
Dot Product	a(q,k)=q^T k	No parameters! But requires sizes to be the same	Luong et al. 2015
Scaled Dot Product	a(q,k)=(q^T k)/√(|k|)	Scale by size of the vector	Vaswani et al. 2017

Attention Integration



$$\alpha_{ts} = \frac{\exp\left(\text{score}(\boldsymbol{h}_t, \bar{\boldsymbol{h}}_s)\right)}{\sum_{s'=1}^{S} \exp\left(\text{score}(\boldsymbol{h}_t, \bar{\boldsymbol{h}}_{s'})\right)} \quad \text{[Attention weights]} \quad (1)$$

$$\boldsymbol{c}_t = \sum_s \alpha_{ts} \bar{\boldsymbol{h}}_s \quad \text{[Context vector]} \quad (2)$$

$$\boldsymbol{a}_t = f(\boldsymbol{c}_t, \boldsymbol{h}_t) = \tanh(\boldsymbol{W}_{\boldsymbol{c}}[\boldsymbol{c}_t; \boldsymbol{h}_t]) \quad \text{[Attention vector]} \quad (3)$$

Improvements in Attention

IN THE CONTEXT OF MT

Coverage

Problem: Neural models tends to drop or repeat content

In MT,

1.Over-translation: some words are unnecessarily translated for multiple times;

2. Under-translation: some words are mistakenly untranslated.

SRC: Señor Presidente, abre la sesión.

TRG: Mr President Mr President Mr President.

Solution: Model how many times words have been covered e.g. maintaining a coverage vector to keep track of the attention history (Tu et al., 2016)

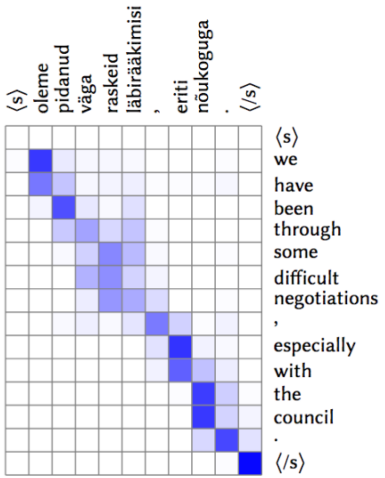
Modeling Coverage for Neural Machine Translation

Zhaopeng Tu[†] Zhengdong Lu[†] Yang Liu[‡] Xiaohua Liu[†] Hang Li[†]

Incorporating Markov Properties

Intuition: Attention from last time tends to be correlated with attention this time

Approach: Add information about the last attention when making the next decision



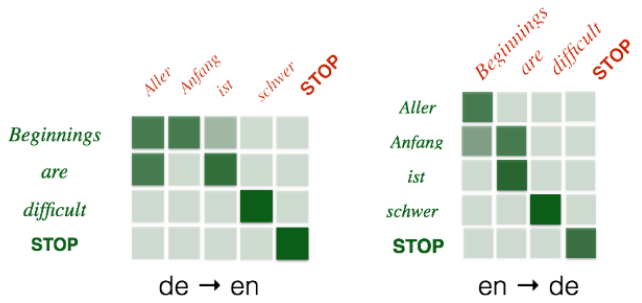
Incorporating Structural Alignment Biases into an Attentional Neural Translation Model

Trevor Cohn and **Cong Duy Vu Hoang** and **Ekaterina Vymolova**

Bidirectional Training

-Background: Established that for latent variable translation models the alignments improve if both directional models are combined (koehn et al, 2005)

-Approach: joint training of two directional models



Incorporating Structural Alignment Biases into an Attentional Neural Translation Model

Trevor Cohn and **Cong Duy Vu Hoang** and **Ekaterina Vymolova**

Supervised Training

Sometimes we can get “gold standard” alignments a –priori

- Manual alignments
- Pre-trained with strong alignment model

Train the model to match these strong alignments

Attention is all you need

Motivation

Sequential nature of RNNs -→ difficult to take advantage of modern computing devices such as TPUs (Tensor Processing Units)

