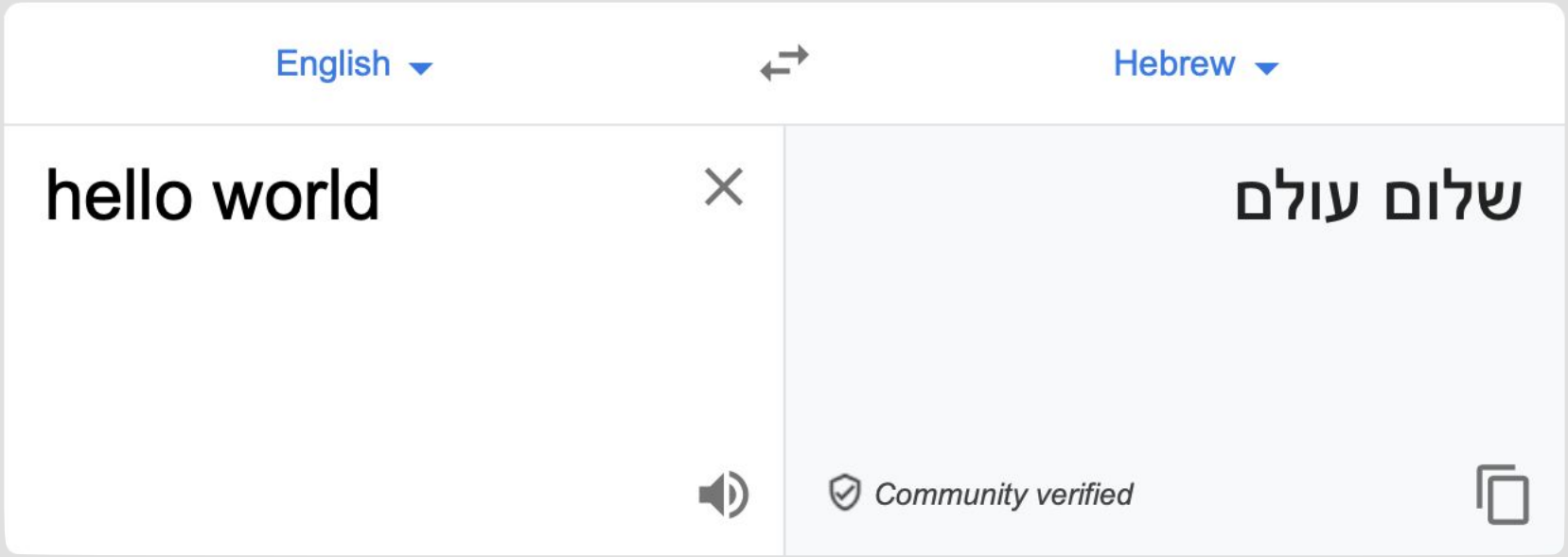
Machine translation

Based on lecture by Ekaterina Artemova

- **Encoder-decoder model**
- Attention mechanism
- Machine translation quality metrics

Encoder-decoder model

Machine translation



Machine translation

- Popular and demanded task

Machine translation

- Popular and demanded task
- One of the first technologies in text preprocessing

- Georgetown–IBM experiment (1954)

Machine translation

- Popular and demanded task
- One of the first technologies in text preprocessing
- Georgetown–IBM experiment (1954)
- Until mid-2010s IBM statistical models prevailed

Machine translation

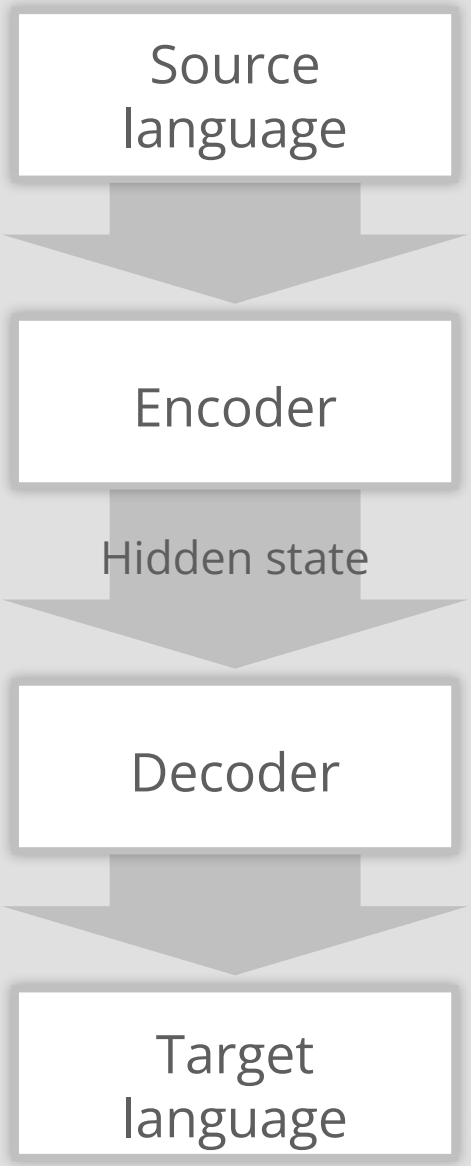
- Popular and demanded task
- One of the first technologies in text preprocessing
- Georgetown–IBM experiment (1954)
- Until mid-2010s IBM statistical models prevailed
- Starting from 2016, neural machine translation (NMT) – industrial standard

Machine translation

- Popular and demanded task
- One of the first technologies in text preprocessing
- Georgetown–IBM experiment (1954)
- Until mid-2010s IBM statistical models prevailed
- Starting from 2016, neural machine translation (NMT) – industrial standard
- As for now, MT works good for simple or formal texts, MT for fiction is still a problem

Encoder-decoder model

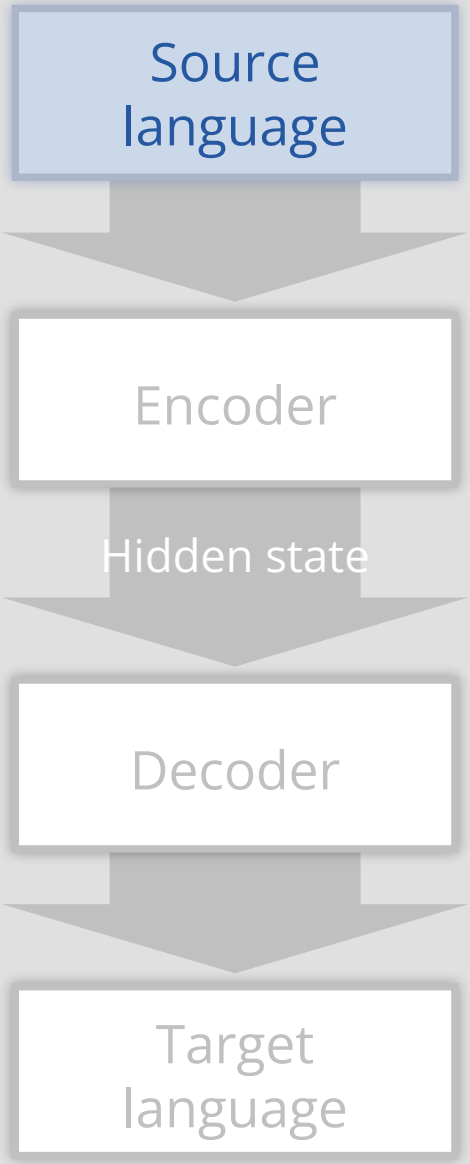
- Problem: translate text from one language to another



Problem formulation

- Given a sentence in source language:

$$\mathbf{x}_{source} = (x_1, \ldots, x_n), x_i \in V_{source}$$



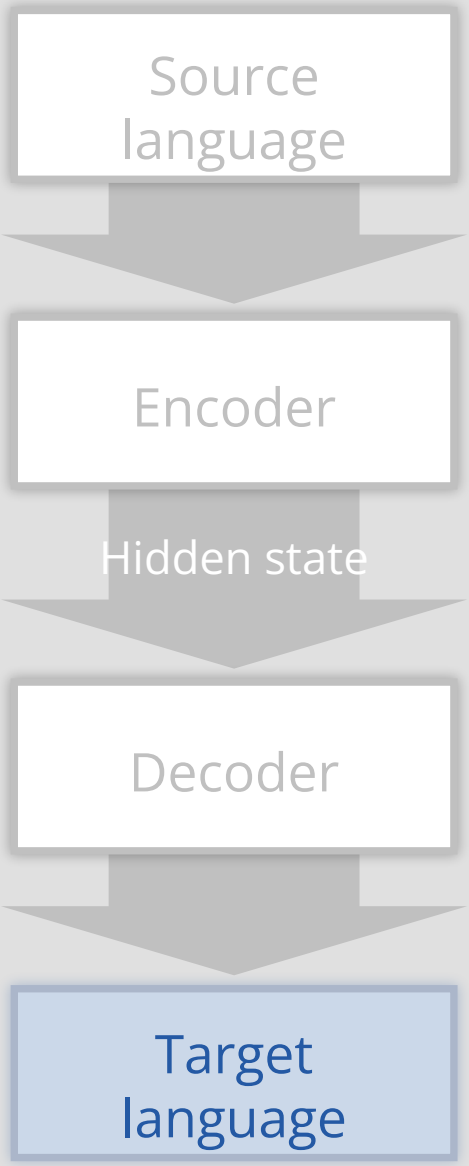
Problem formulation

- Given a sentence in source language:

$$\mathbf{x}_{source} = (x_1, \ldots, x_n), x_i \in V_{source}$$

- and the sentence in the target language:

$$\mathbf{y}_{target} = (y_1, \ldots, y_m), y_i \in V_{target}$$



Problem formulation

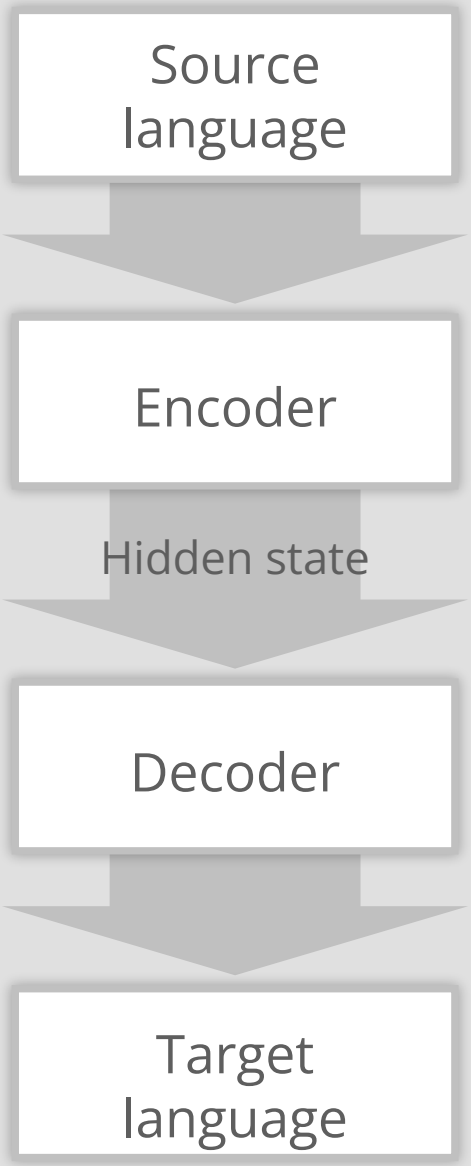
- Given a sentence in source language:

$$\mathbf{x}_{source} = (x_1, \ldots, x_n), x_i \in V_{source}$$

- the sentence in the target language:

$$\mathbf{y}_{target} = (y_1, \ldots, y_m), y_i \in V_{target}$$

- A parallel corpus **C** of pairs (x, y)



Problem formulation

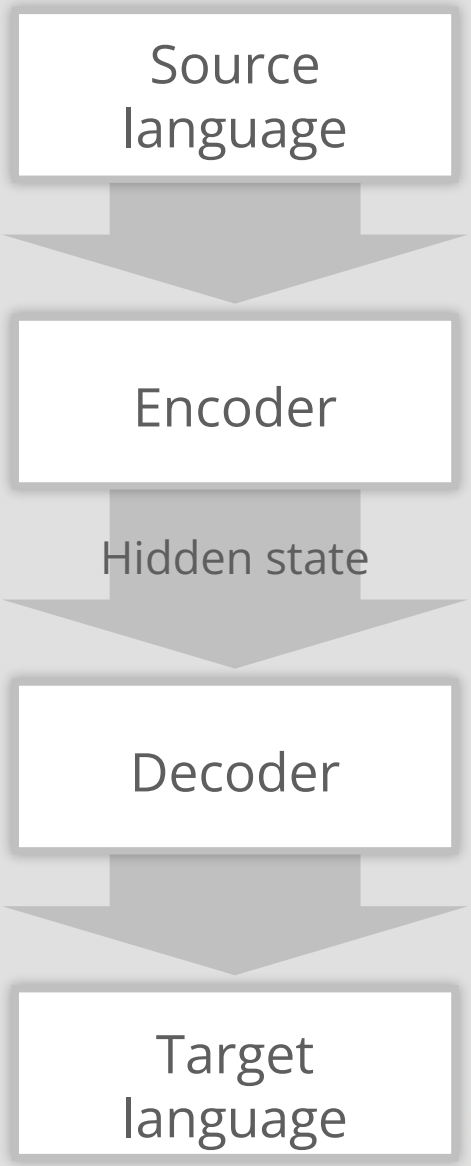
- Given a sentence in source language:

$$\mathbf{x}_{source} = (x_1, \ldots, x_n), x_i \in V_{source}$$

- the sentence in the target language:

$$\mathbf{y}_{target} = (y_1, \ldots, y_m), y_i \in V_{target}$$

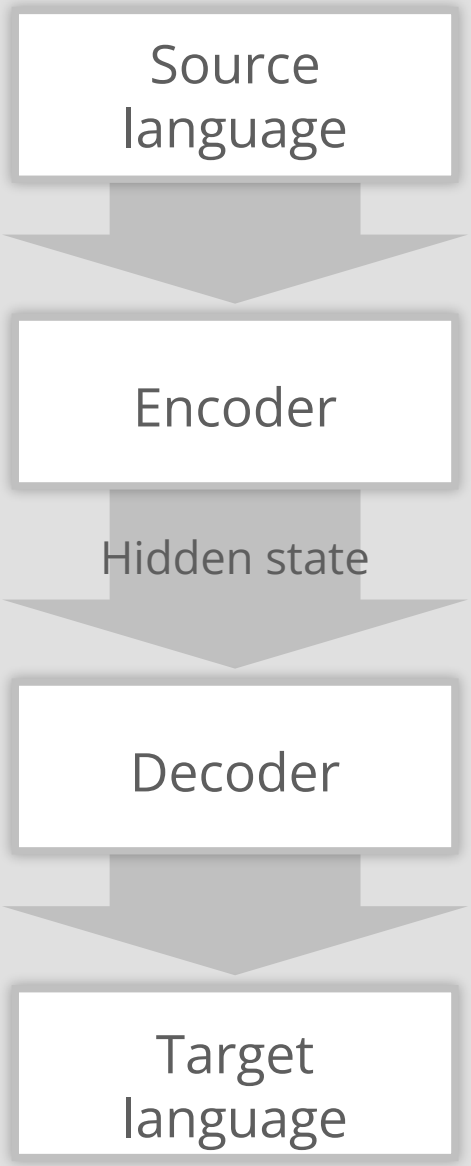
- A parallel corpus **C** of pairs (x, y)
- Maximize the likelihood p(y|x)
- Loss function $\mathcal{L}_\theta = \sum_{\mathbf{x},\mathbf{y} \in \mathbf{C}} \log p(\mathbf{y}|\mathbf{x};\theta)$



Problem formulation

- Probability of target sentence provided source sentence:

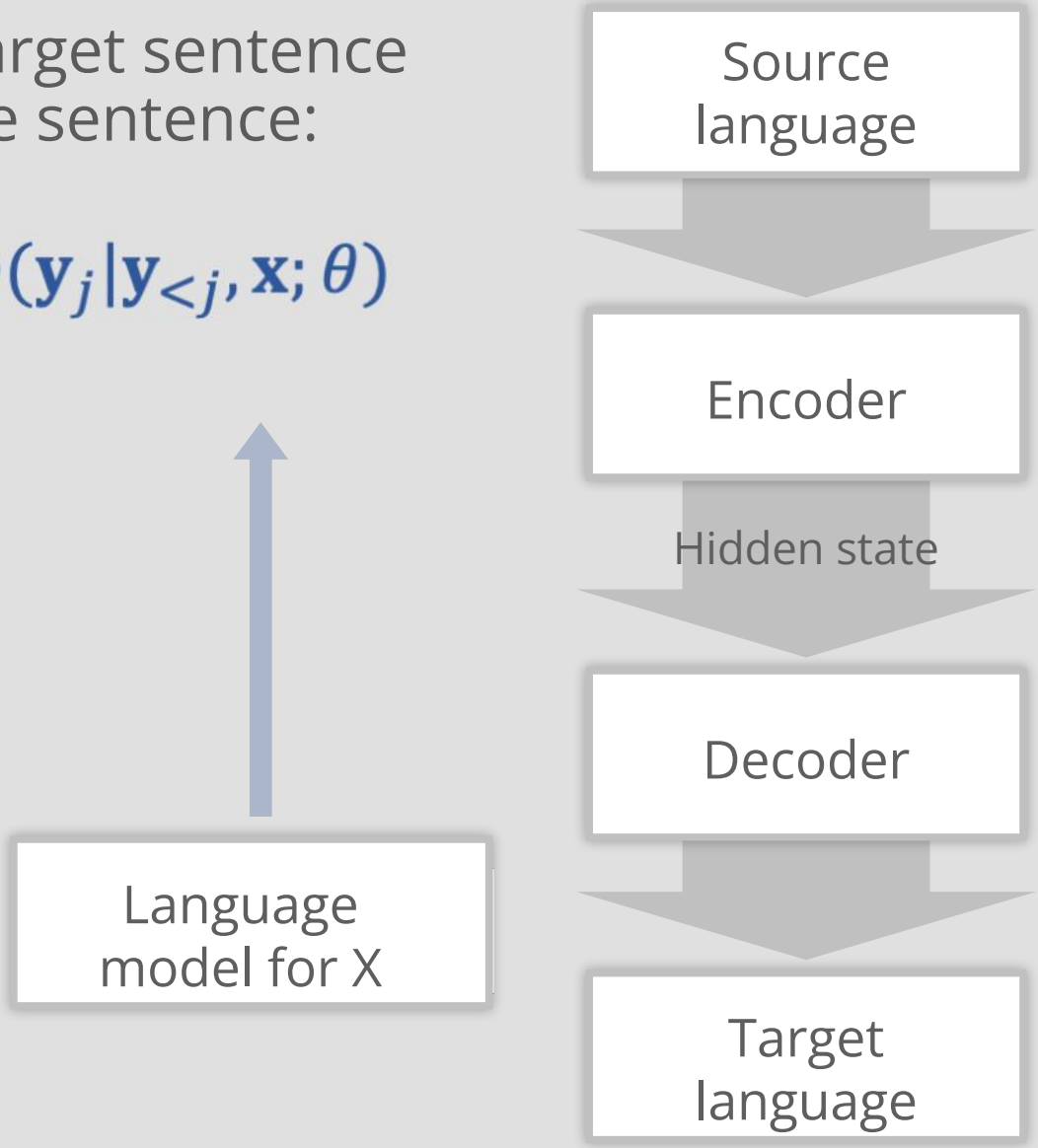
$$p(\mathbf{y}|\mathbf{x};\theta) = \prod_{j=1}^{m} p(\mathbf{y}_j|\mathbf{y}_{<j},\mathbf{x};\theta)$$



Problem formulation

- Probability of target sentence provided source sentence:

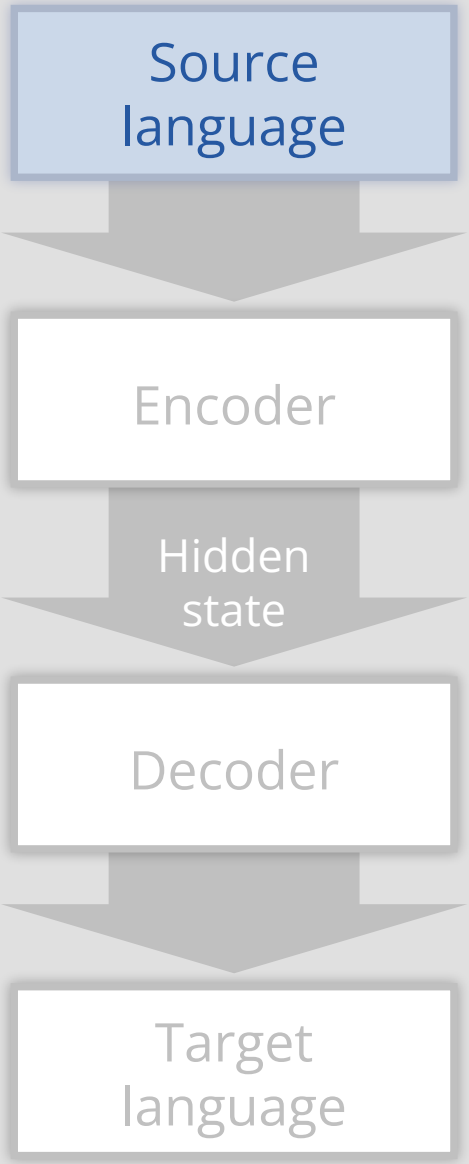
$$p(\mathbf{y}|\mathbf{x};\theta) = \prod_{j=1}^{m} p(\mathbf{y}_j|\mathbf{y}_{<j},\mathbf{x};\theta)$$



Neural machine translation

- Input embeddings:

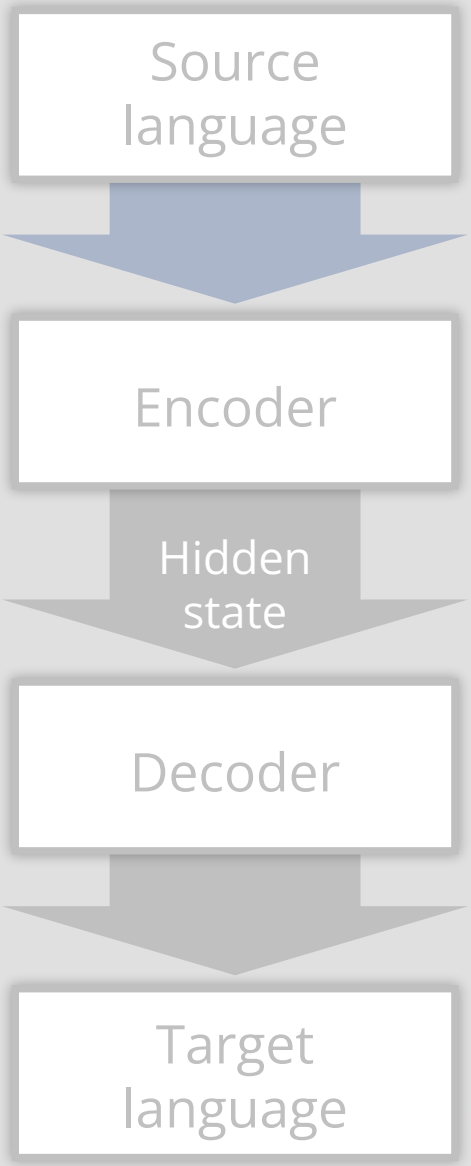
$$f_i^s = E_{source} x_i$$



Neural machine translation

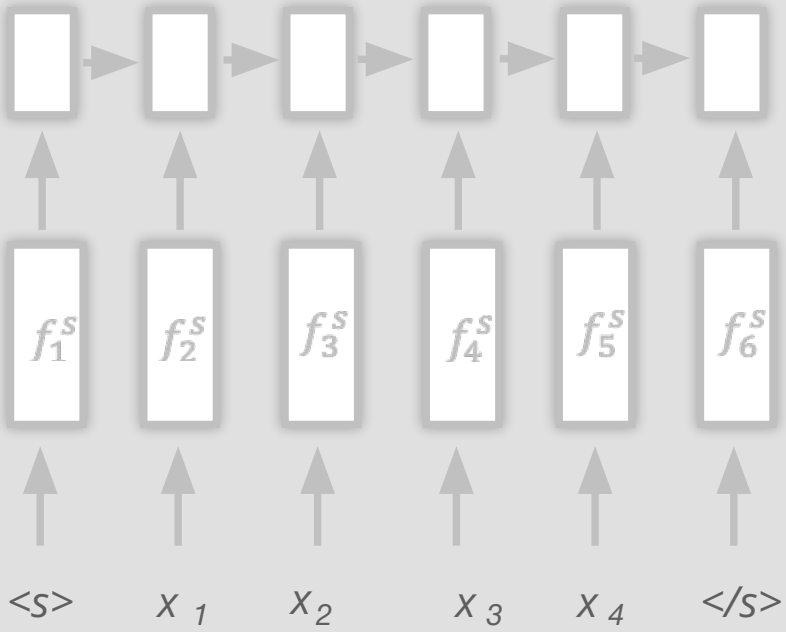
- Input embeddings:

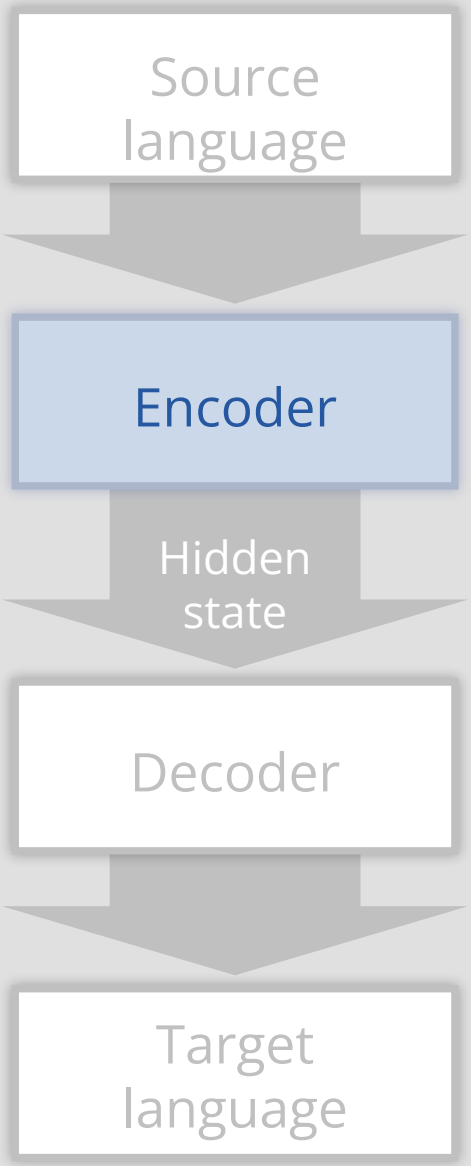
$$f_i^s = E_{source} x_i$$



Neural machine translation

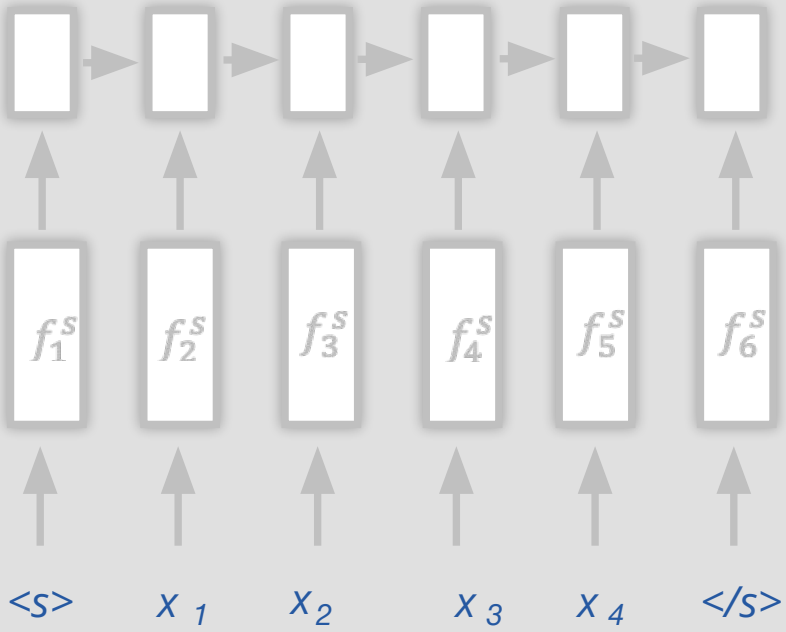
- Encoder architecture

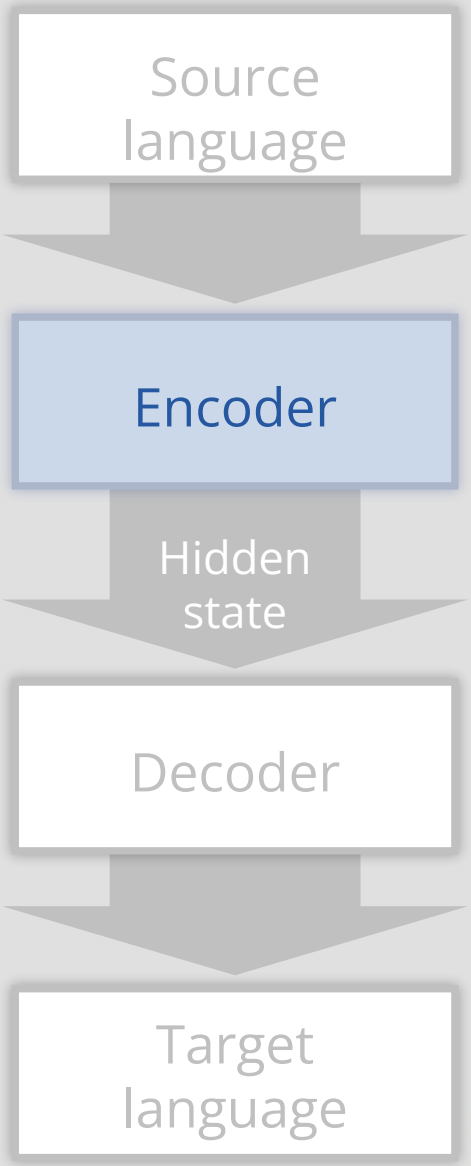




Neural machine translation

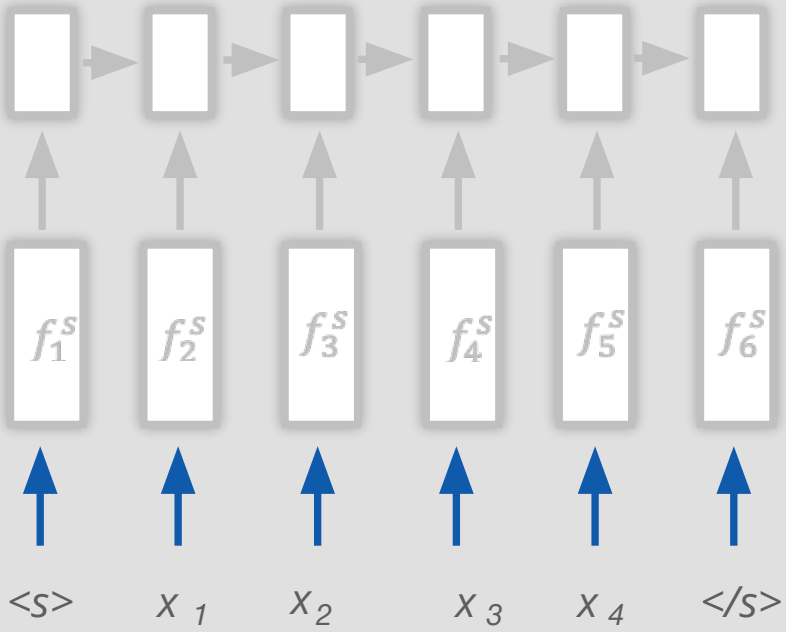
- Encoder architecture

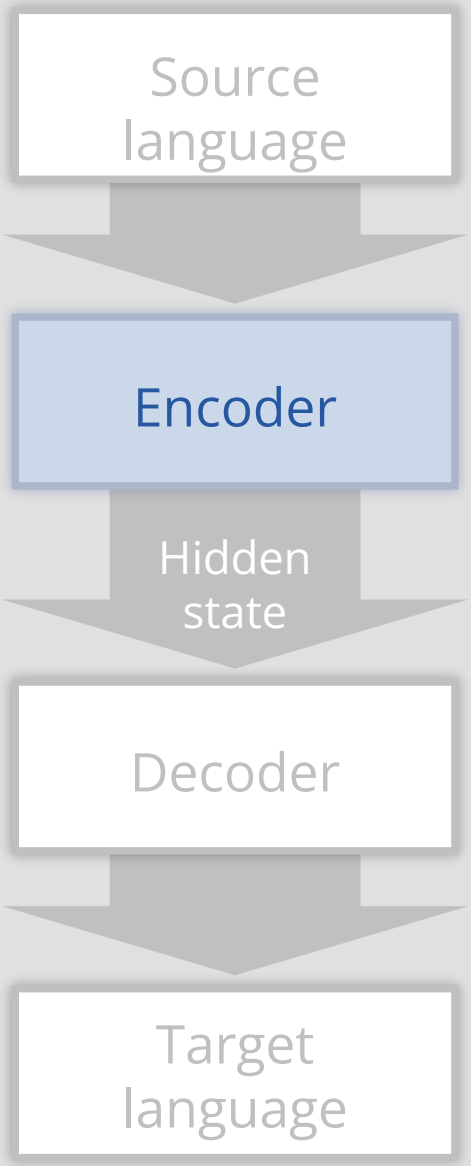




Neural machine translation

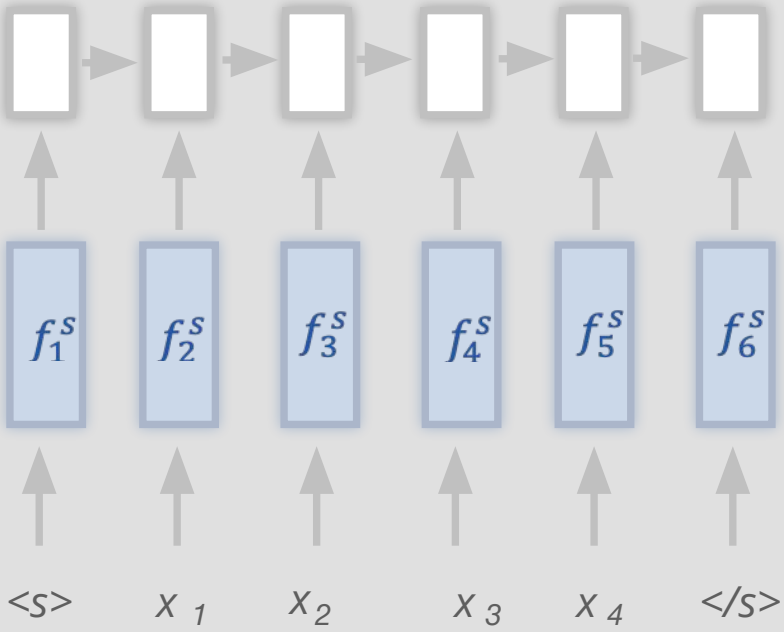
- Encoder architecture

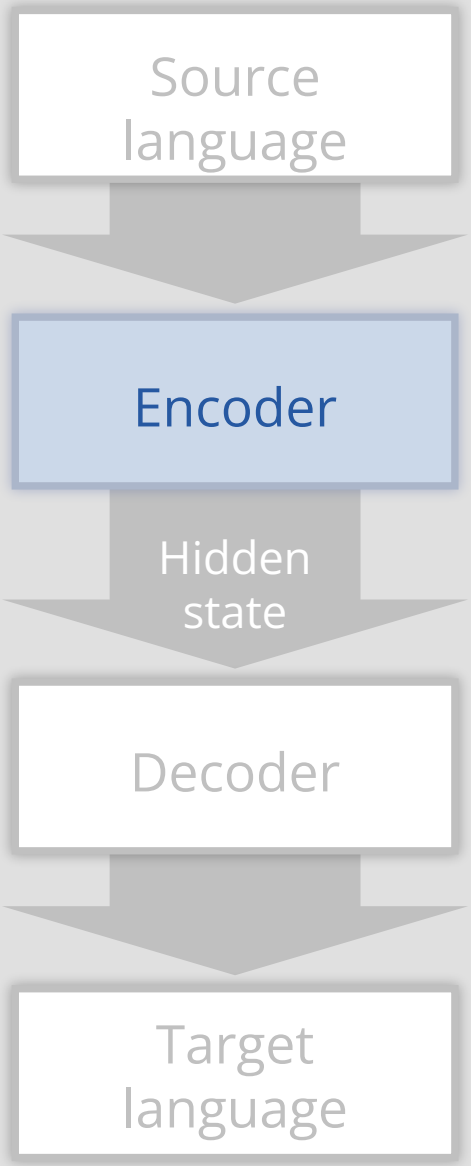




Neural machine translation

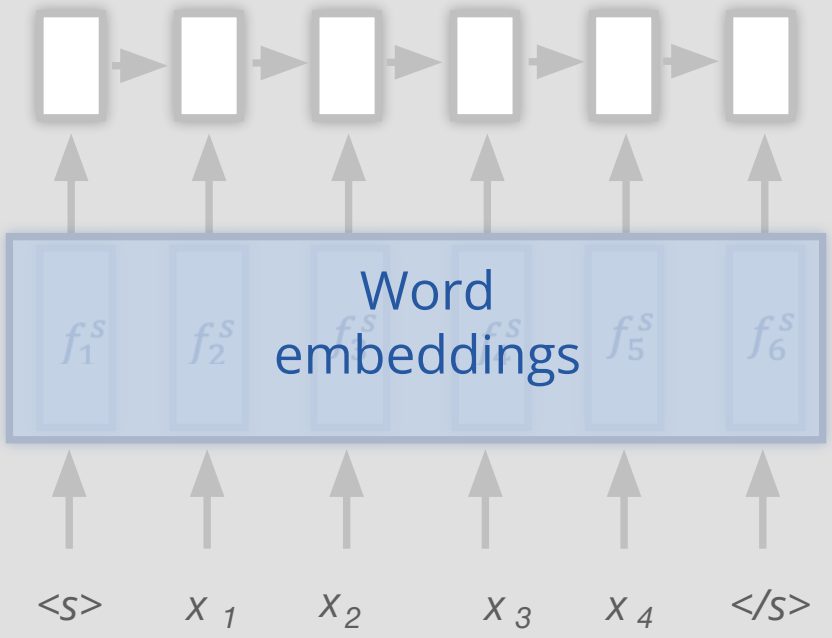
- Encoder architecture

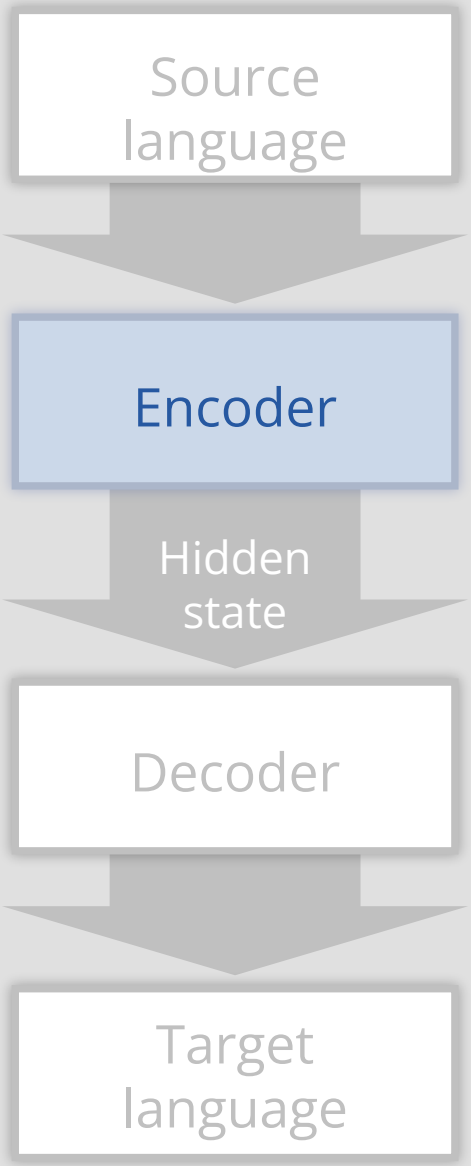




Neural machine translation

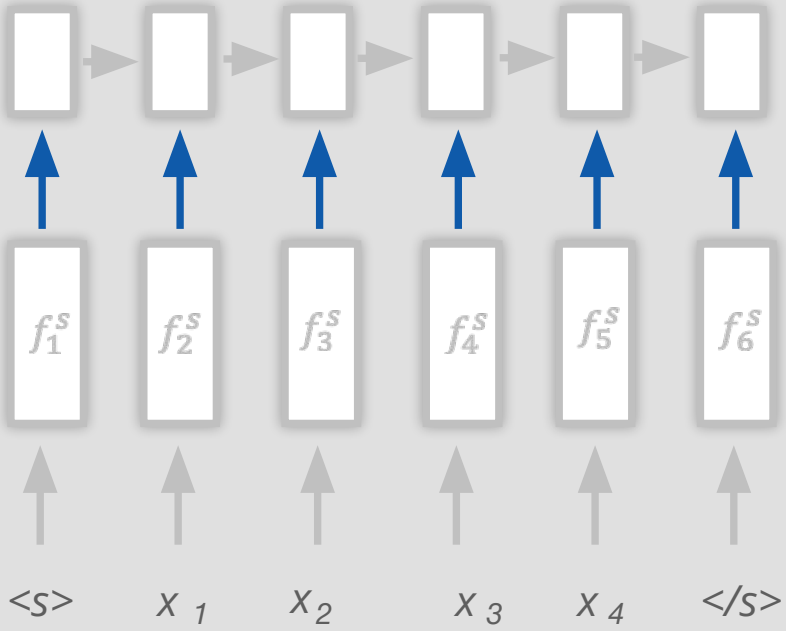
- Encoder architecture

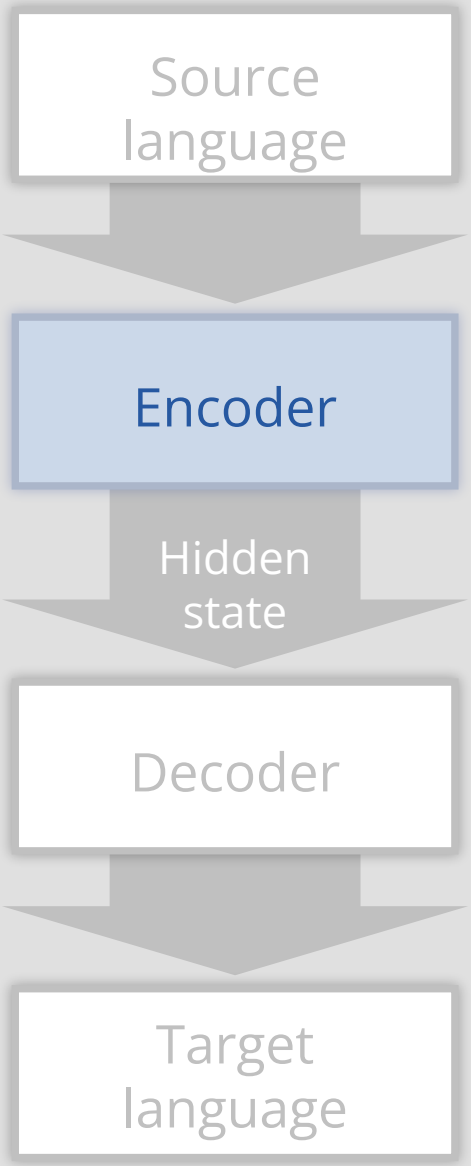




Neural machine translation

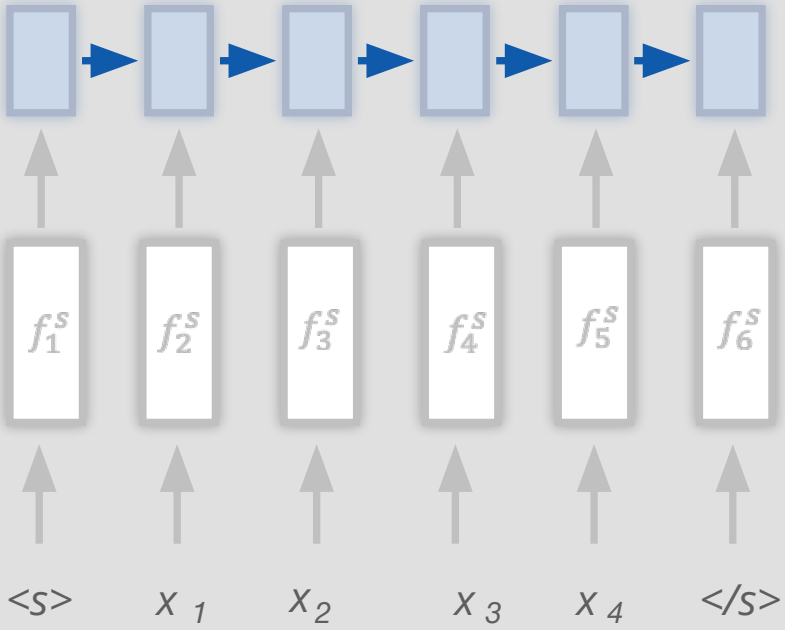
- Encoder architecture

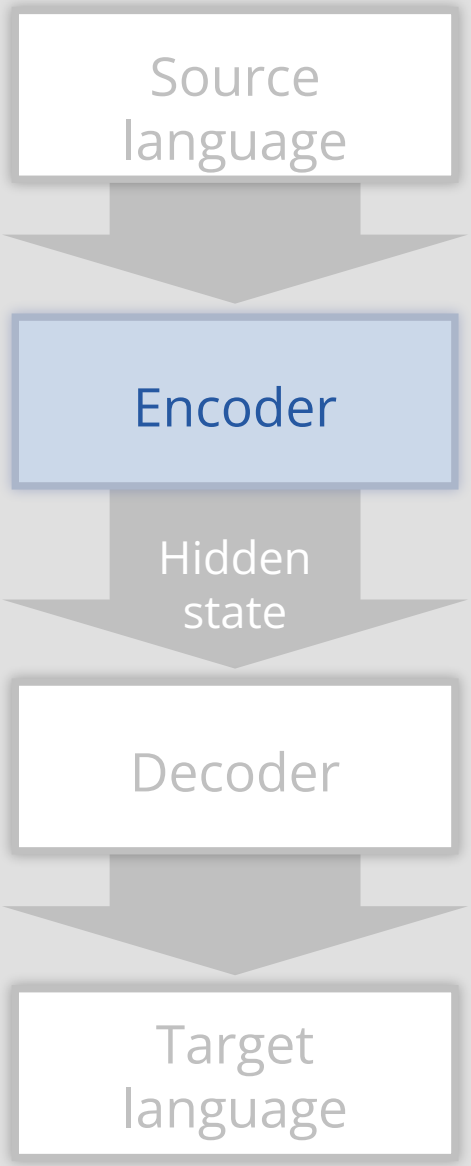




Neural machine translation

- Encoder architecture

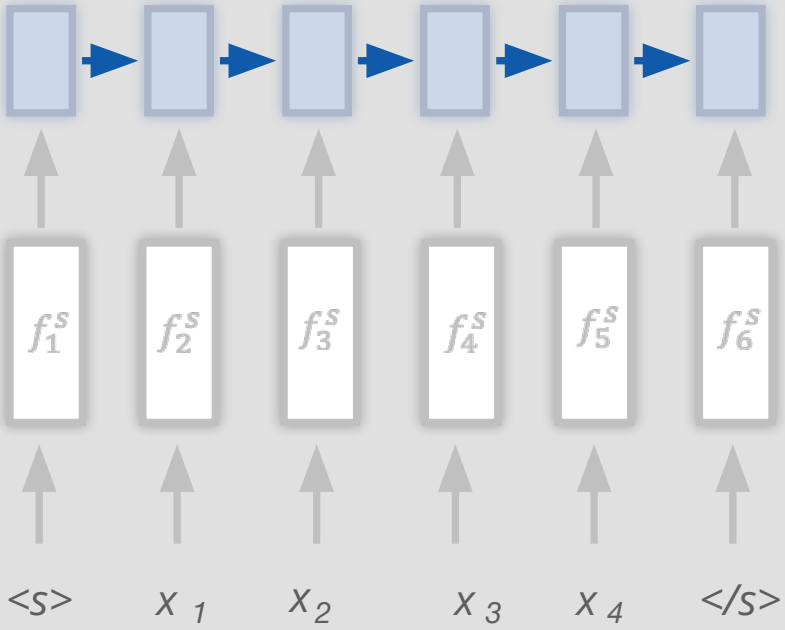


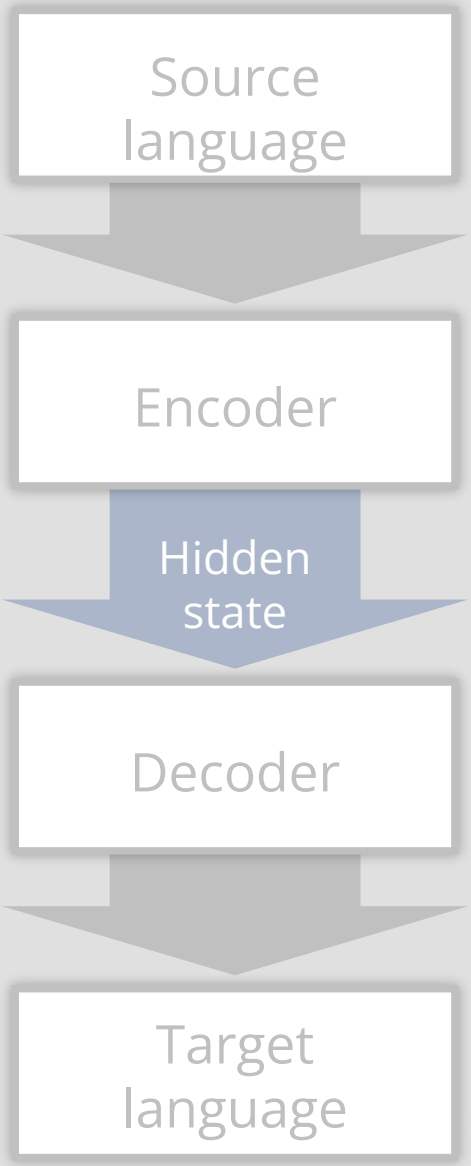


Neural machine translation

- Encode the input sentence

$$h^{encoder} = RNN(\mathbf{x})$$

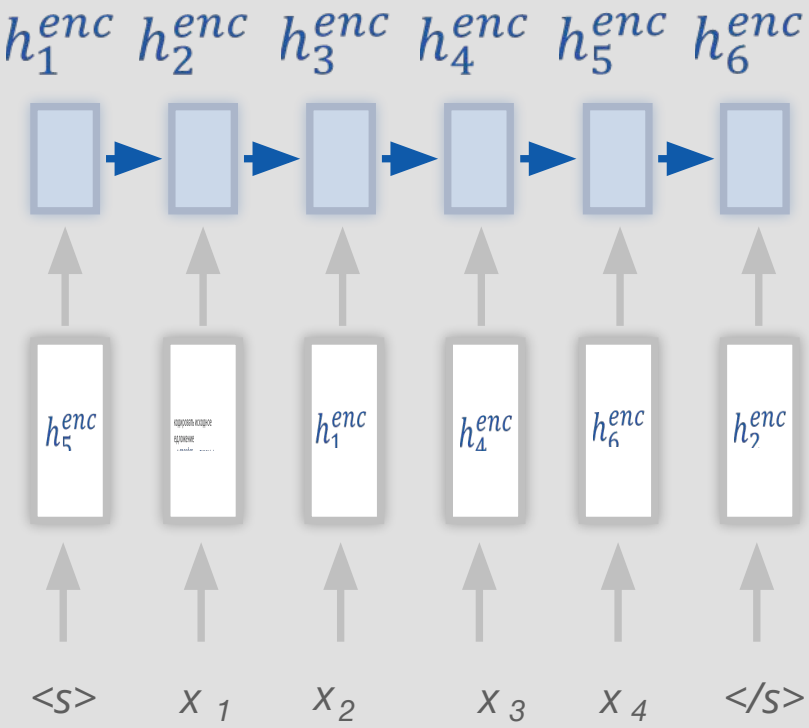


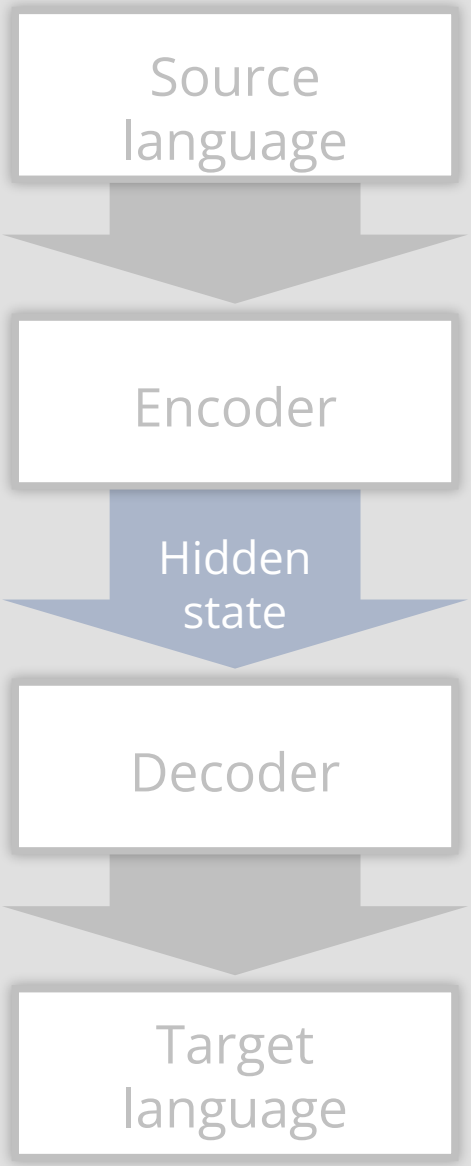


Neural machine translation