Transformer

I arrived at the bank after crossing the river

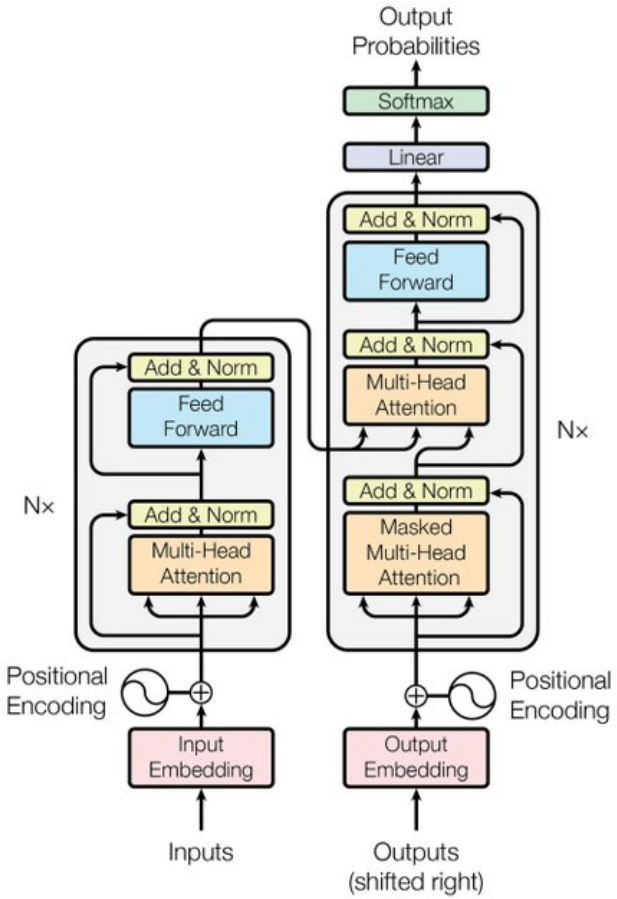
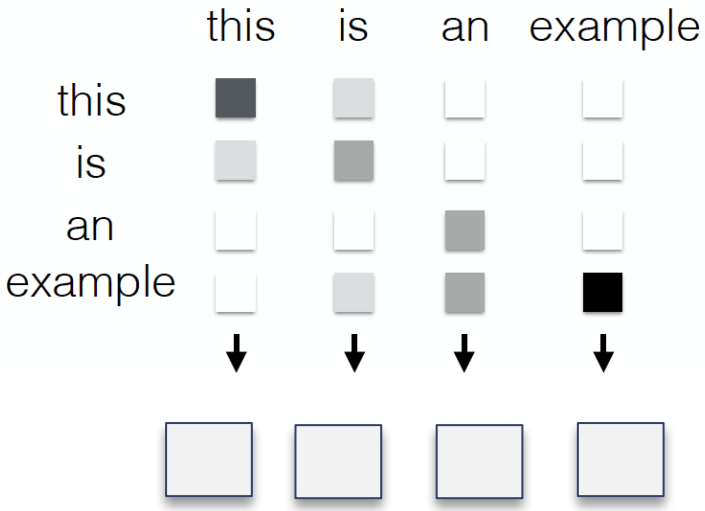


Figure 1: The Transformer - model architecture.

https://arxiv.org/abs/1706.03762

Intra-Attention / Self- Attention

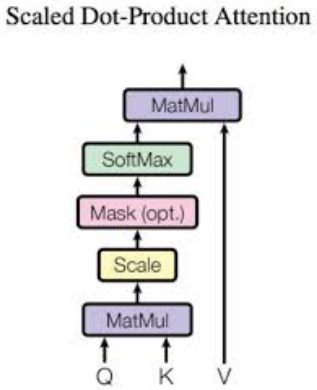
Each element in the sentence attends to other elements from the SAME sentence → context sensitive encodings!



Multi-headed Attention I

Multiple attention “heads” focus on different parts of the sentence

$$a(q,k) = \frac{q^T k}{\sqrt{|k|}}$$

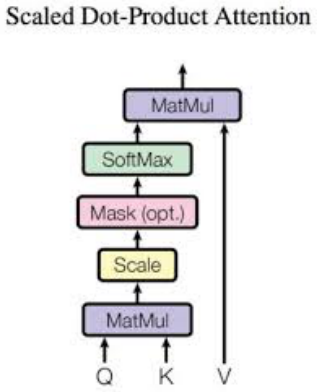


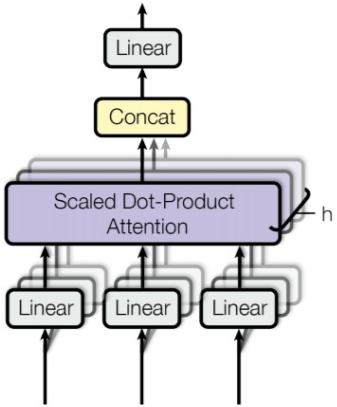
Multi-headed Attention II

Multiple attention “heads” focus on different parts of the sentence

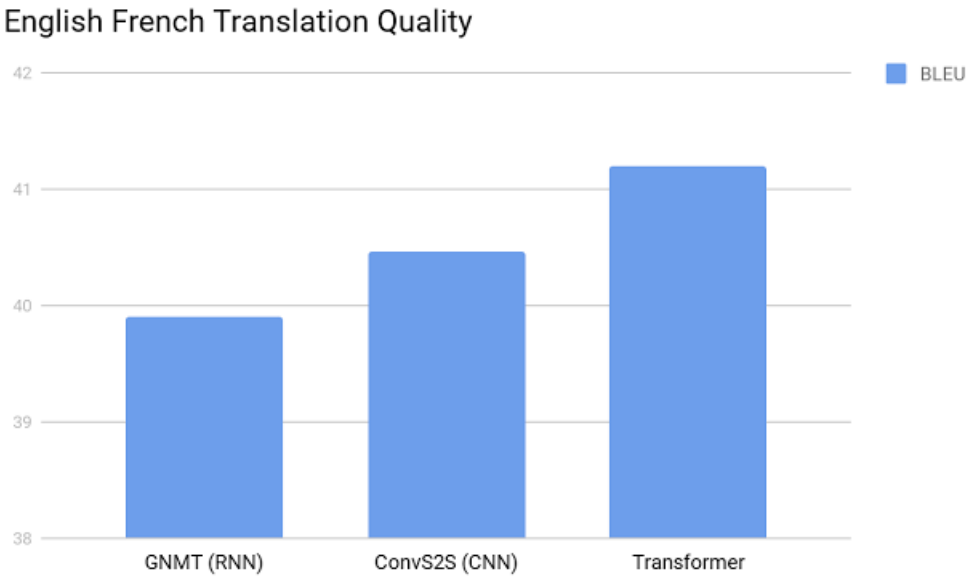
E.g. Multiple independently learned heads (Vaswani et al. 2017)

$$a(q,k) = \frac{q^T k}{\sqrt{|k|}}$$





Transformer results

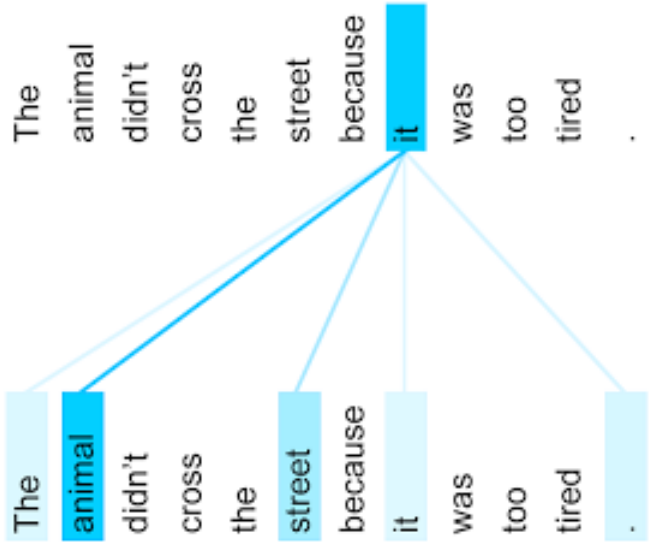


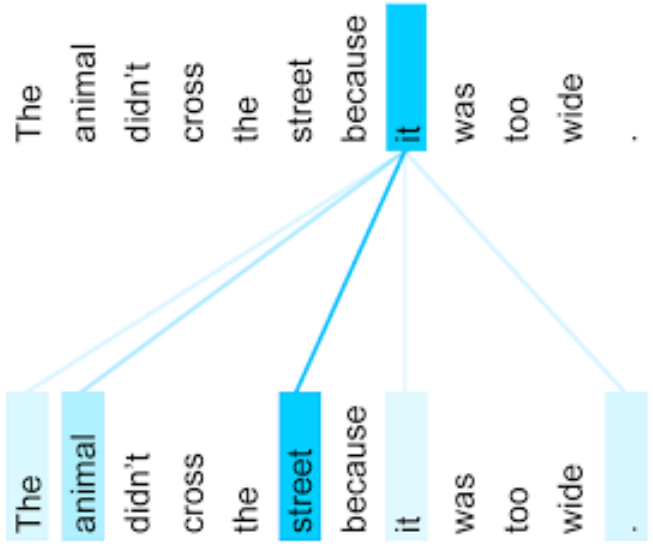
Attention weights

The animal didn't cross the street because it was too tired.
L'animal n'a pas traversé la rue parce qu'il était trop fatigué.

The animal didn't cross the street because it was too wide.
L'animal n'a pas traversé la rue parce qu'elle était trop large.