- Encode the input sentence

$$h^{encoder} = RNN(\mathbf{x})$$

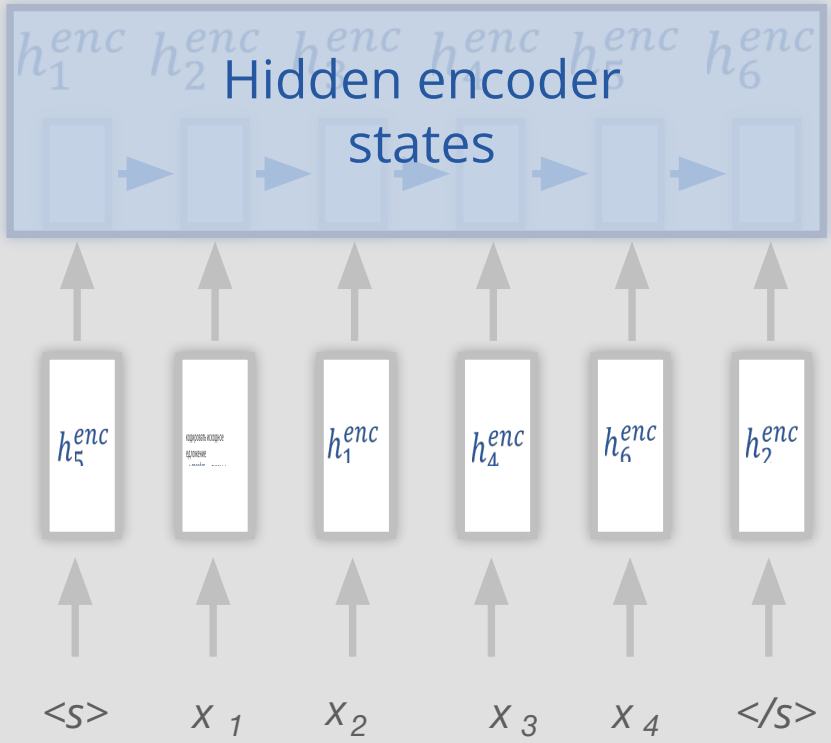


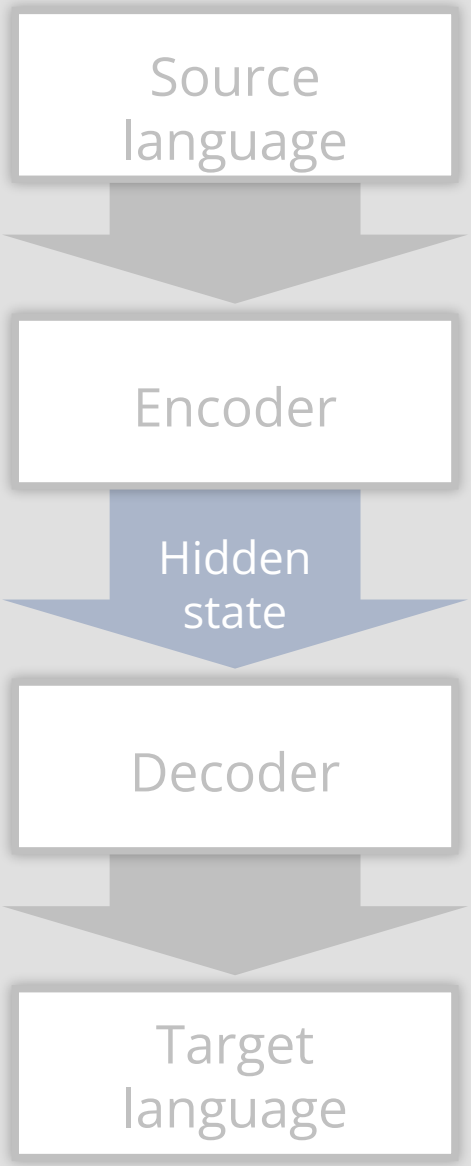


Neural machine translation

- Encode the input sentence

$$h^{encoder} = RNN(\mathbf{x})$$

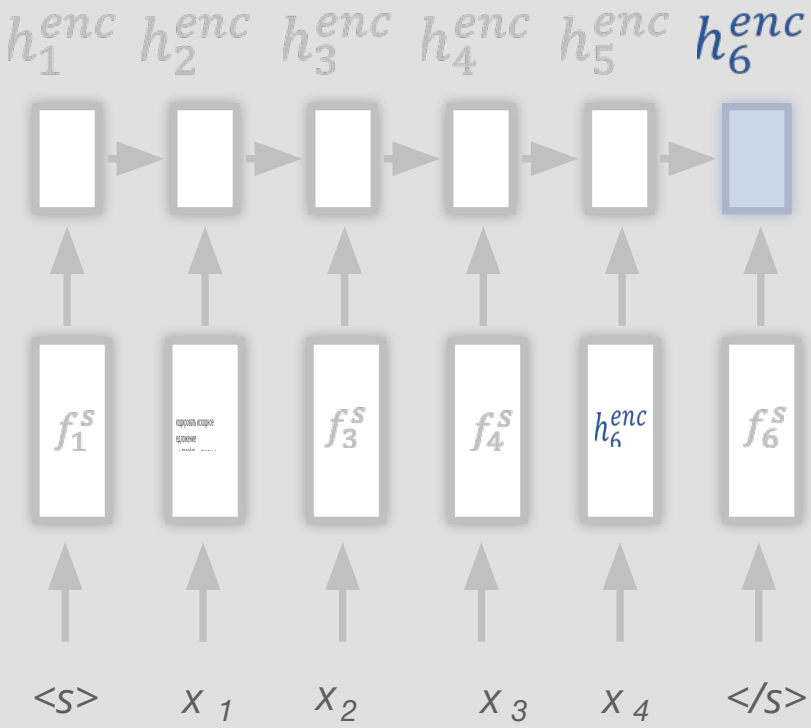


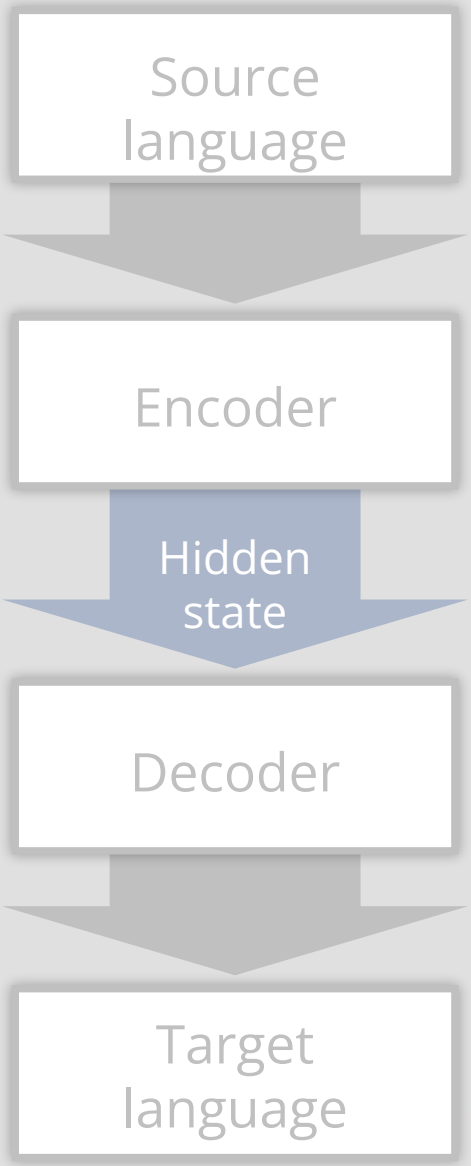


Neural machine translation

- Encode the input sentence

$$h^{encoder} = RNN(\mathbf{x})$$

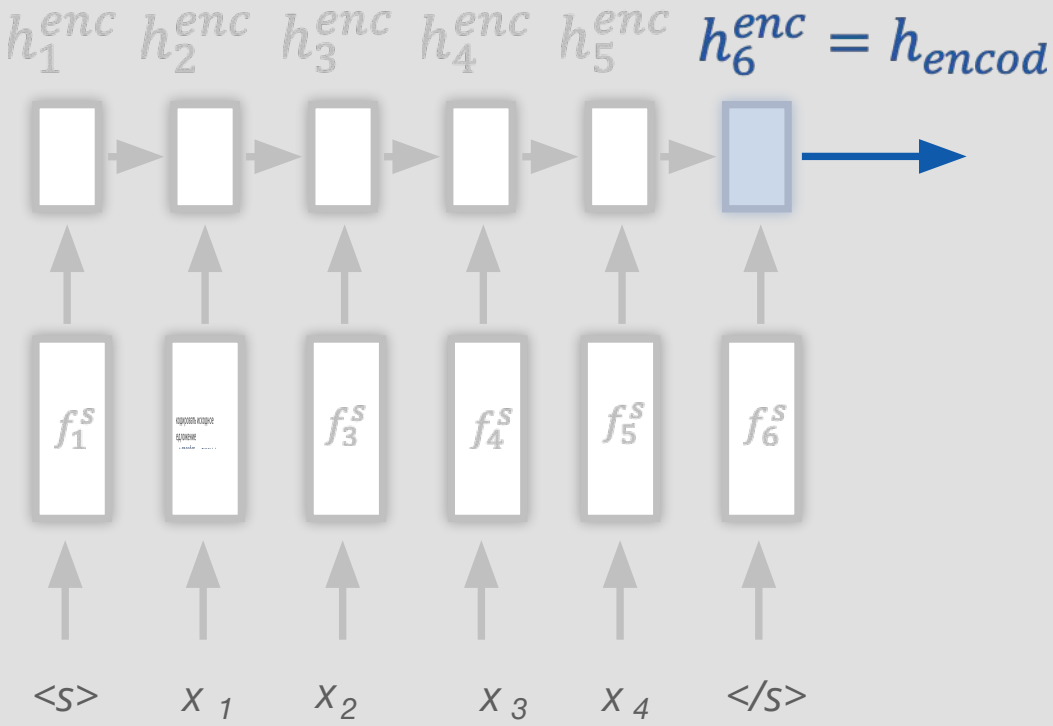


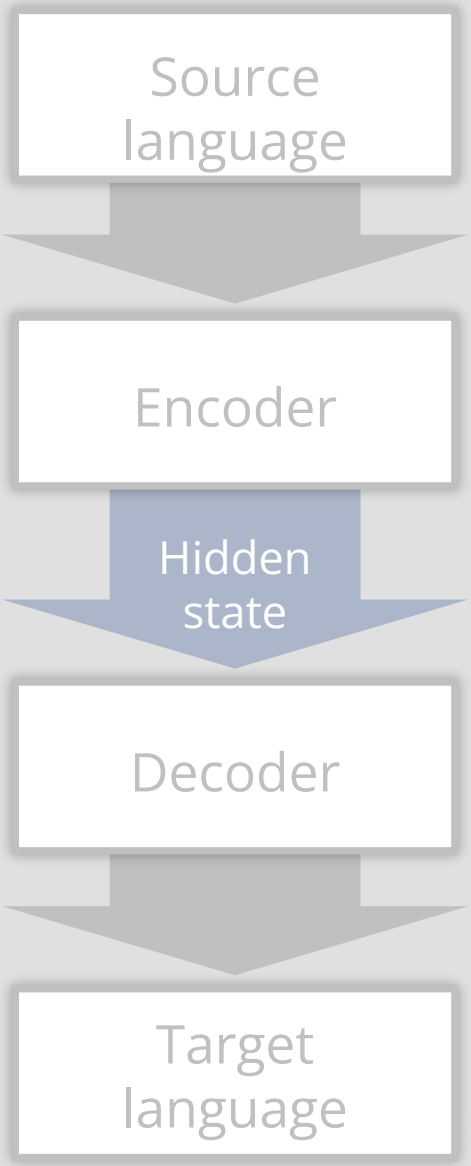


Neural machine translation

- Encode the input sentence

$$h^{encoder} = RNN(\mathbf{x})$$

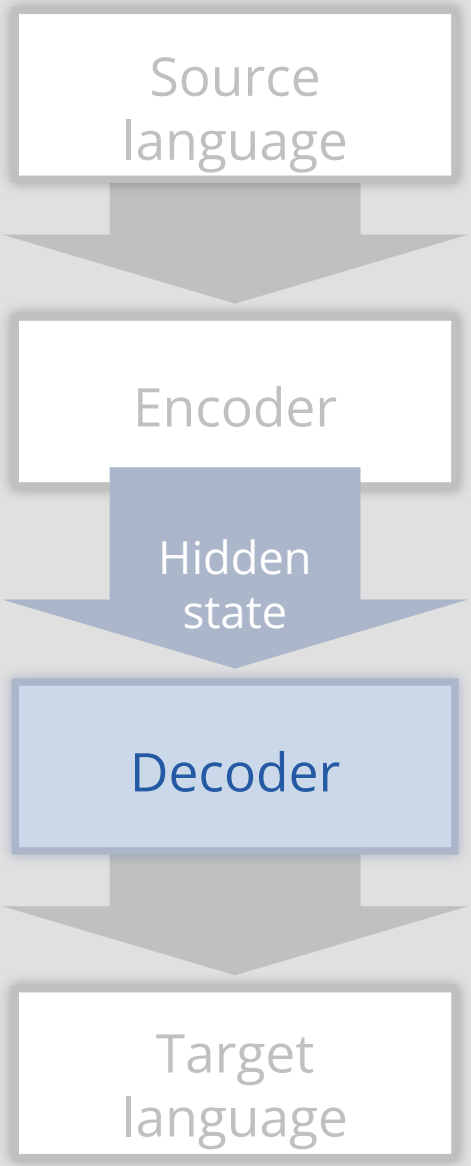




Decoder architecture

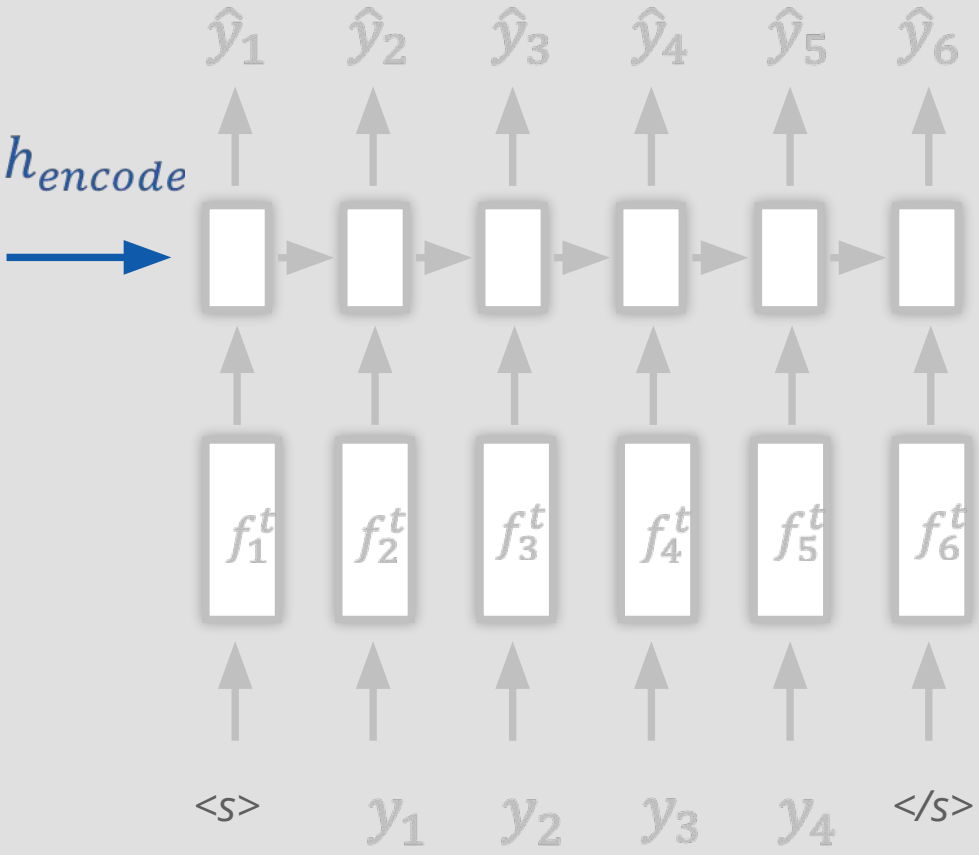
- Decode the whole sentence

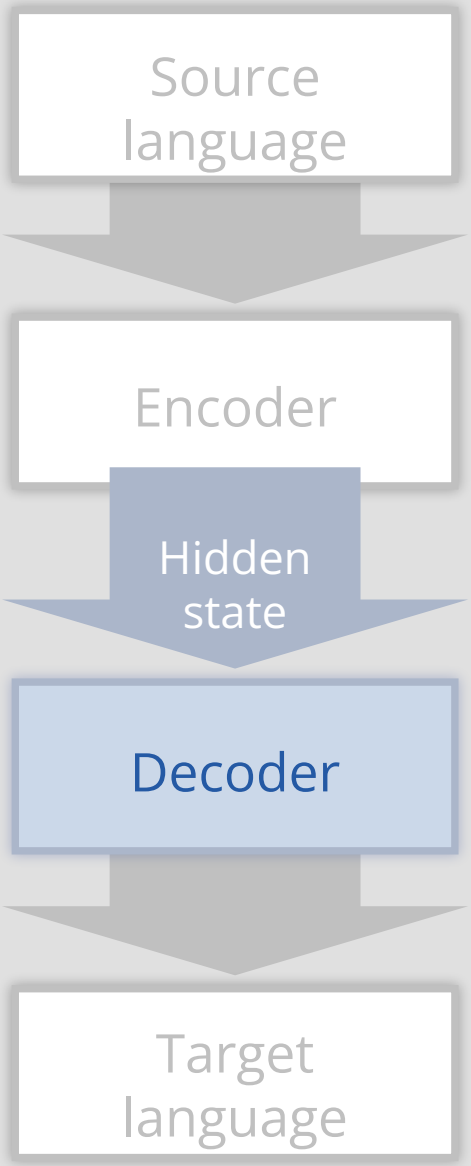
h_{encode}



Decoder architecture

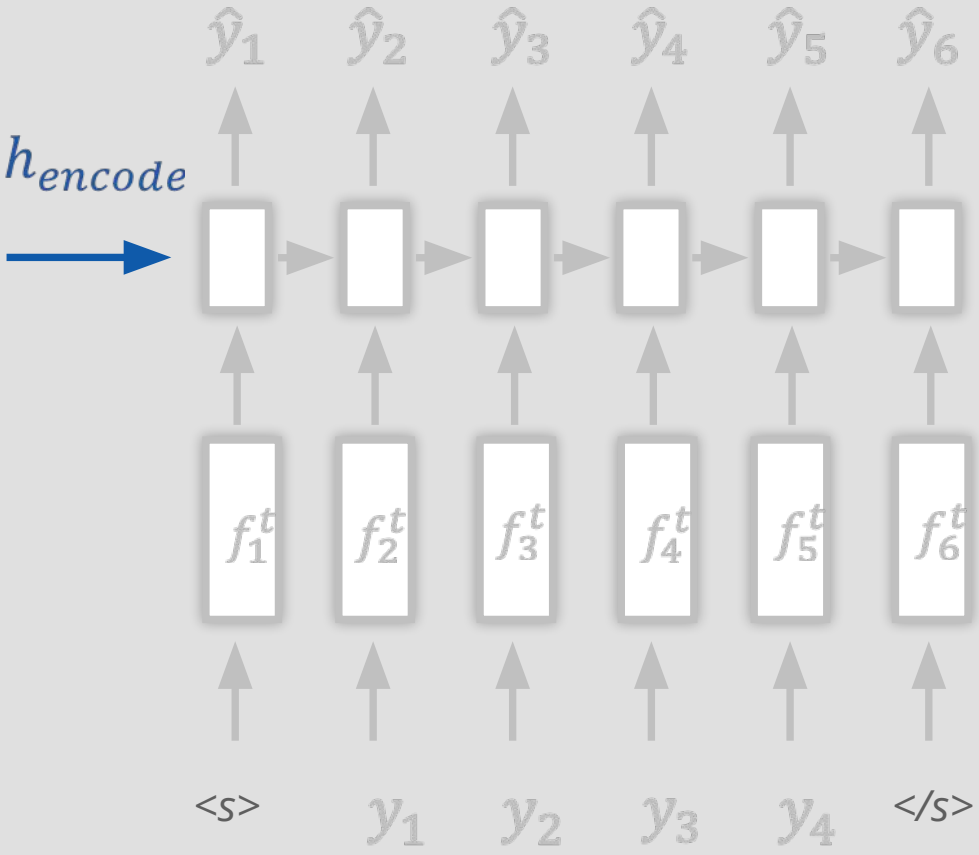
- Decode the whole sentence

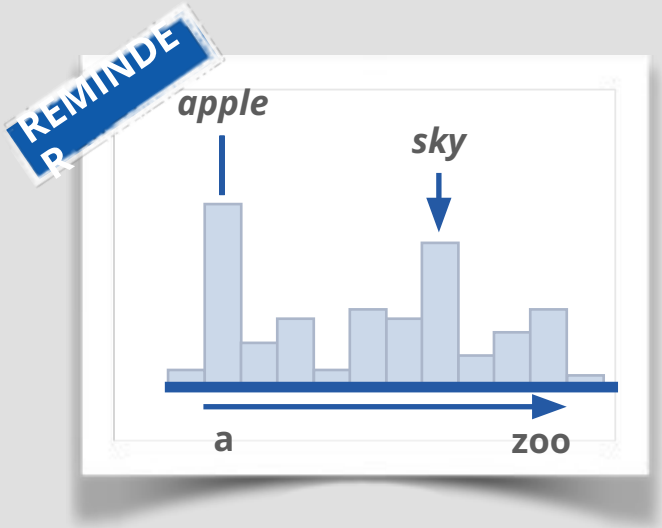




Decoder architecture

- Decode the whole sentence

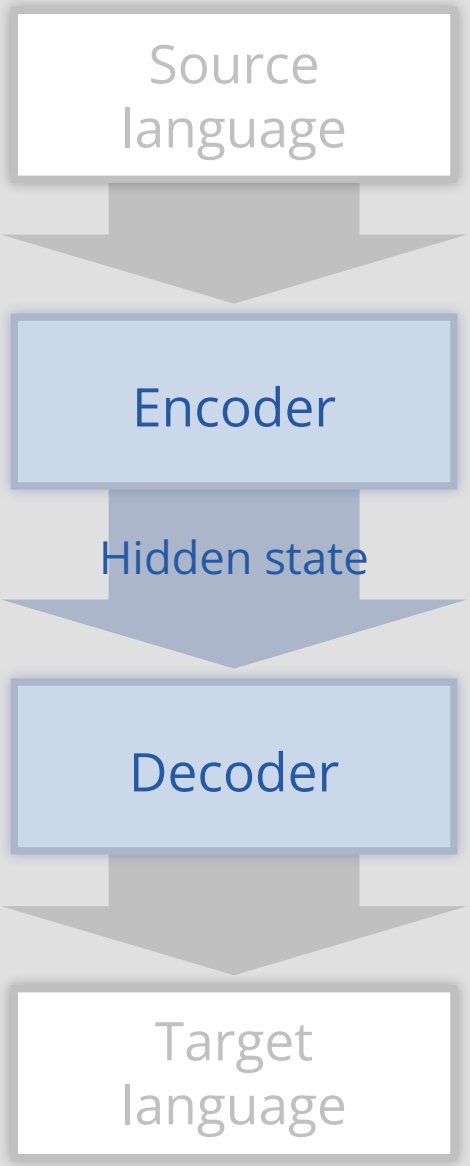




Training process

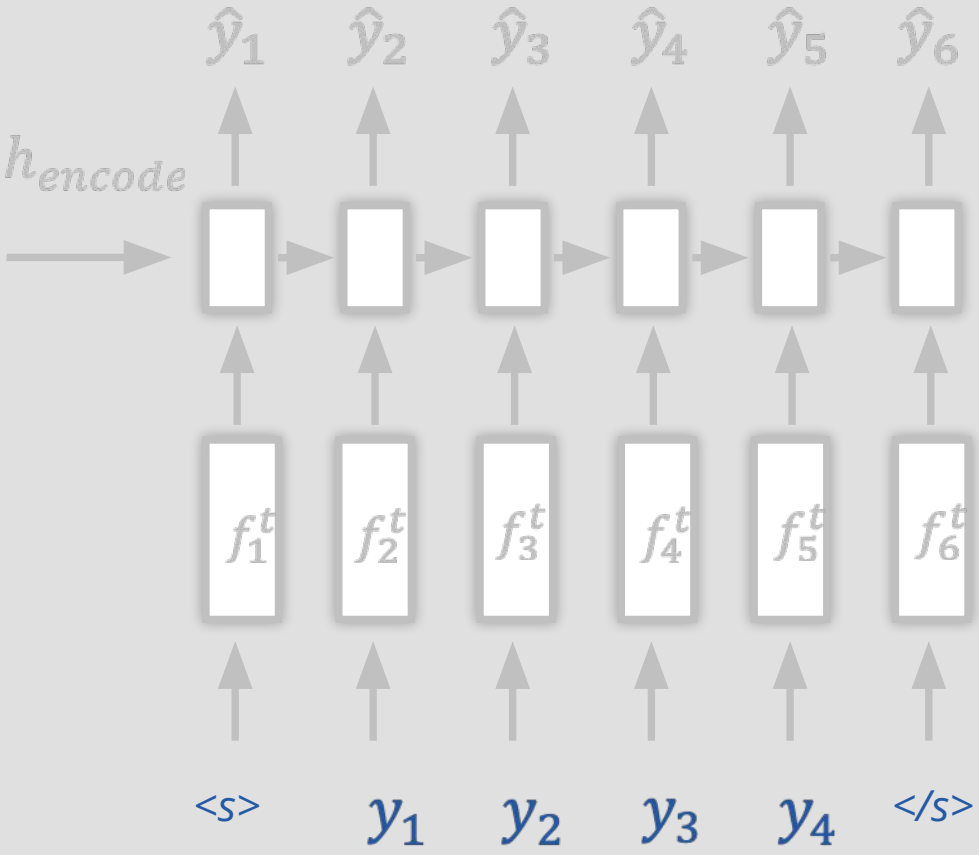
- Loss function

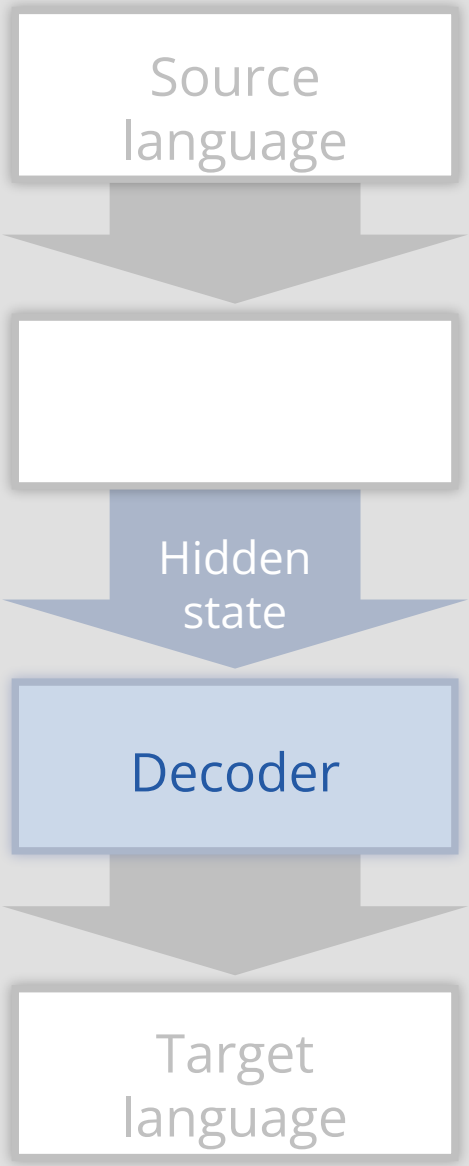
$$L(\mathbf{y}, \hat{\mathbf{y}}) = \sum_{j=1}^{V_{target}} y_j \log \hat{y}_j$$



Decoder architecture

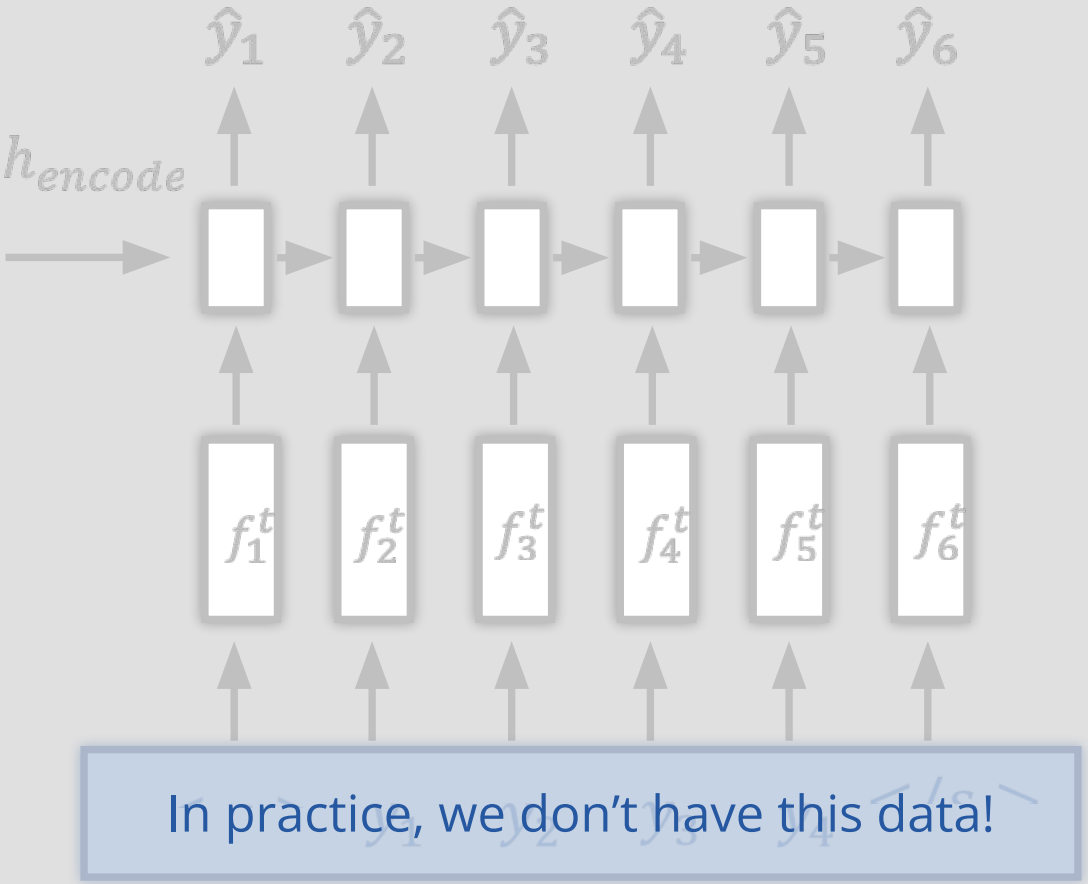
- Decode the whole sentence

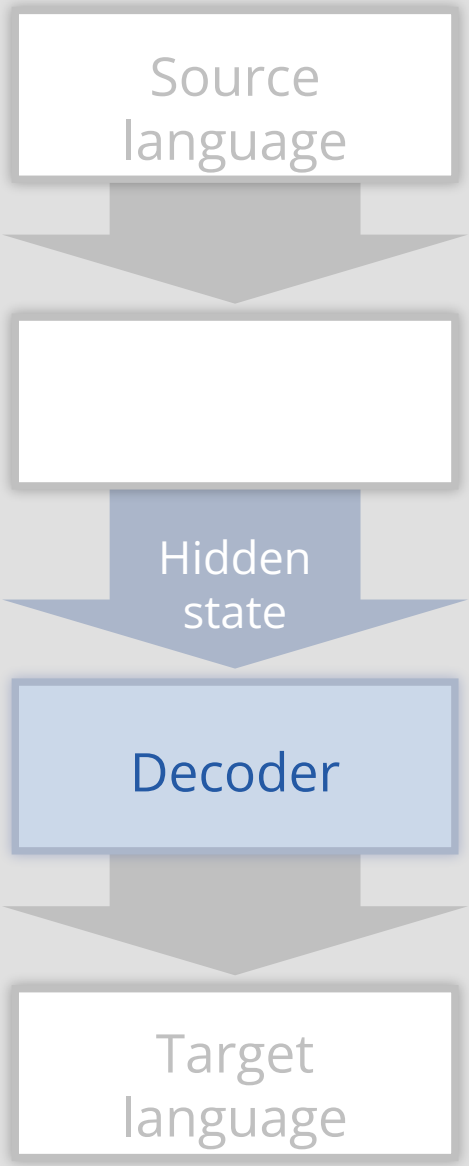




Decoder architecture

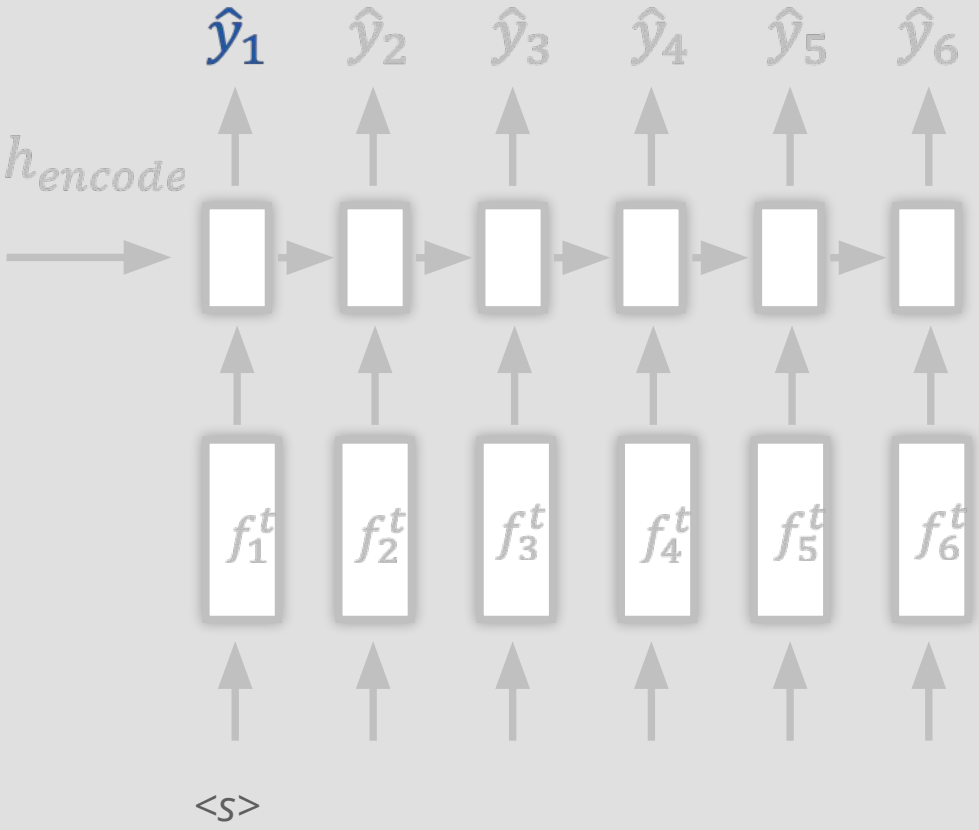
- Decode the whole sentence

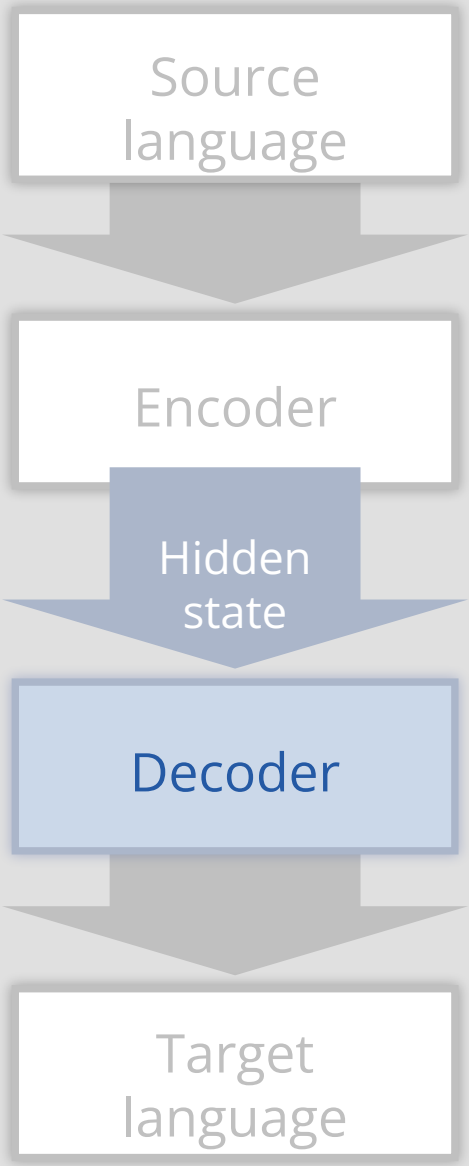




Decoder architecture

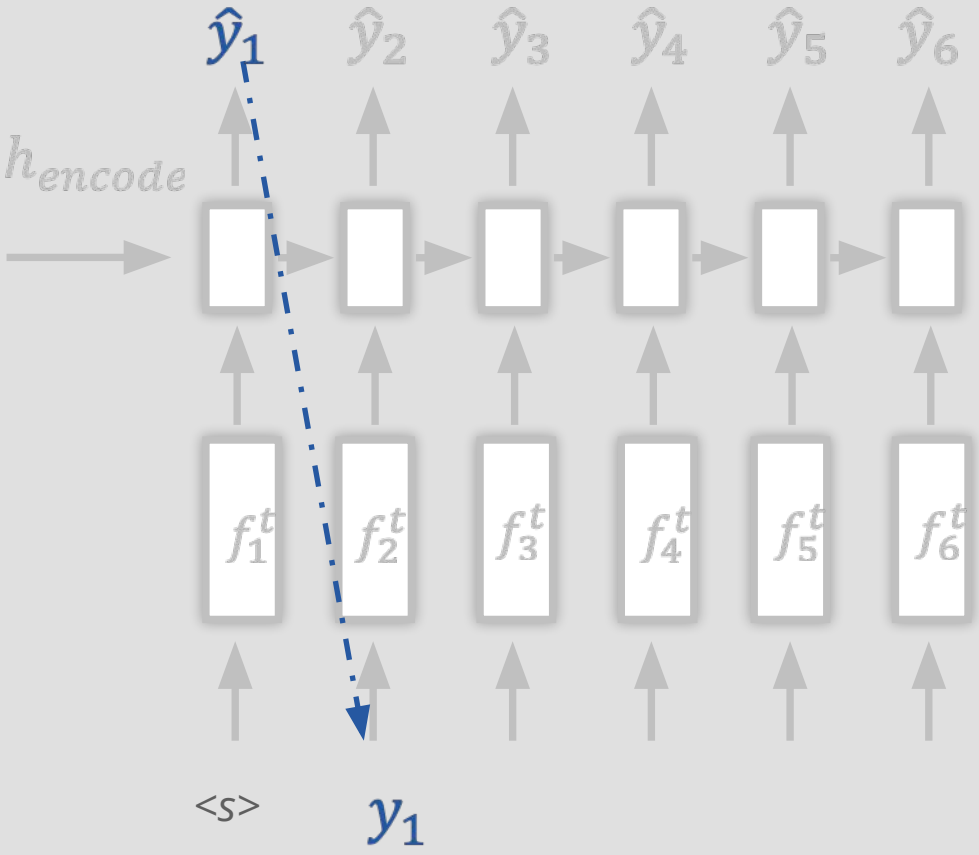
- Decode the whole sentence

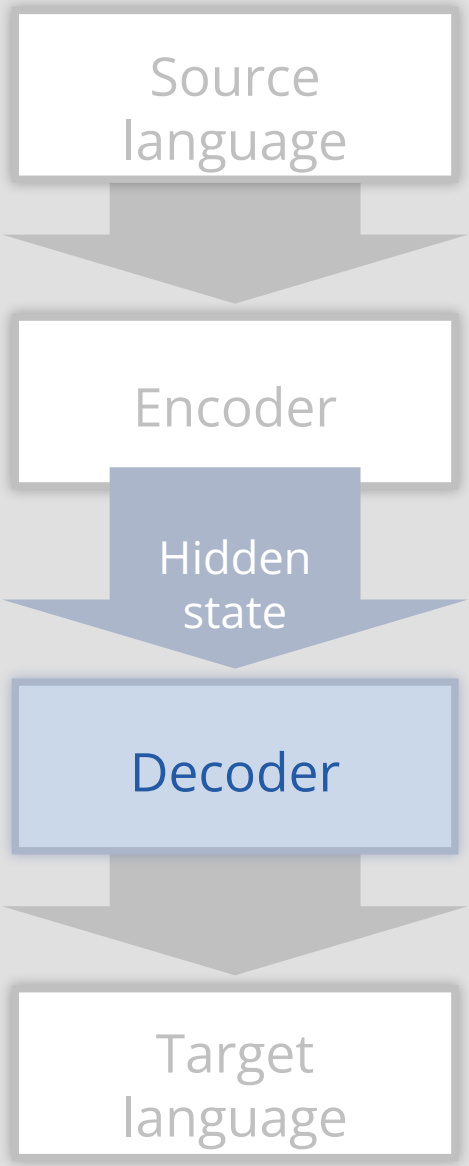




Decoder architecture

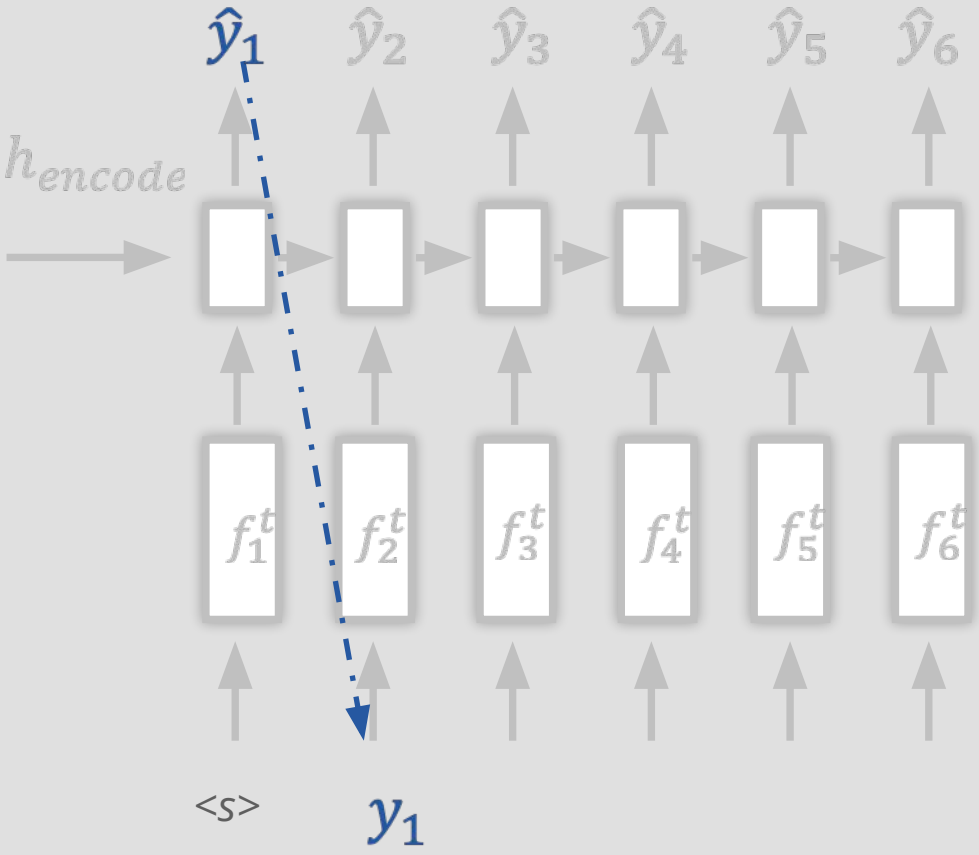
- Decode the whole sentence

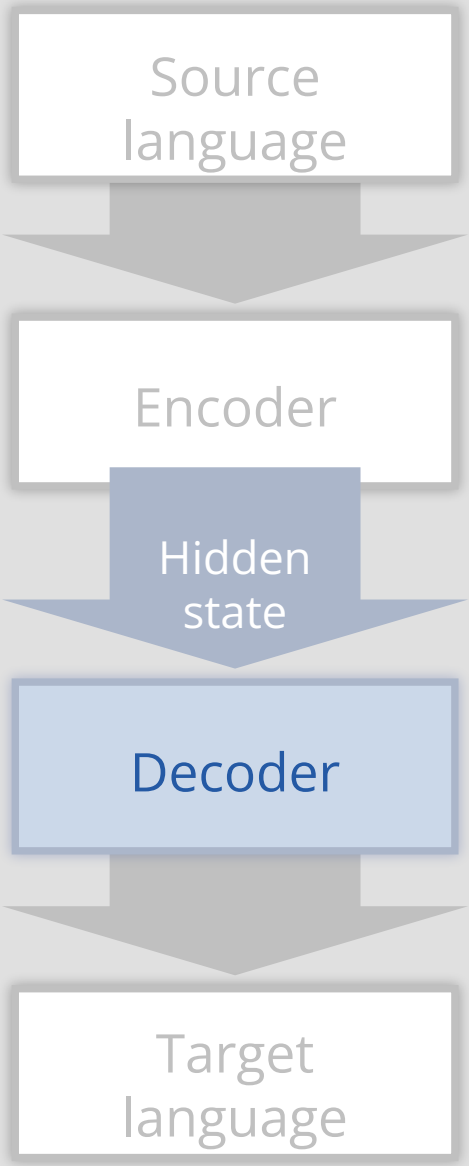




Decoder architecture

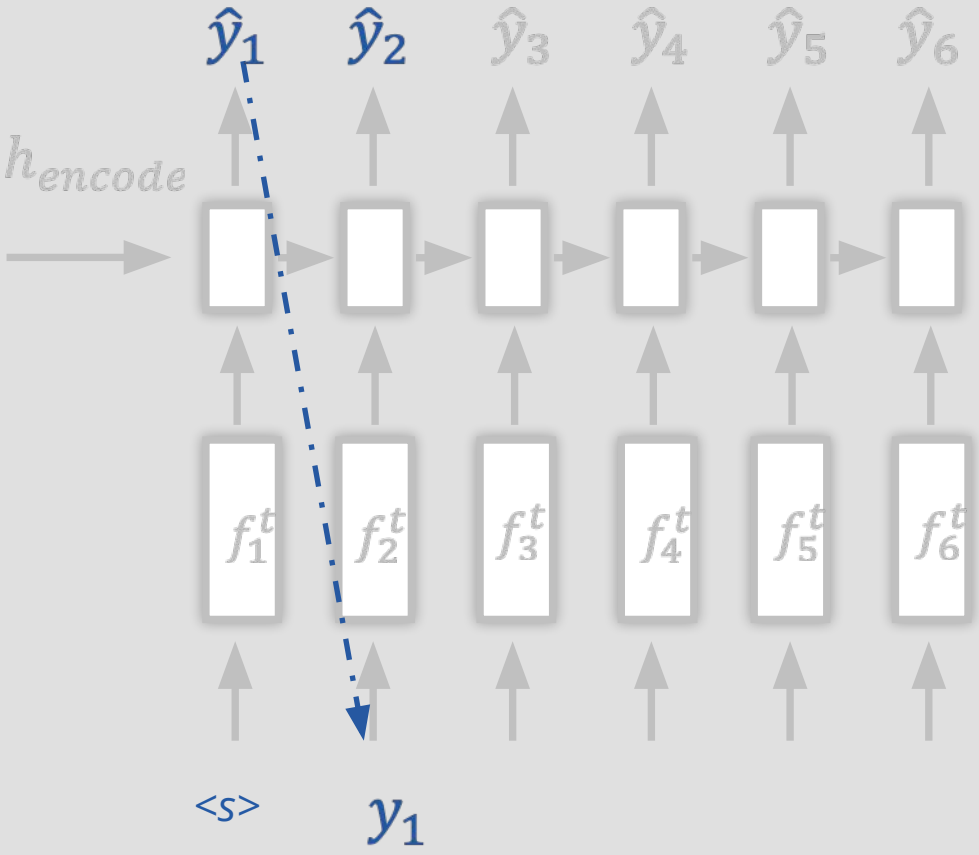
- Decode the whole sentence

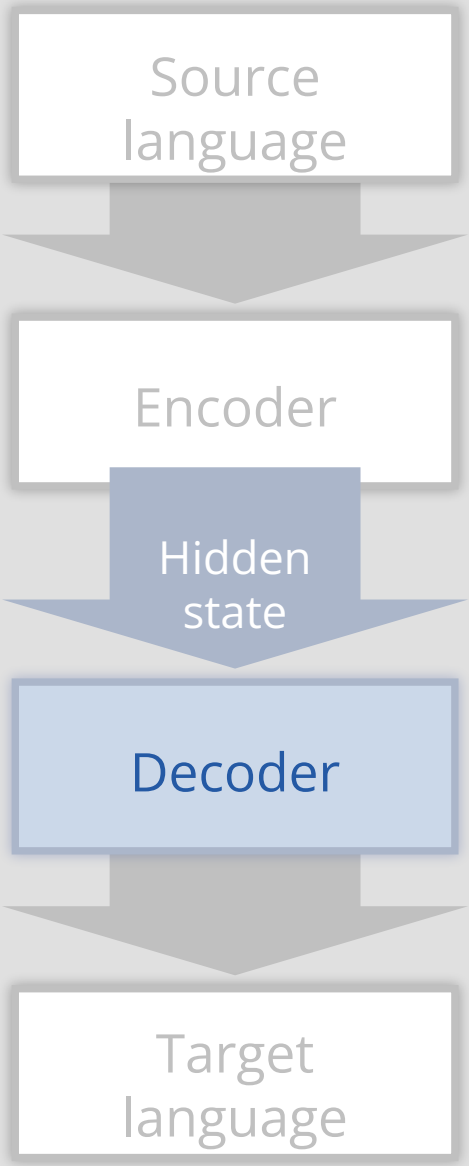




Decoder architecture

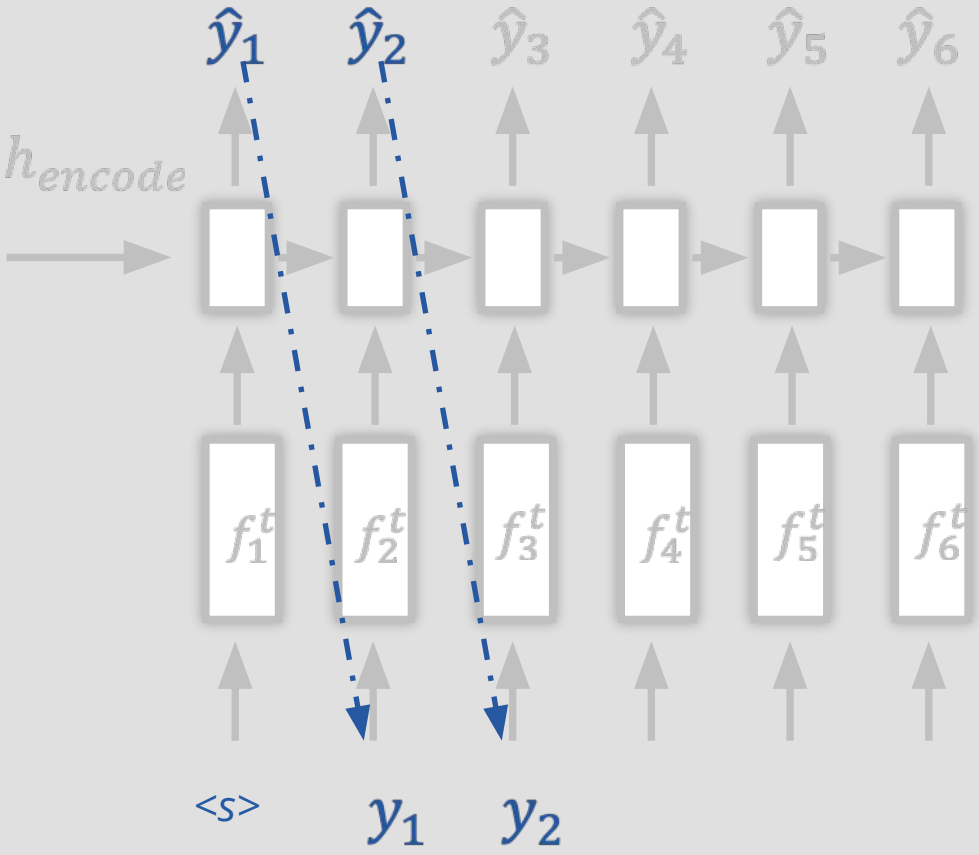
- Decode the whole sentence

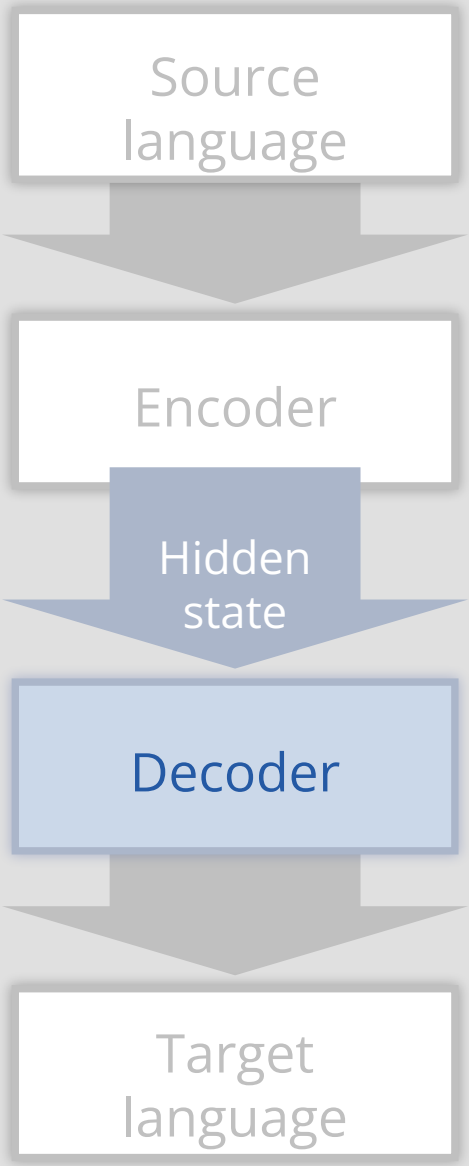




Decoder architecture

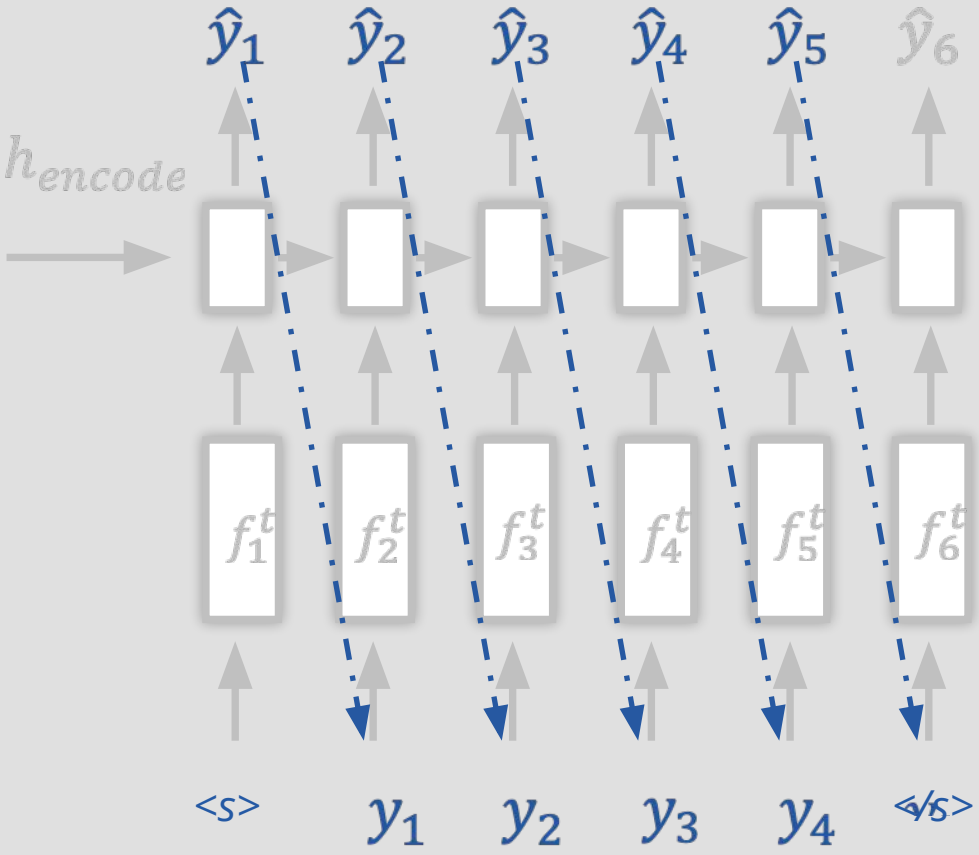
- Decode the whole sentence

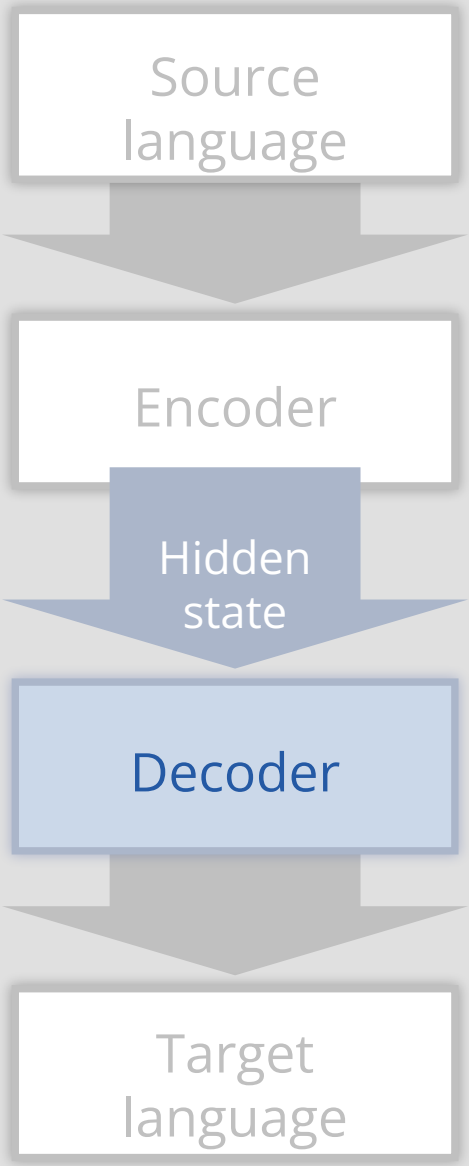




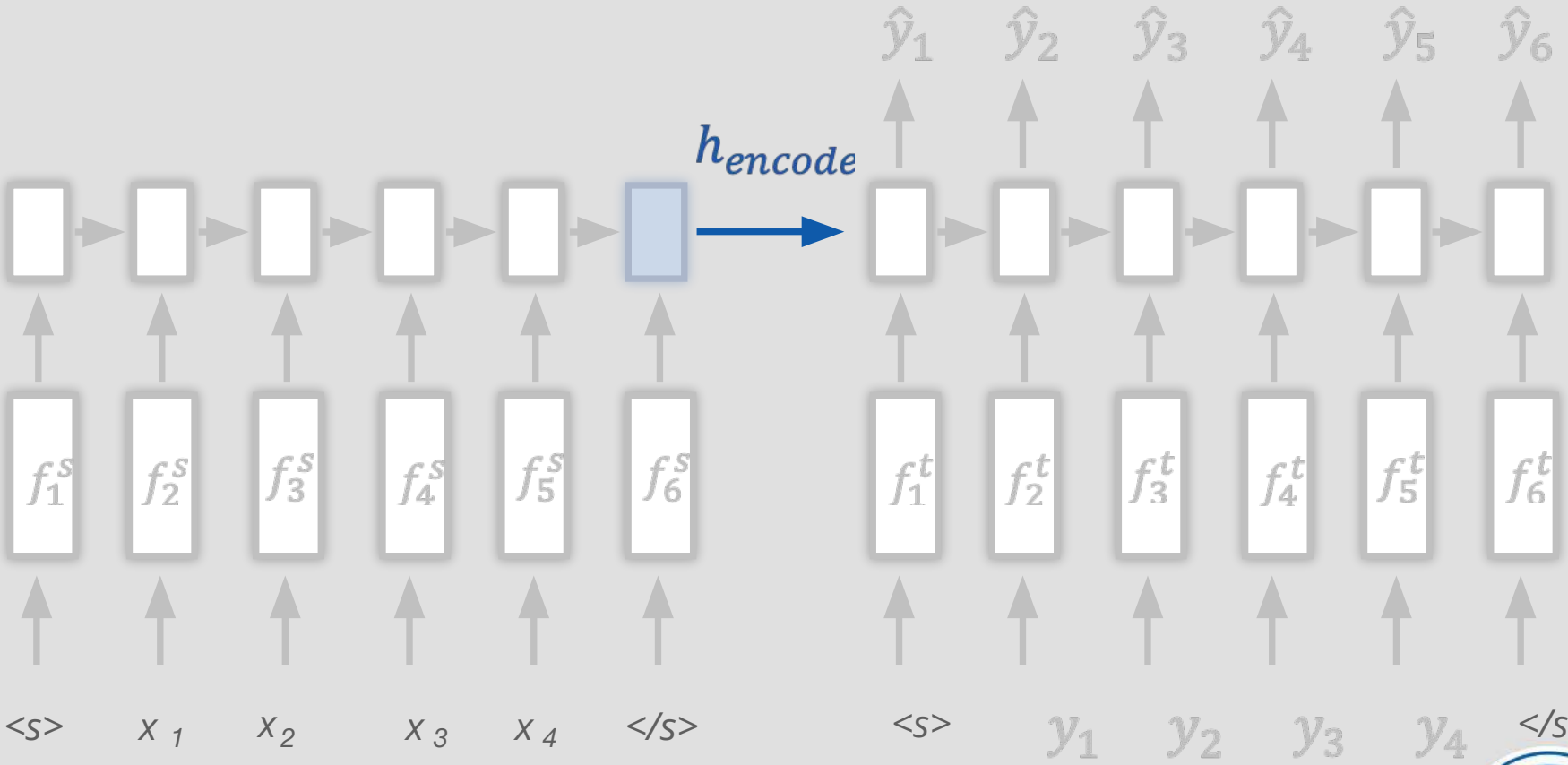
Decoder architecture

- Decode the whole sentence



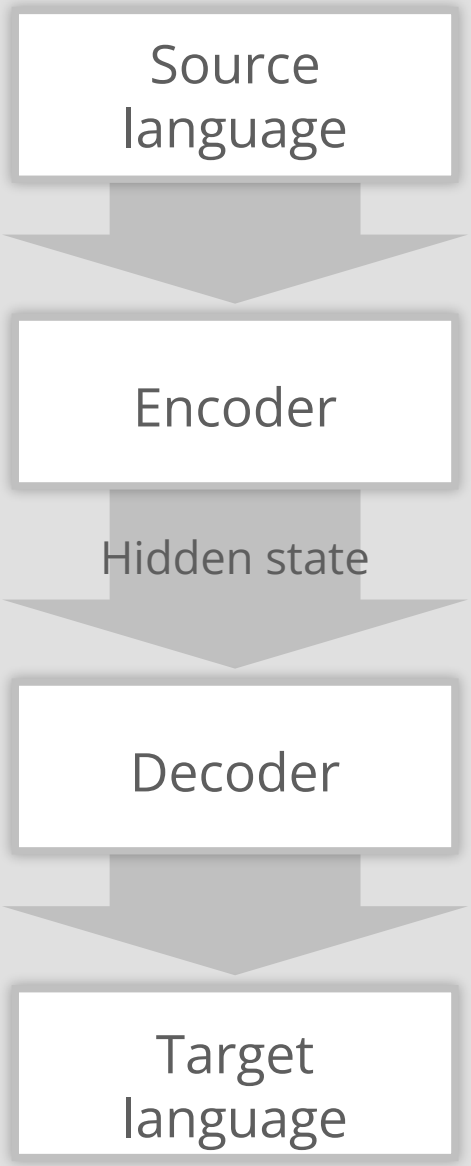


Neural machine translation



Encoder-decoder architecture

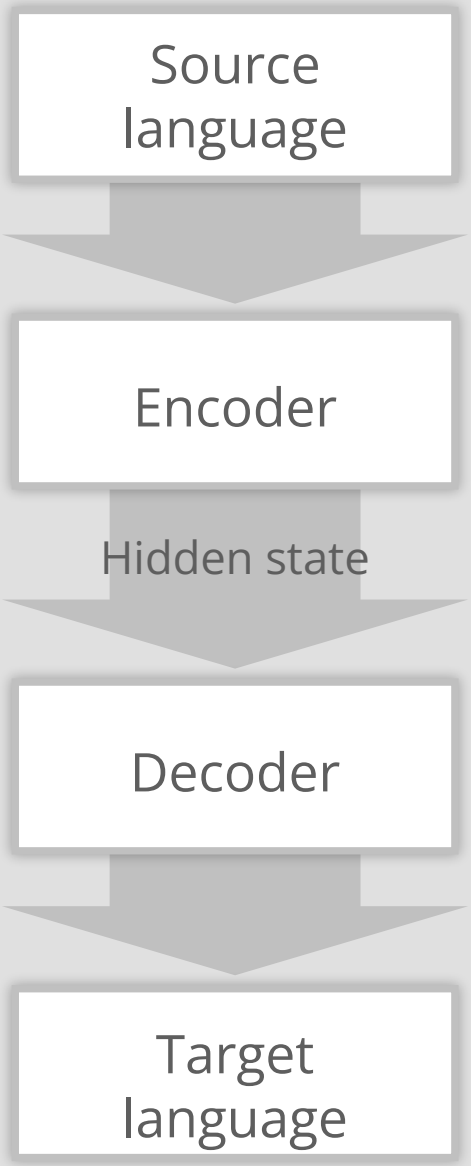
Consists of two parts:



Encoder-decoder architecture

Consists of two parts:

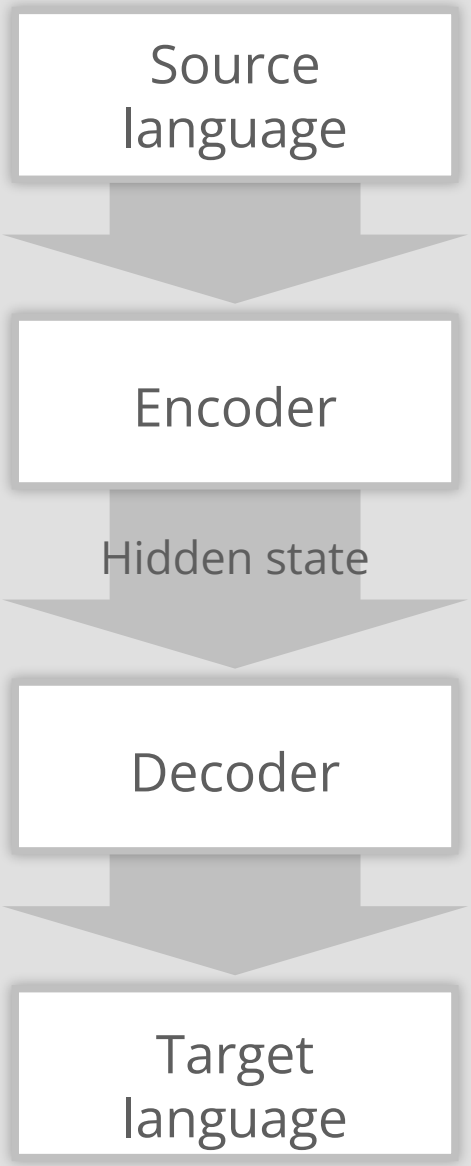
- Encoder encodes input word sequence



Encoder-decoder architecture

Consists of two parts:

- Encoder encodes input word sequence