Attention weights

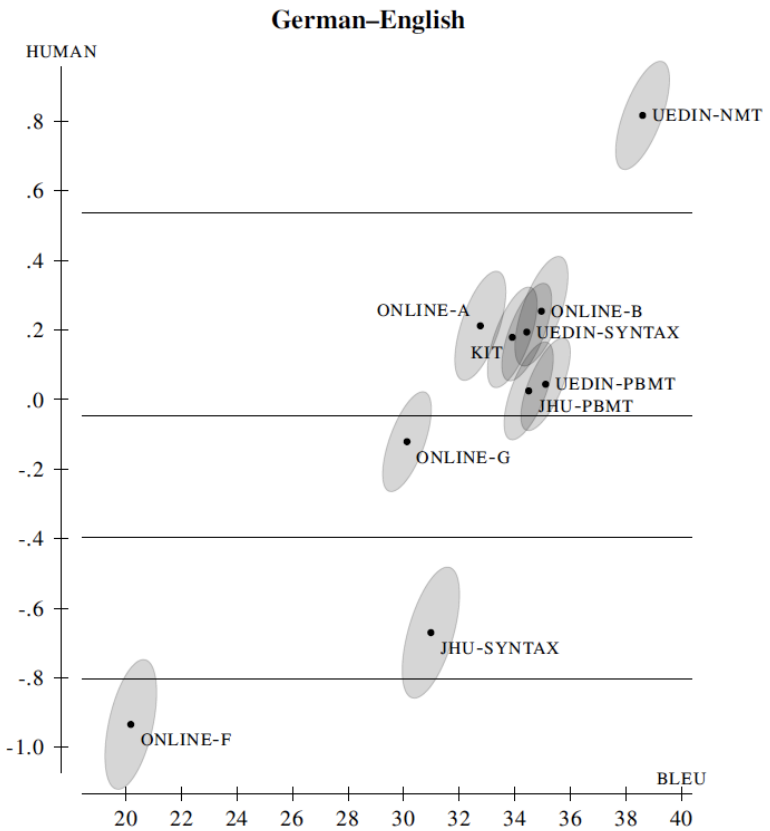


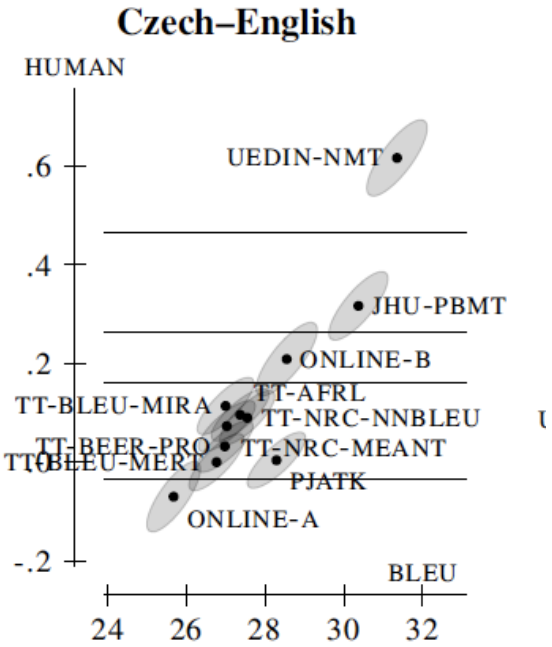


Neural MT is better than phrase-based

Neural Network for Machine Translation at Production Scale

Results in WMT 2016 international evaluation



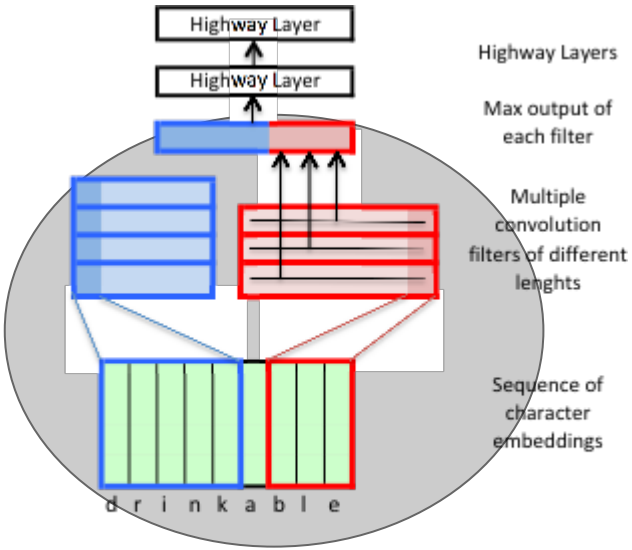


What Next?

Character-based Neural Machine Translation: Motivation

- Word embeddings have been shown to boost the performance in many NLP tasks, including machine translation.
- However, the standard look-up based embeddings are limited to a finite-size vocabulary for both computational and sparsity reasons.
- The orthographic representation of the words is completely ignored.
- The standard learning process is blind to the presence of stems, prefixes, suffixes and any other kind of affixes in words.

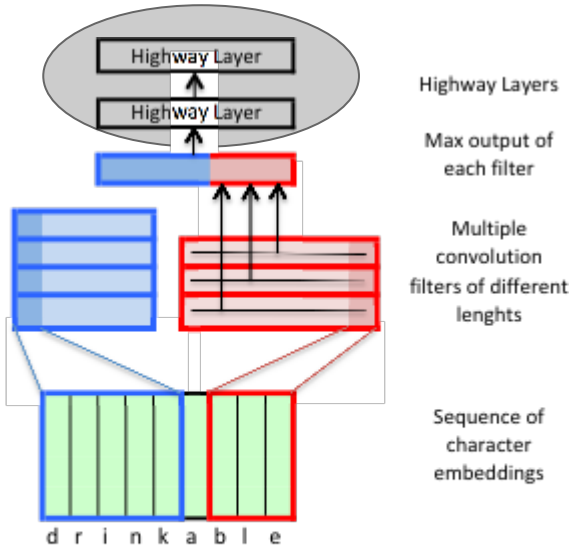
Character-based Neural MT: Proposal (Step 1)



Kim et al, 2015

- The computation of the representation of each word starts with a character-based embedding layer that associates each word (sequence of characters) with a sequence of vectors.
- This sequence of vectors is then processed with a set of 1D convolution filters of different lengths followed with a max pooling layer.
- For each convolutional filter, we keep only the output with the maximum value. The concatenation of these max values already provides us with a representation of each word as a vector with a fixed length equal to the total number of convolutional kernels.