- Decoder generates the output words sequence

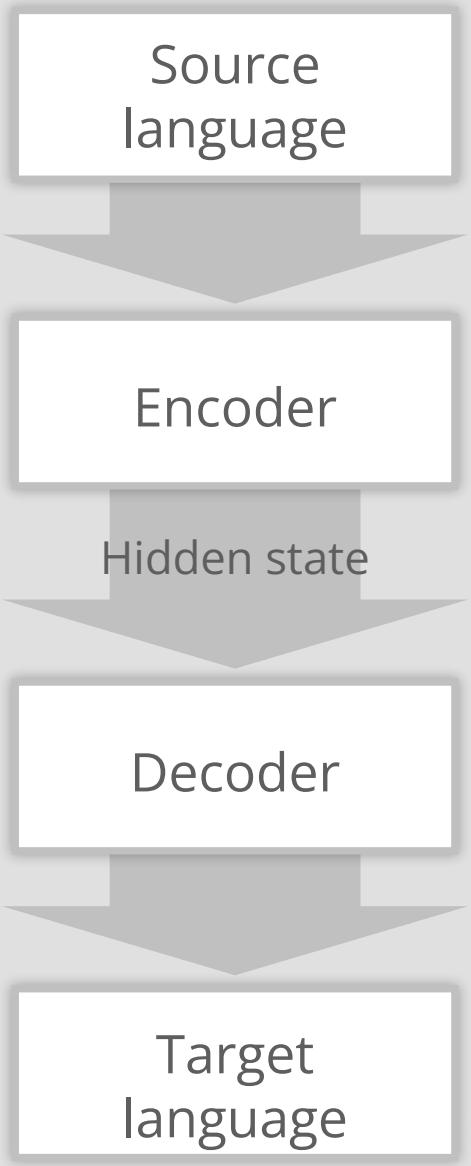


Encoder-decoder architecture

Consists of two parts:

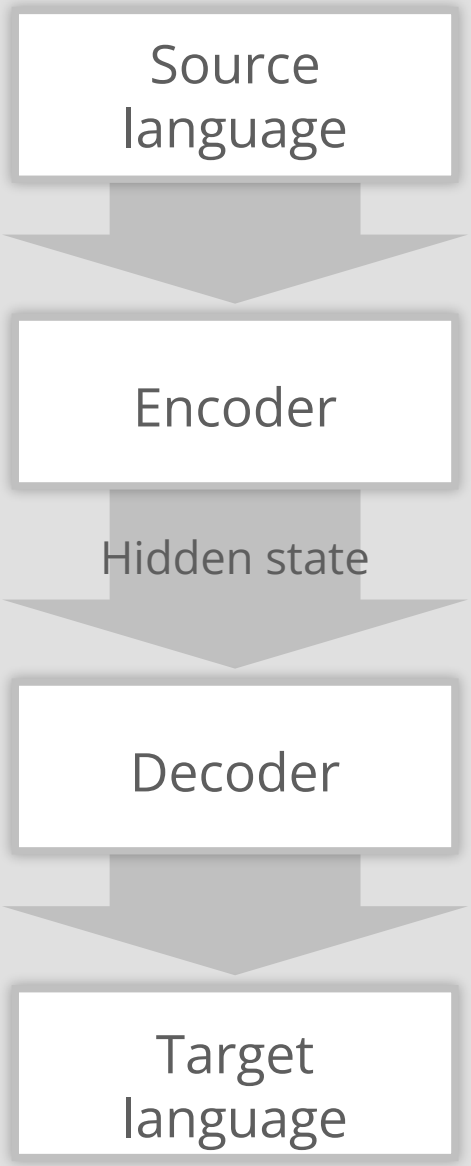
- Encoder encodes input word sequence
- Decoder generates the output words sequence

Decoder is a language model!



Encoder-decoder architecture

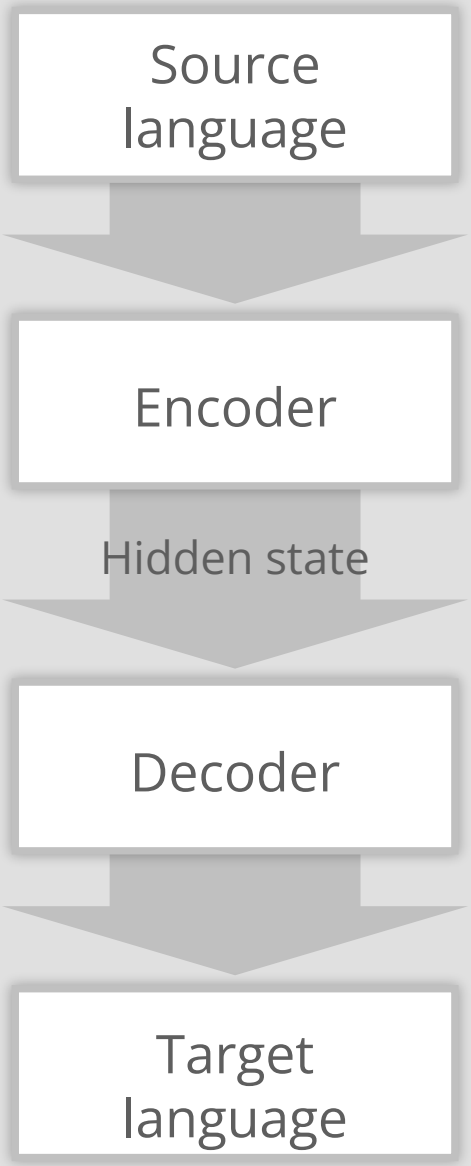
Encoder-decoder architectures can be used in various tasks:



Encoder-decoder architecture

Encoder-decoder architectures can be used in various tasks:

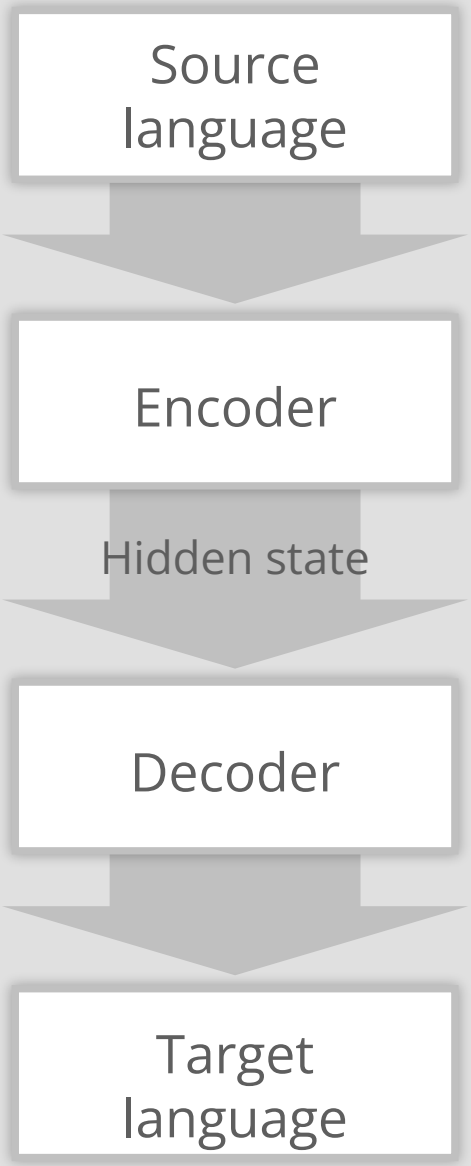
- question answering



Encoder-decoder architecture

Encoder-decoder architectures can be used in various tasks:

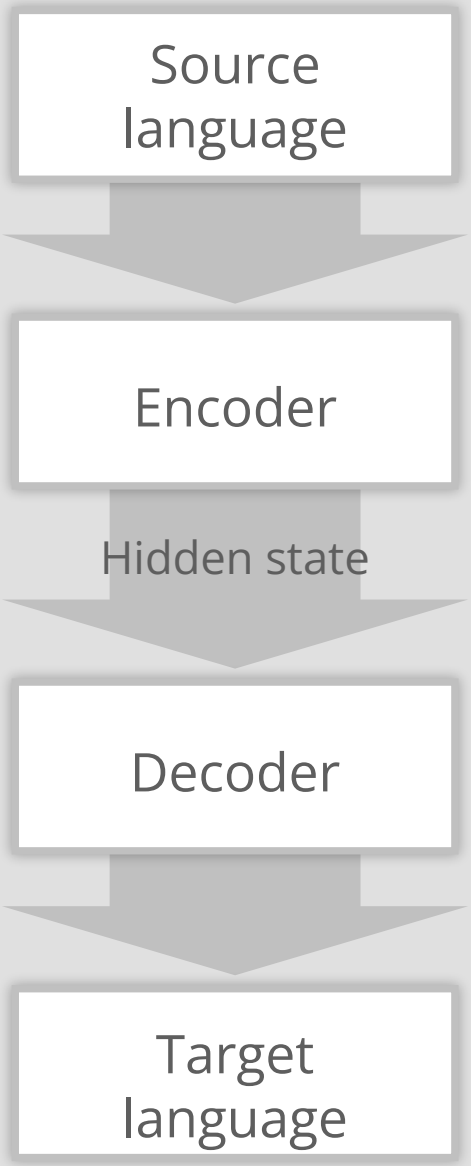
- question answering
- Image capturing



Encoder-decoder architecture

Encoder-decoder architectures can be used in various tasks:

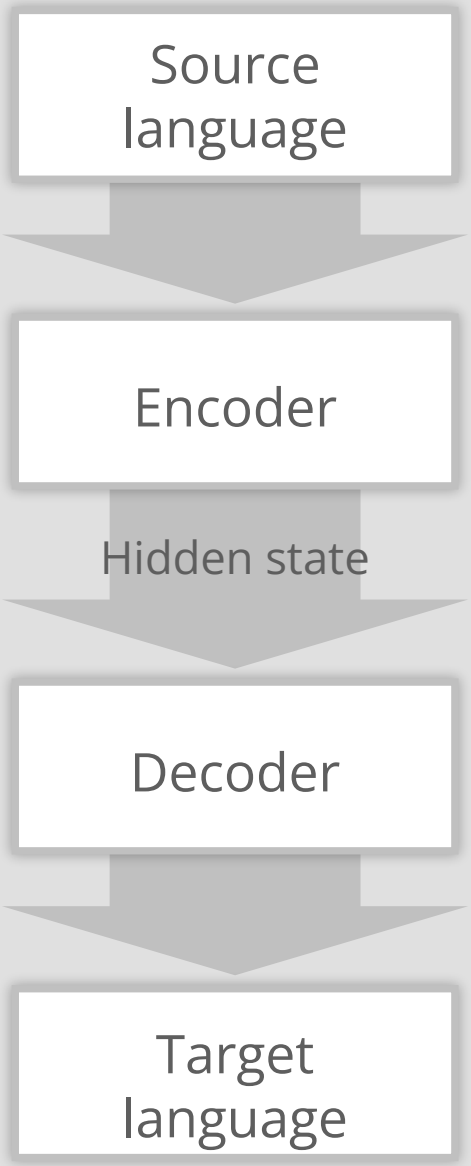
- question answering
- Image capturing
- Speech recognition



Encoder-decoder architecture

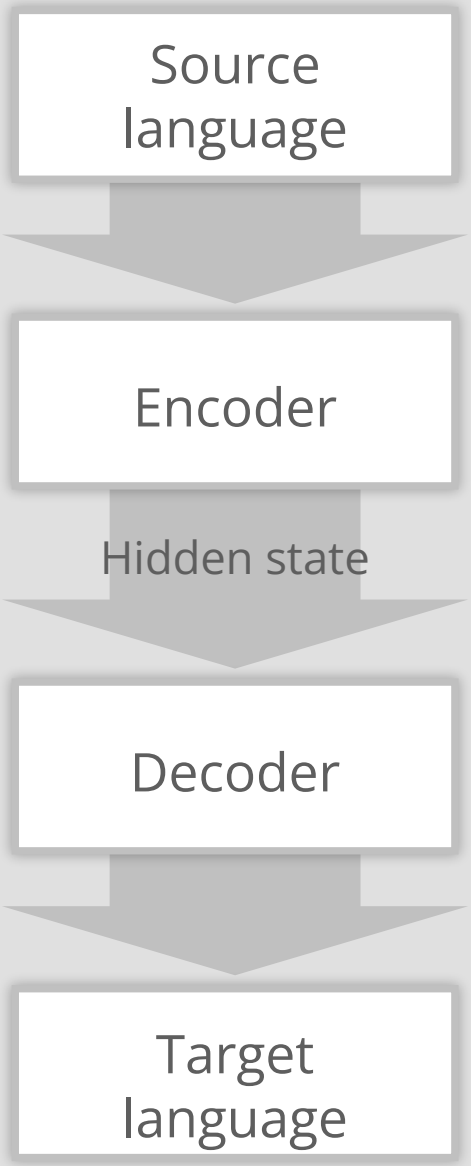
Encoder-decoder architectures can be used in various tasks:

- question answering
- Image capturing
- Speech recognition
- Code switching



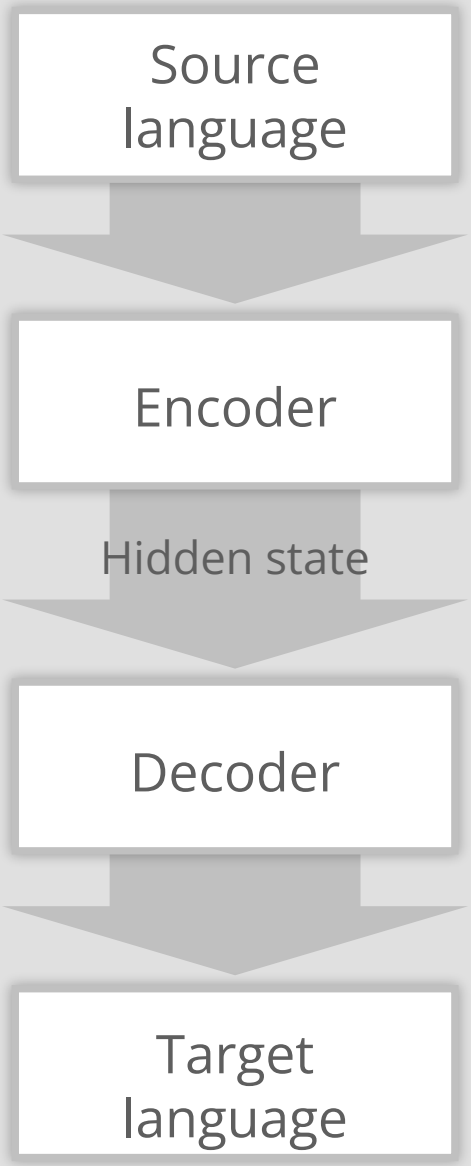
Encoder-decoder architecture

- Problem: only the last decoder state is used



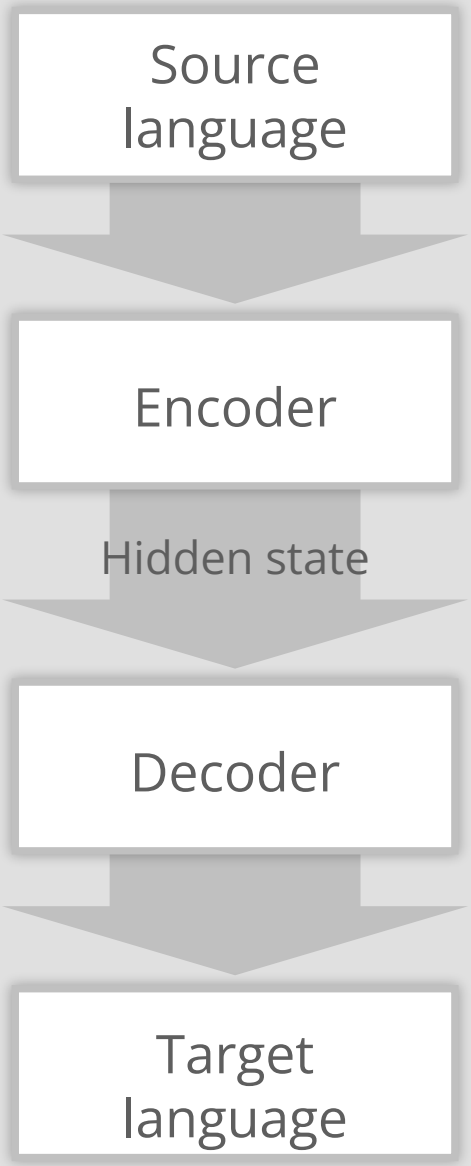
Encoder-decoder architecture

- Problem: only the last decoder state is used
- Last words have more impact than the first ones