Character-based Neural MT: Proposal (Step 2)

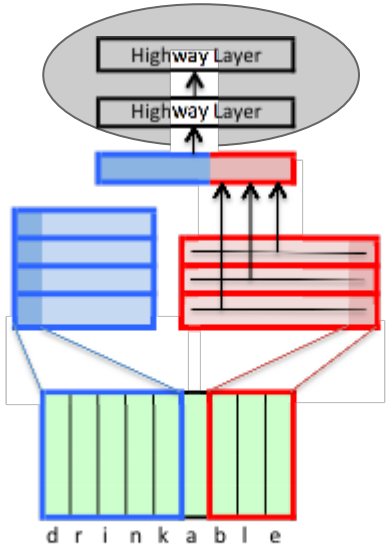


Kim et al, 2015

architecture designed to ease gradient-based training of deep networks

- The addition of two highway layers was shown to improve the quality of the language model in (Kim et al., 2016).
- The output of the second Highway layer will give us the final vector representation of each source word, replacing the standard source word embedding in the neural machine translation system.

Character-based Neural MT: Integration with NMT



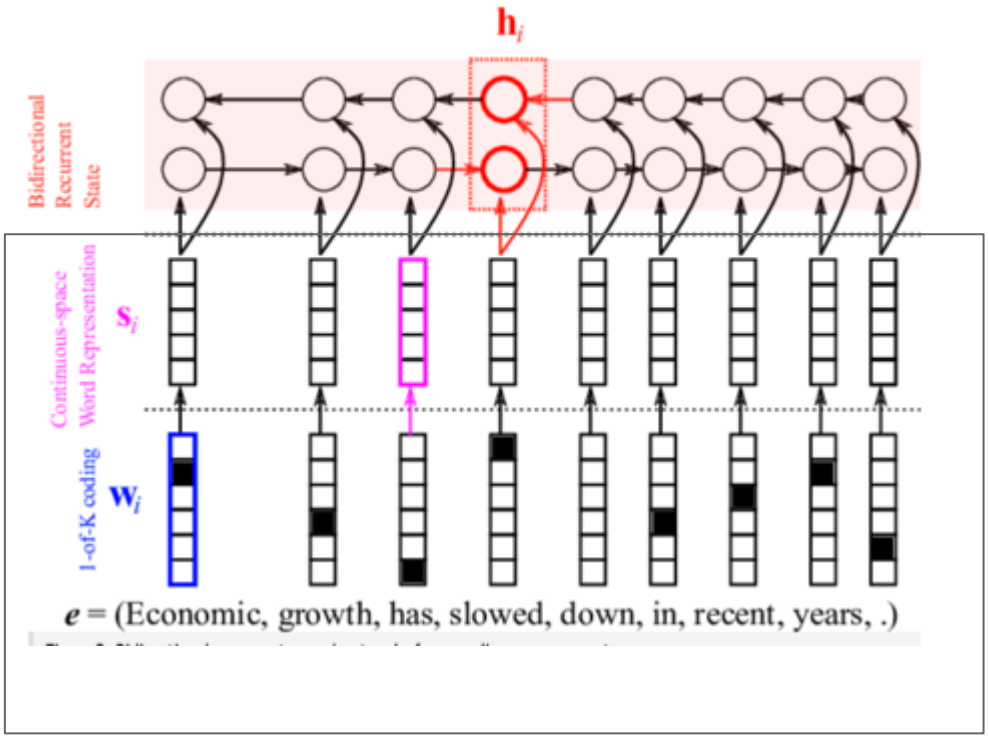
Highway Layers

Max output of each filter

Multiple convolution filters of different lenghts

Sequence of character embeddings

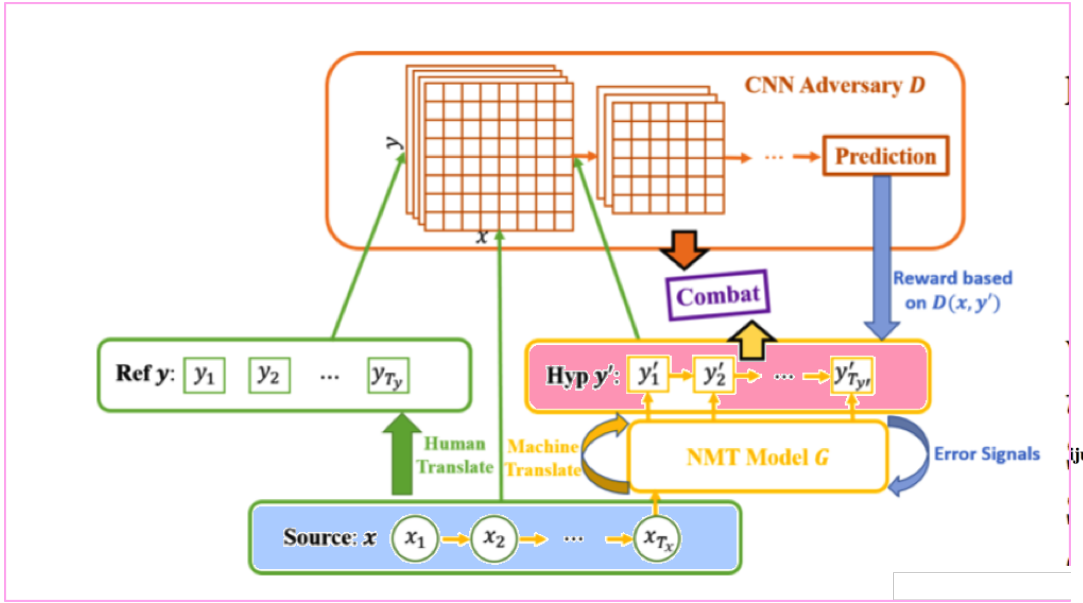
Kim et al, 2015



Examples

1	SRC	Berichten zufolge hofft Indien darber hinaus auf einen Vertrag zur **Verteidigungszusammenarbeit** zwischen den beiden Nationen .
	Phrase	reportedly hopes India , in addition to a contract for the defence cooperation between the two nations .
	NN	according to reports , India also hopes to establish a contract for the UNK between the two nations .
	CHAR	according to reports , India hopes to see a Treaty of **Defence Cooperation** between the two nations .
	REF	India is also reportedly hoping for a deal on **defence collaboration** between the two nations .
4	SRC	der durchtrainierte Mainzer sagt von sich , dass er ein " ambitionierter **Rennradler** " ist .
	Phrase	the will of Mainz says that he a more ambitious .
	NN	the UNK Mainz says that he is a " ambitious , . "
	CHAR	the UNK in Mainz says that he is a ' **ambitious racer** ' .
	REF	the well-conditioned man from Mainz said he was an " **ambitious racing cyclist** . "
3	SRC	die GDL habe jedoch nicht gesagt , wo sie **streiken** wolle , so dass es schwer sei , die Folgen konkret vorherzusehen .
	Phrase	the GDL have , however , not to say , where they strike , so that it is difficult to predict the consequences of concrete .
	NN	however , the UNK did not tell which they wanted to UNK , so it is difficult to predict the consequences .
	CHAR	however , the UNK did not say where they wanted to **strike** , so it is difficult to predict the consequences .
	REF	the GDL have not said , however , where they will **strike** , making it difficult to predict exactly what the consequences will be .
4	SRC	die Premierminister Indiens und Japans trafen sich in Tokio .
	Phrase	the Prime Minister of India and Japan in Tokyo .
	NN	the Prime Minister of India and Japan met in Tokyo
	CHAR	the Prime **Ministers** of India and Japan met in Tokyo
	REF	India and Japan prime **ministers** meet in Tokyo
5	SRC	wo die Beamten es aus den Augen verloren .
	Phrase	where the officials lost sight of
	NN	where the officials lost it out of the eyes
	CHAR	where officials **lose sight of it**
	REF	causing the officers to **lose sight of it**

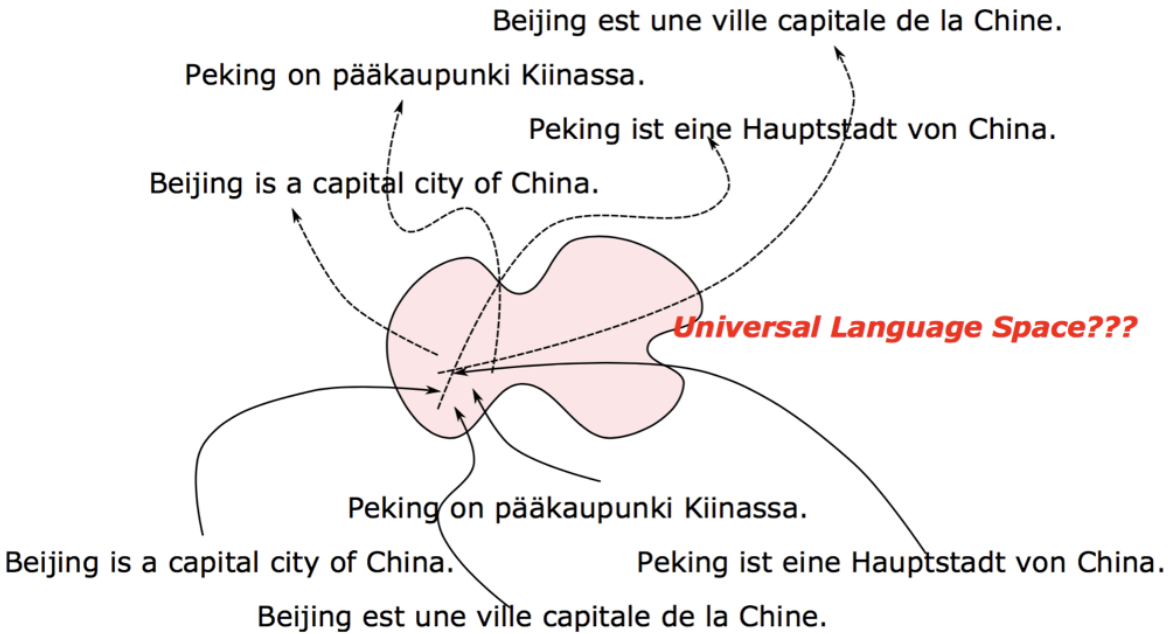
Generative Adversarial Networks



Adversarial Neural Machine Translation

ijun Wu[1], Yingce Xia[2], Li Zhao[3], Fei Tian[3], Tao Qin[3], Jianhuang Lai[1,4] and Tie-Yan Liu[3]
[1]School of Data and Computer Science, Sun Yat-sen University
[2]University of Science and Technology of China
[3]Microsoft Research Asia
[4]Guangdong Key Laboratory of Information Security Technology
wulijun3@mail2.sysu.edu.cn; yingce.xia@gmail.com;
{lizo,fetia,taoqin,tie-yan.liu}@microsoft.com; stsljh@mail.sysu.edu.cn

Multilingual Translation



Kyunghyun Cho, "DL4MT slides" (2015)

Multilingual Translation Approaches

Sharing attention-based mechanism across language pairs

Orhan Firat et al, “Multi-way, Multilingual Neural Machine Translation with a Shared-based Mechanism” (2016)

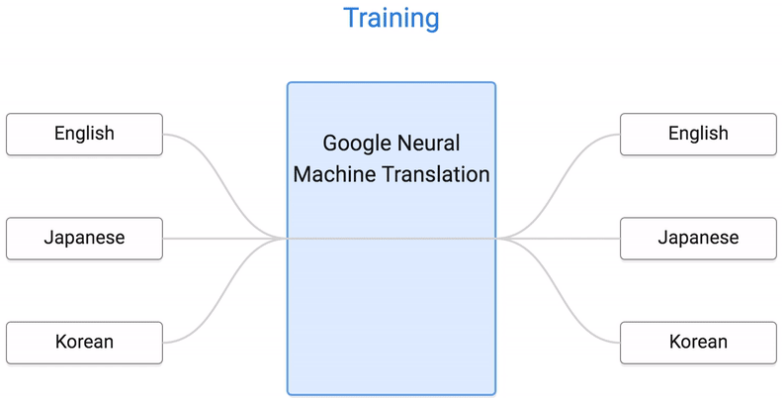
Multilingual Translation Approaches

Sharing attention-based mechanism across language pairs

Orhan Firat et al, "Multi-way, Multilingual Neural Machine Translation with a Shared-based Mechanism" (2016)

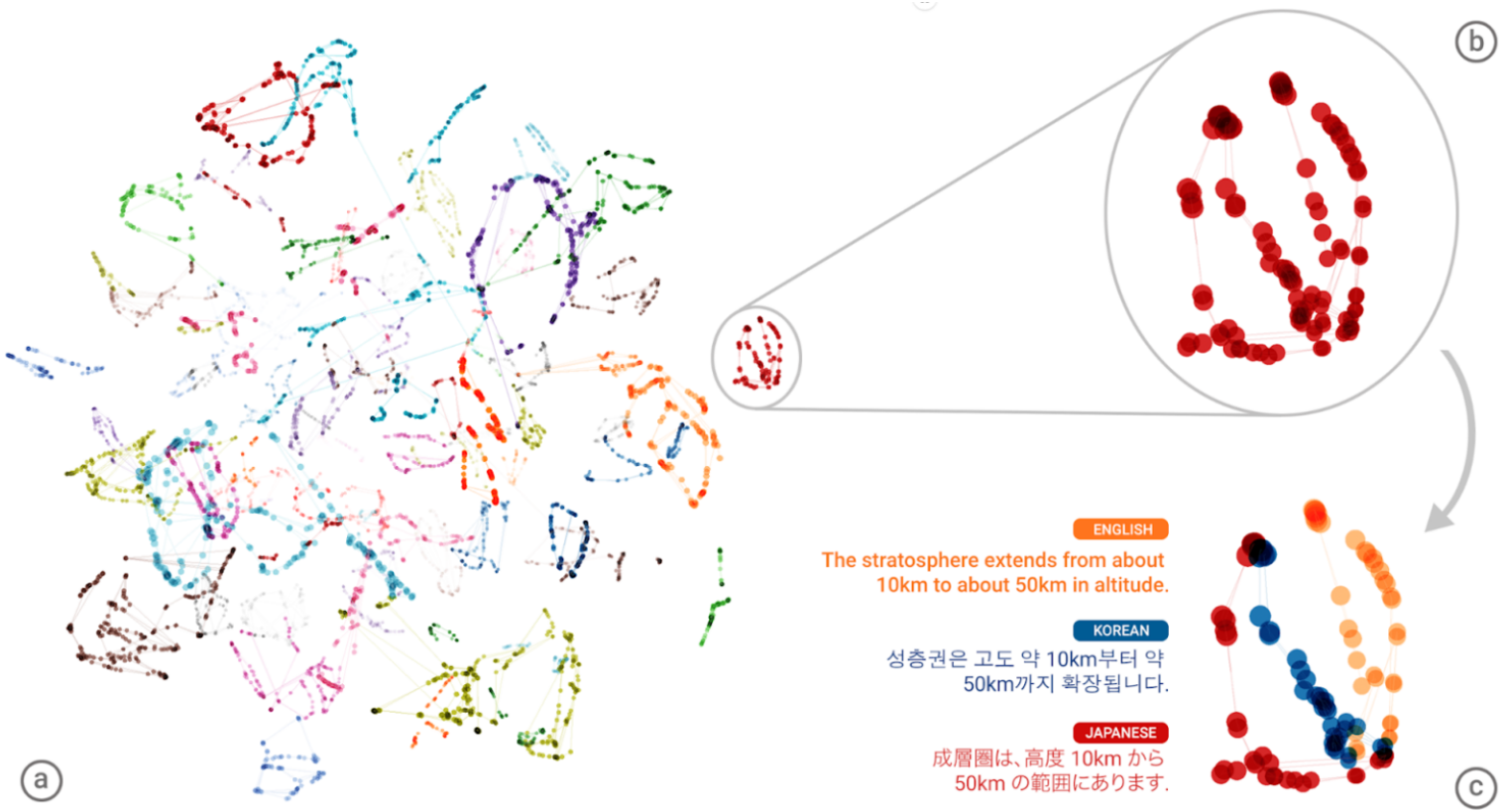
Share encoder, decoder, attention accross language pairs

Johnson et al, "Google's Multilingual Neural Machine Translation System: Enabling Zero-Shot Translation" (2016)



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

Is the system learning an Interlingua?



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

Summary

- Machine Translation is faced as a **sequence-to-sequence** problem
- The source sentence is **encoded** into a fixed length vector and this fixed length vector is **decoded** into the final most probable target sentence
- Only **parallel corpus** and **automatic evaluation** measures are required to train a neural machine translation system

Summary

- Attention-based mechanism allows to achieve state-of-the-art results
- Progress in MT includes character-based, multilinguality...

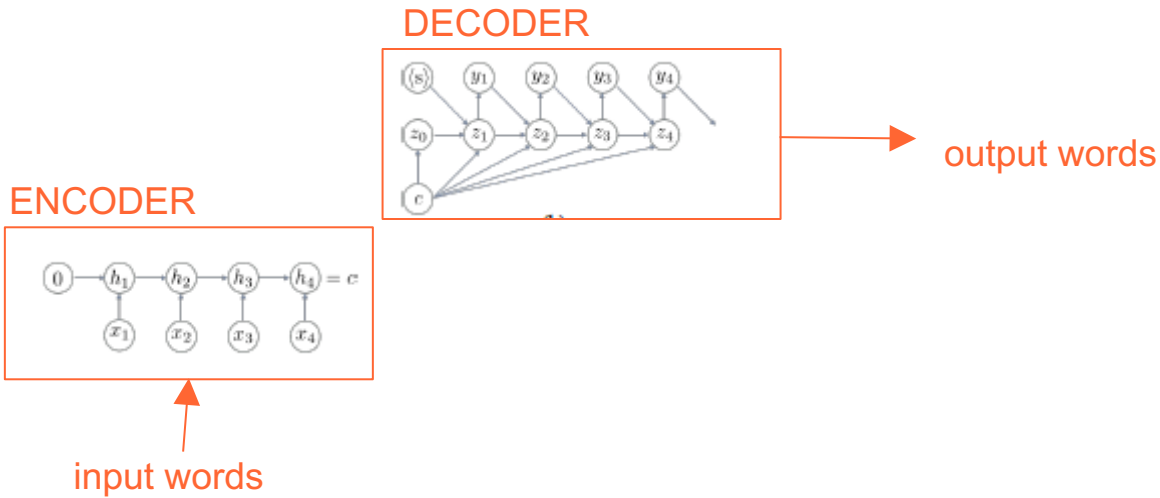
Learn more

Natural Language Understanding with Distributed Representation, Kyunghyun Cho, Chapter 6, 2015 (available in github)

Thanks ! Q&A ?

https://www.costa-jussa.com
marta.ruiz@upc.edu

Another useful image for encoding-decoding



Kyunghyun Cho, “Natural Language Understanding with Distributed Representations” (2015)