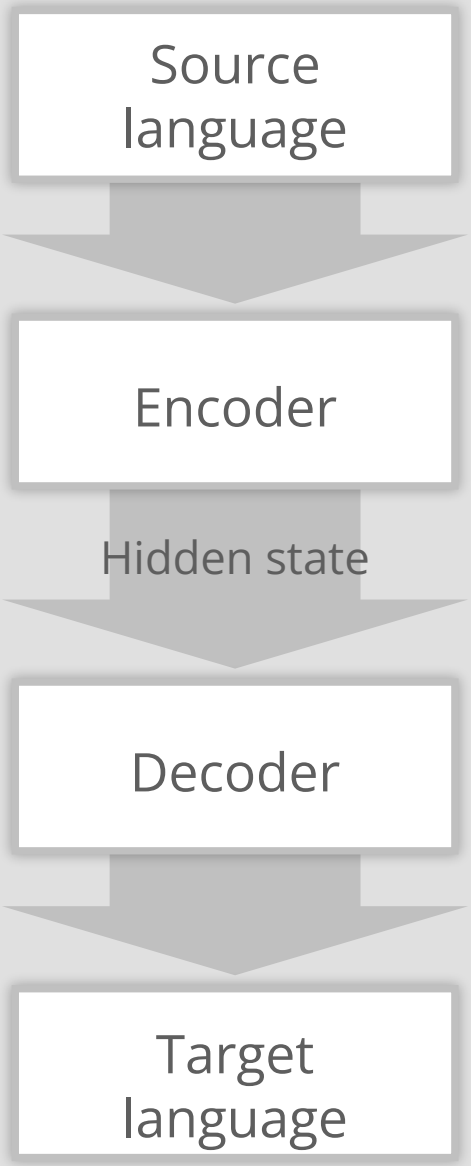
Encoder-decoder architecture

- Problem: only the last decoder state is used
- Last words have more impact than the first ones
- Short sentences are processed better



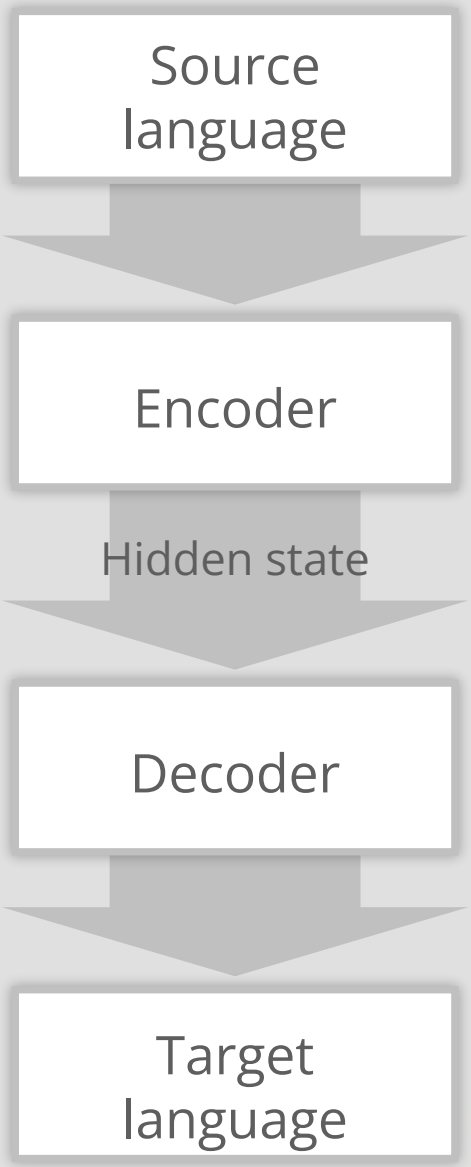
Encoder-decoder architecture

- Problem: only the last decoder state is used
- Last words have more impact than the first ones
- Short sentences are processed better
- The architecture can be improved by adding attention mechanism

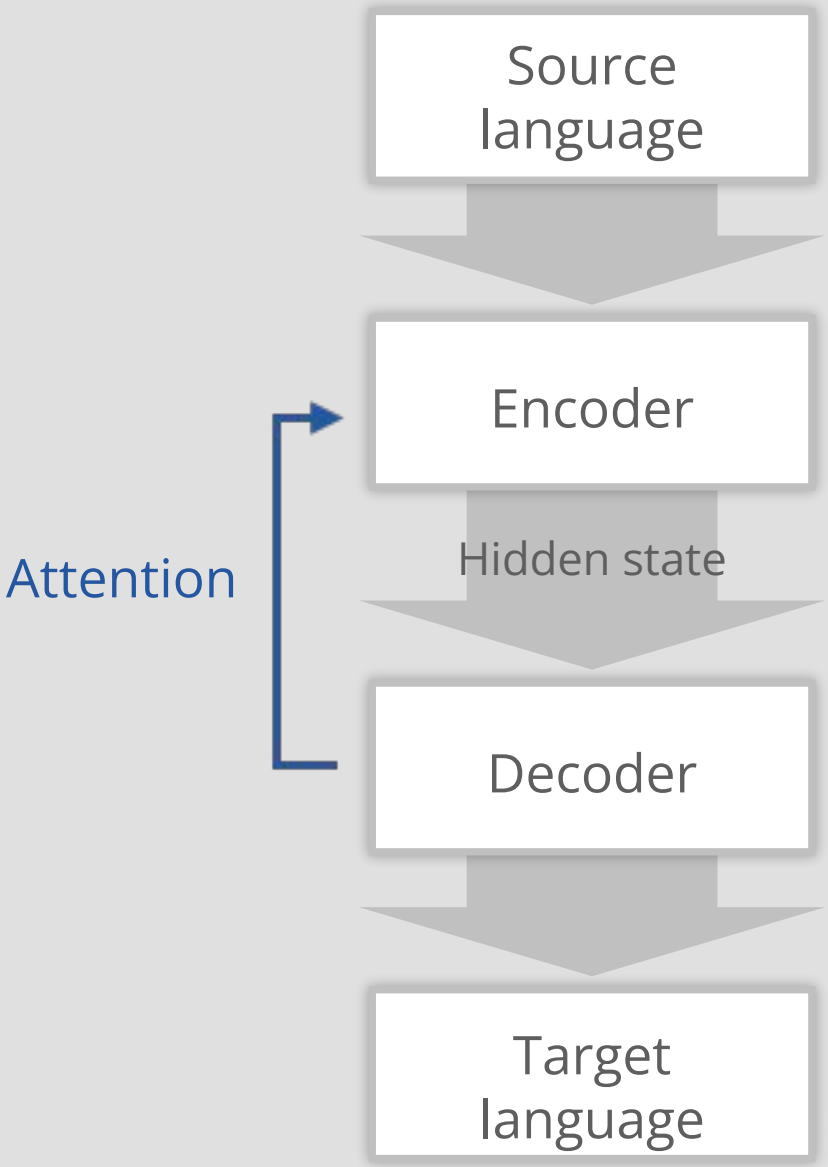


Attention mechanism

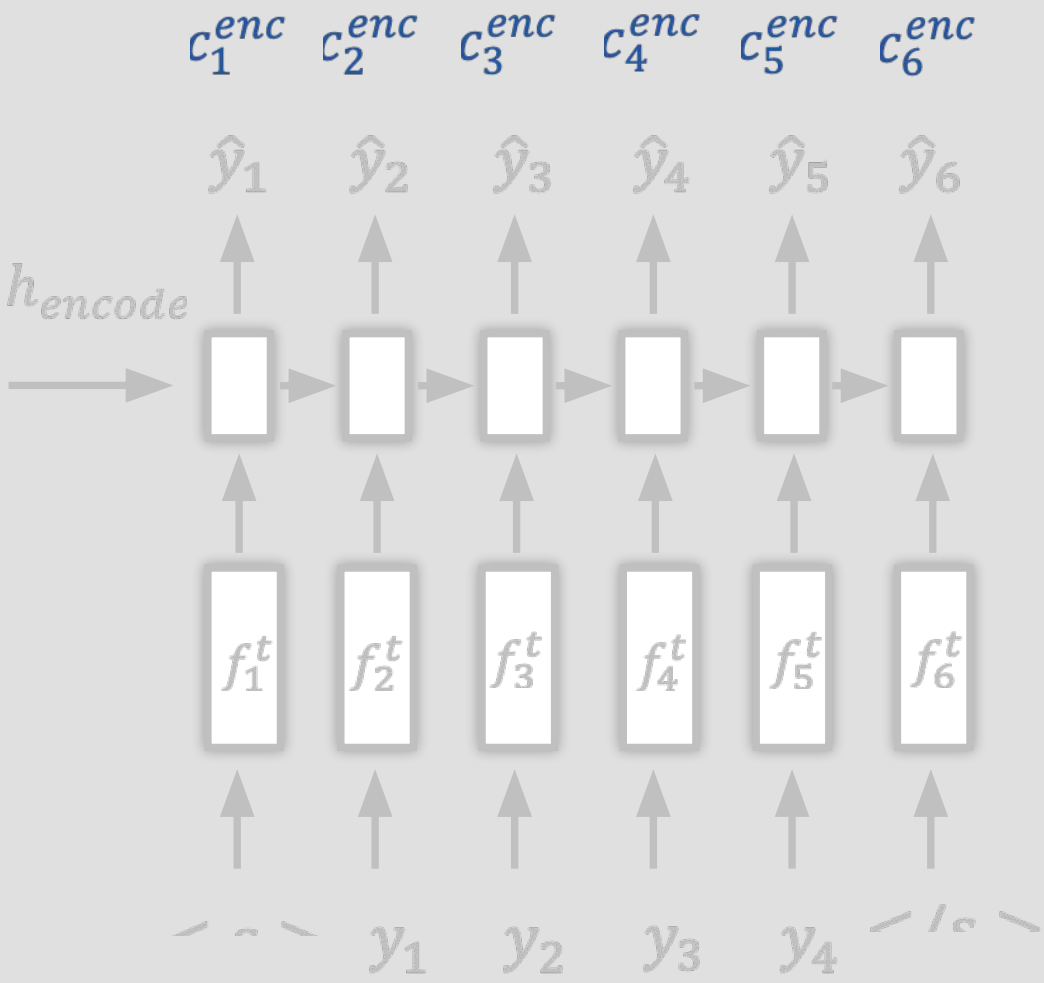
Attention mechanism

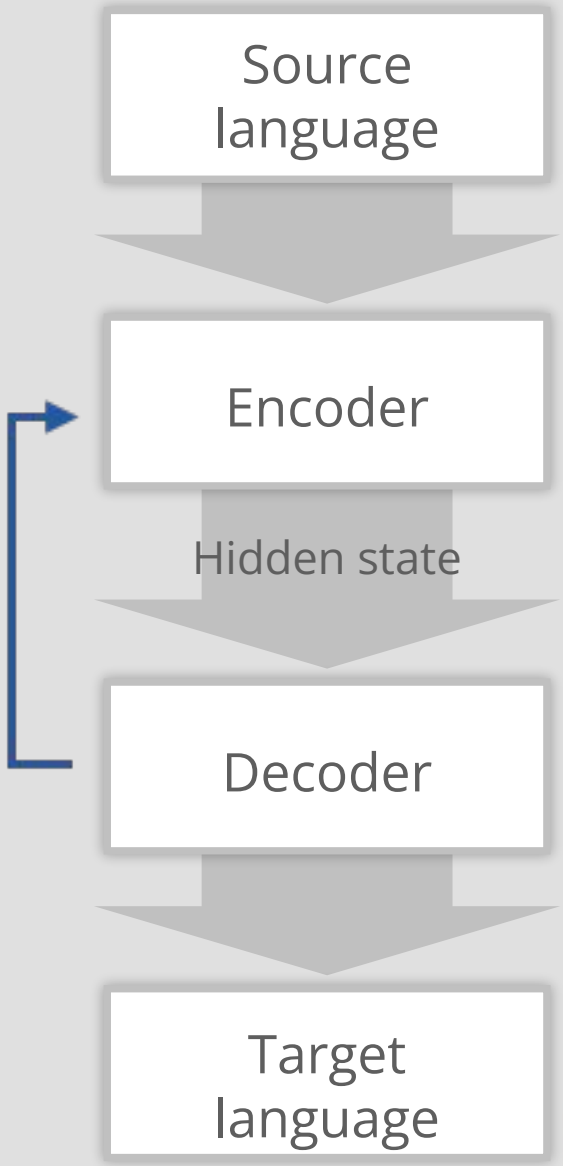


Attention mechanism

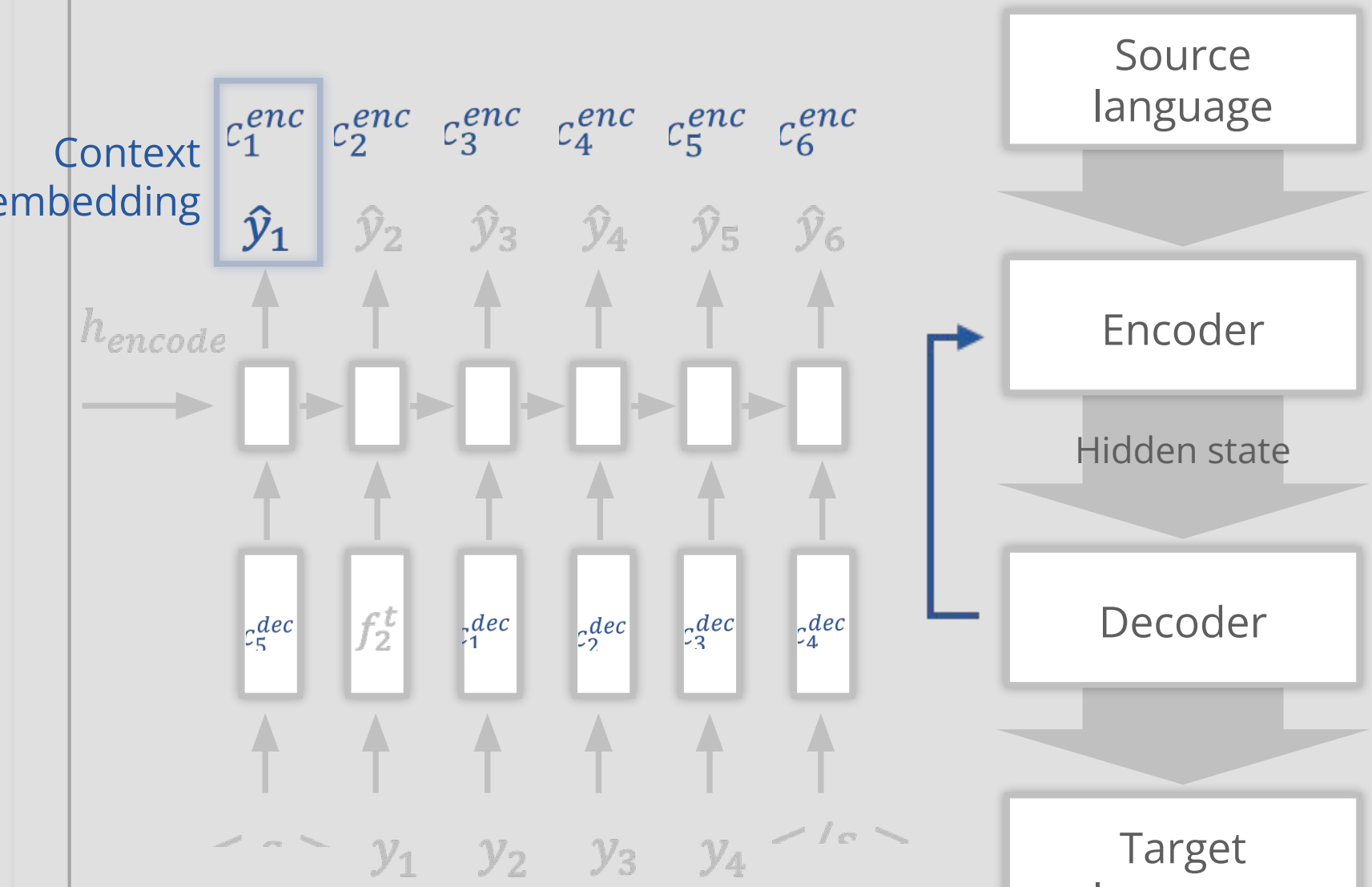


Attention mechanism





Attention mechanism



Attention mechanism

- Weights:

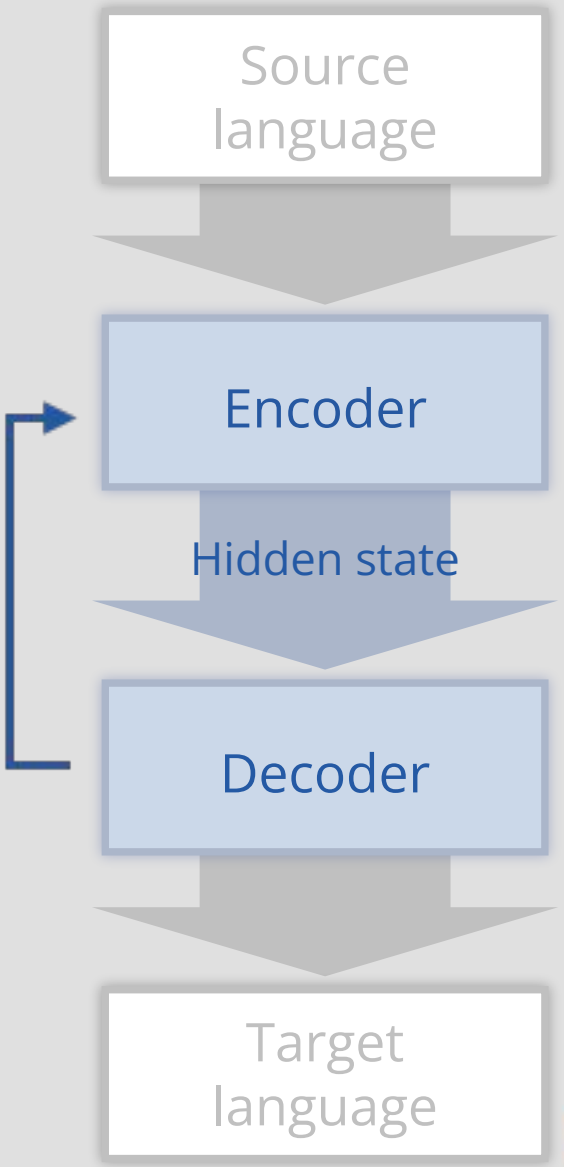
$$a_{ij} = sim(h_i^{enc}, h_j^{dec})$$

- Normalization:

$$\alpha_{ij} = \frac{e^{a_{ij}}}{\sum_k e^{a_{kj}}}$$

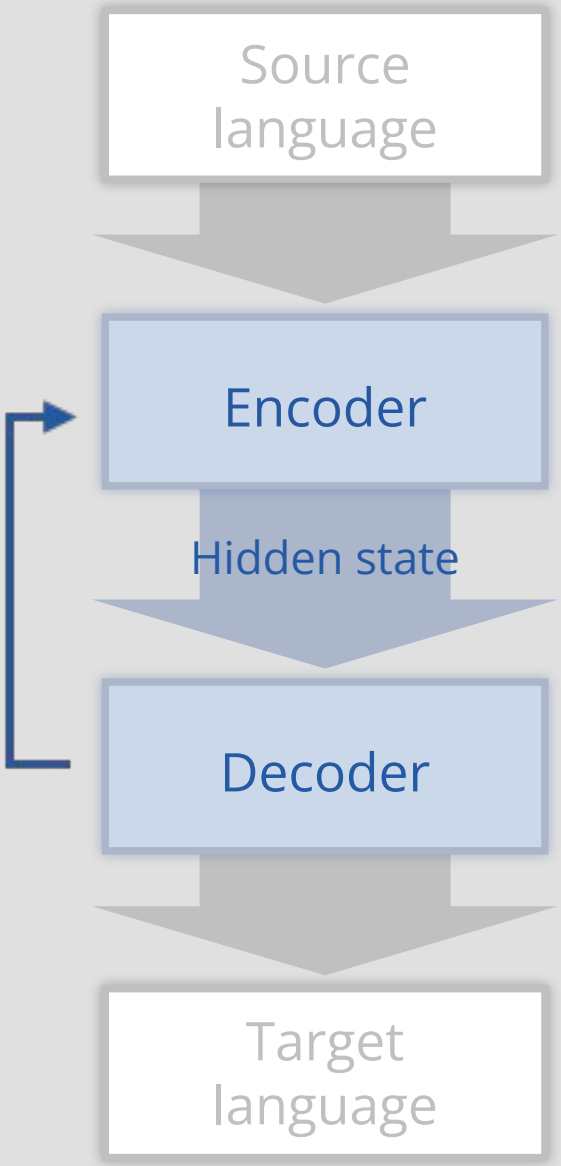
- Context embedding:

$$c_j = \sum_i \alpha_{ij} h_i^{enc}$$



Attention mechanism

- dot-product: $sim(h, s) = h^\top s$

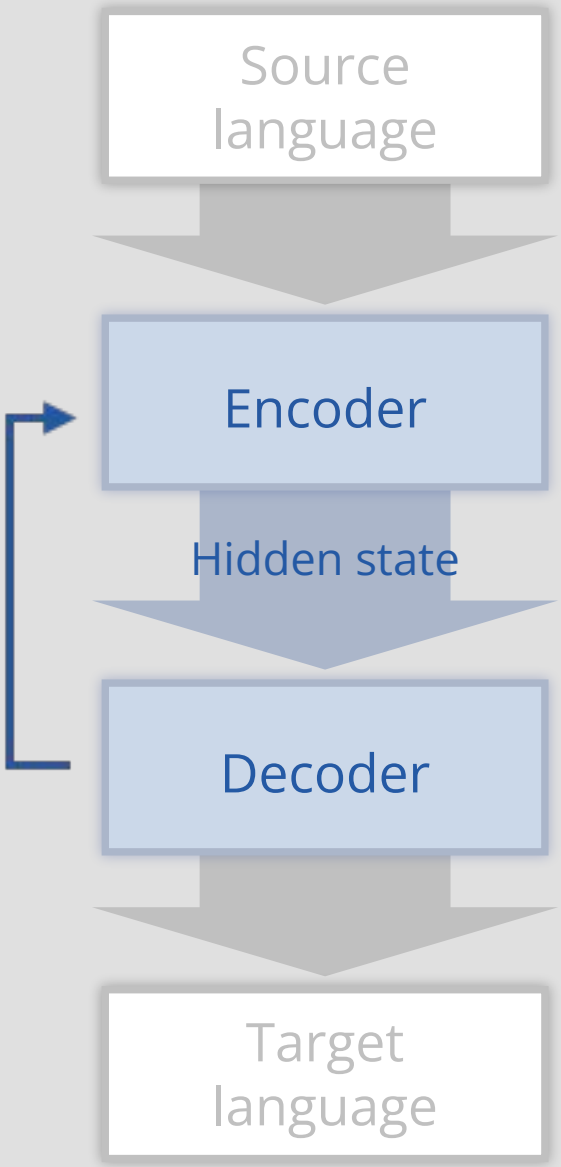


Attention mechanism

- dot-product: $sim(h, s) = h^\top s$

- Additive:

$sim(h, s) = w^\top \tanh(W_h h + W_s s)$



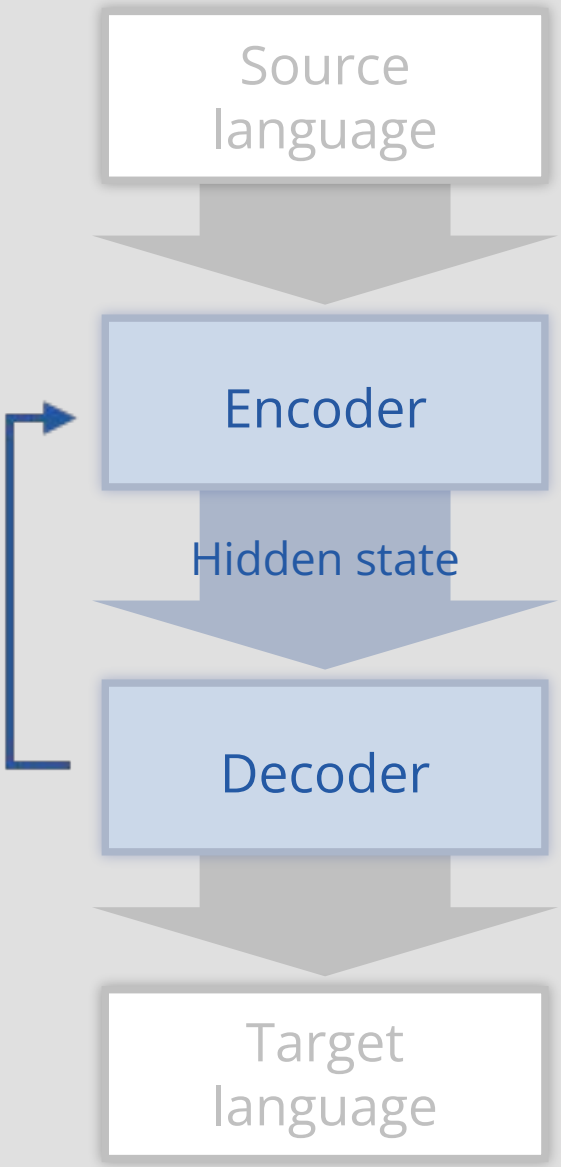
Attention mechanism

- dot-product: $sim(h, s) = h^\top s$
- Additive:

$$sim(h, s) = w^\top \tanh(W_h h + W_s s)$$

- Multiplicative:

$$sim(h, s) = h^\top W s$$



Attention mechanism

- dot-product: $sim(h, s) = h^\top s$

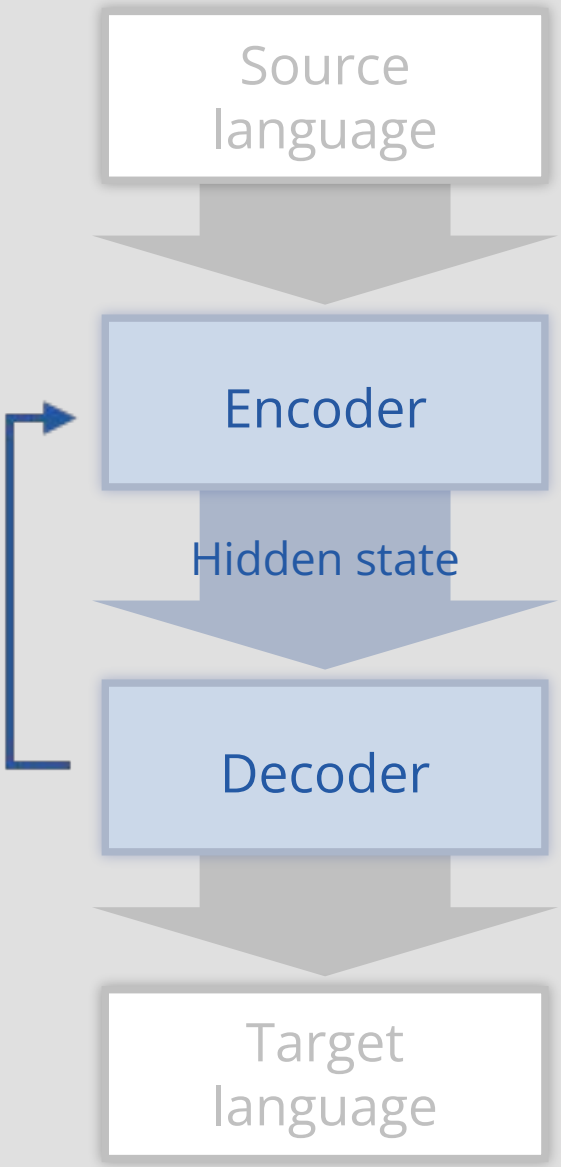
- Additive:

$sim(h, s) = w^\top \tanh(W_h h + W_s s)$

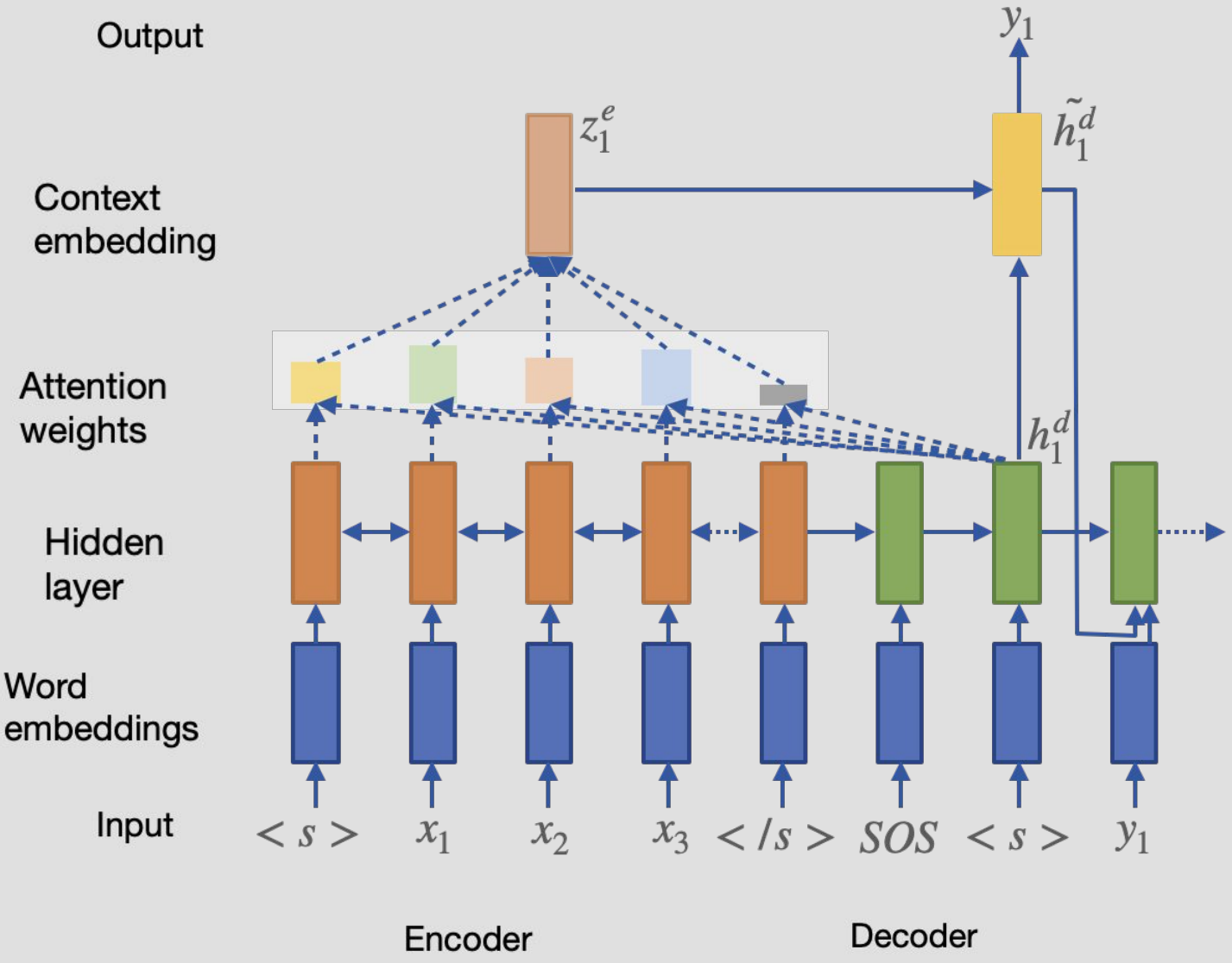
- Multiplicative:

$sim(h, s) = h^\top W s$

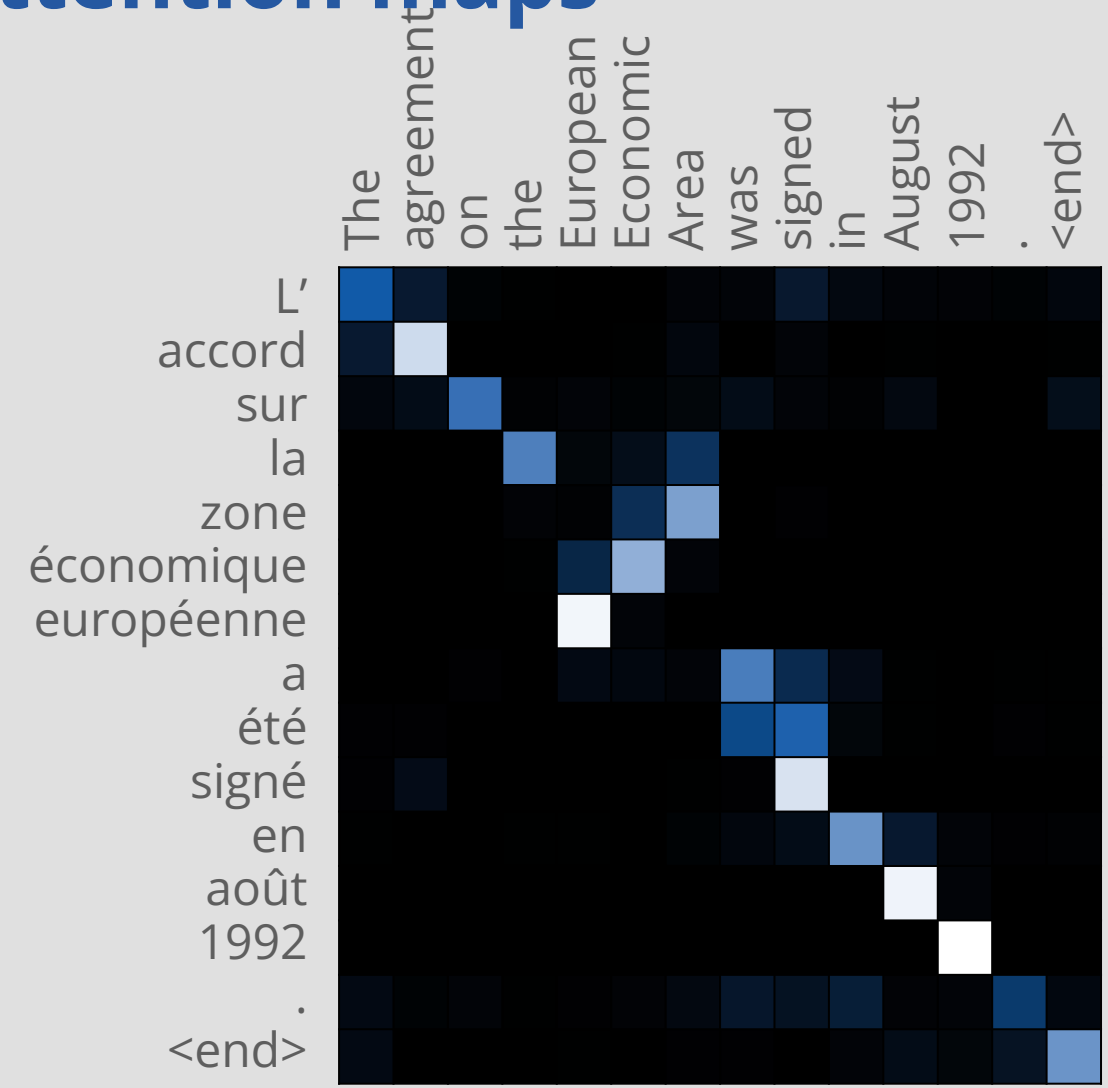
Weights are trained.



Attention mechanism



Attention maps



Source language

Encoder

Hidden state

Decoder

Target language

Machine translation quality metrics

Expert review

- We will show the sentence in the source language and the translation to the experts and ask them to rate the translation on a scale, for example, from 0 to 5

Expert review

- We will show the sentence in the source language and the translation to the experts and ask them to rate the translation on a scale, for example, from 0 to 5

Advantages	Disadvantages
• exact	• expensive • slow

Comparison with the gold standard

- Machine translation model is trained on a parallel corpus: sentence on a source language and several options of translations
- Split data into train and test

Comparison with the gold standard

- Machine translation model is trained on a parallel corpus: sentence on a source language and several options of translations
- Split data into train and test
- Estimate MT quality on a test data

Comparison with the gold standard

- Machine translation model is trained on a parallel corpus: sentence on a source language and several options of translations
- Split data into train and test
- Estimate MT quality on a test data

Advantages	Disadvantages
• quick (if corpus is given)	• corpuses can become obsolete

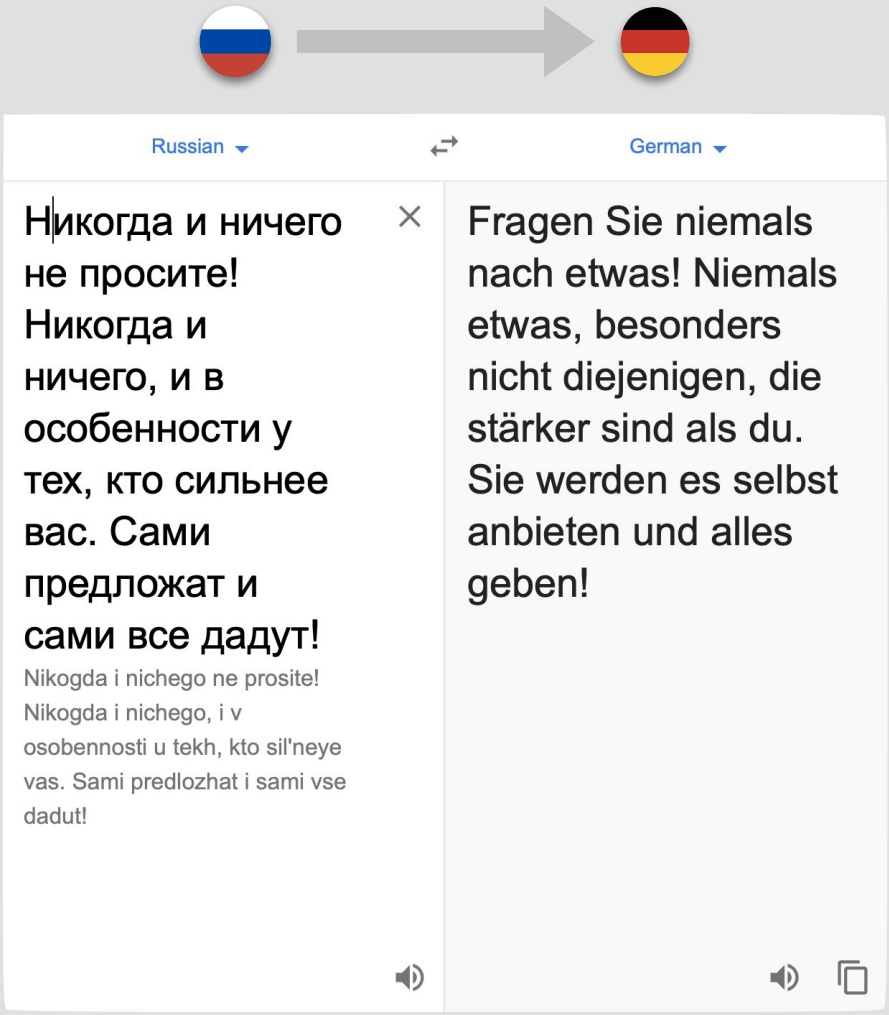
Back translation

- Apply MT to translate sentence from source tp target language

translate.google.com

Back translation

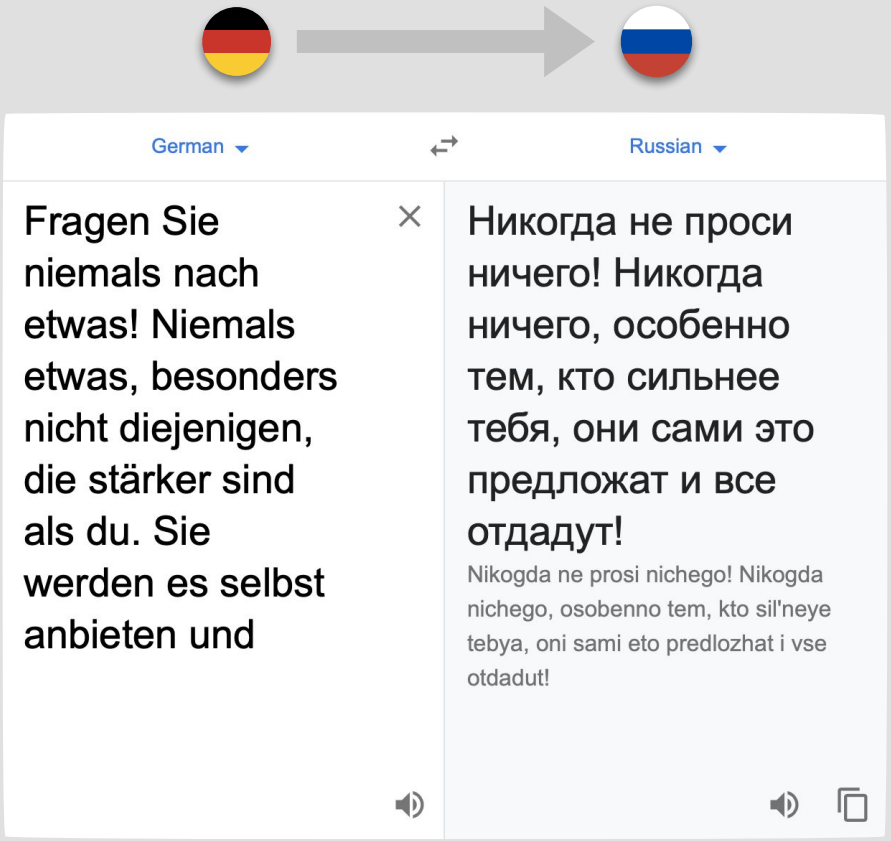
- Apply MT to translate sentence from source to target language



translate.google.com

Back translation

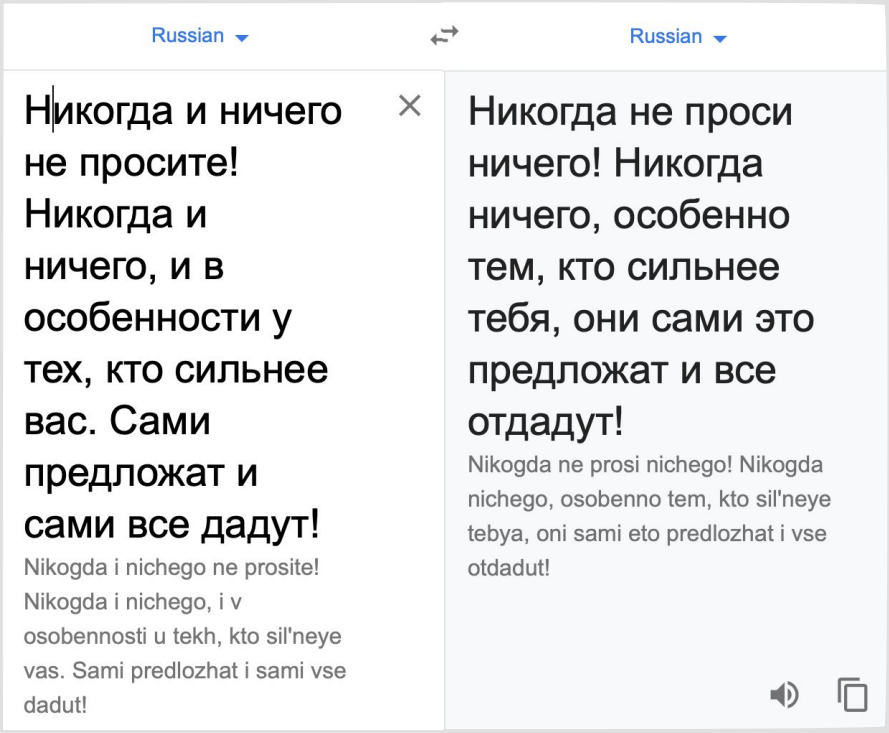
- Apply MT to translate sentence from source to target language
- … and back



translate.google.com

Back translation

- Apply MT to translate sentence from source to target language
- … and back
- Compare the original text and the resulting one after back translation



translate.google.com

BLEU (bilingual evaluation understudy)

- Compare sentences:
  - *Never ask for help!*
  - *Don’t ask for help!*

translate.google.com

BLEU (bilingual evaluation understudy)

- Compare sentences:
  - *Never ask for help!*
  - *Don't ask for help!*
- BLEU – algorithm for comparing an automatically translated sentence with the gold standard

translate.google.com

BLEU (bilingual evaluation understudy)

- Input - machine translation and gold standard

translate.google.com

BLEU (bilingual evaluation understudy)

- Input - machine translation and gold standard
- Count precision of word N-grams (N = 1 .. 4) $pr_i, i \in \{1,4\}$

translate.google.com

BLEU (bilingual evaluation understudy)

- Input - machine translation and gold standard
- Count precision of word N-grams (N = 1 .. 4) $pr_i, i \in \{1,4\}$

$$score = \sqrt[4]{\prod_{i=1}^{4} pr_i}$$

translate.google.com

BLEU (bilingual evaluation understudy)

- Input - machine translation and gold standard
- Count precision of word N-grams (N = 1 .. 4) $pr_i, i \in \{1,4\}$

$$score = \sqrt[4]{\prod_{i=1}^{4} pr_i}$$

- Brevity penalty: $$bp = min(1, \frac{mt-output-length}{gold-standard-length})$$

translate.google.com

BLEU (bilingual evaluation understudy)

- Input - machine translation and gold standard
- Count precision of word N-grams (N = 1 .. 4)

$$score = \sqrt[4]{\prod_{i=1}^{4} pr_i}$$

- Brevity penalty: $$bp = min(1, \frac{mt-output-length}{gold-standard-length})$$

$$BLEU = bp \times score$$

translate.google.com

BLEU (bilingual evaluation understudy)

Quote:

Remember that the most dangerous prison is the one in your head.

BLEU (bilingual evaluation understudy)

Quote:

Remember that the most dangerous prison is the one in your head.

Gold standard:

Помни, что самая опасная тюрьма – в твоей голове.

BLEU (bilingual evaluation understudy)

Quote:

Remember that the most dangerous prison is the one in your head.

Gold standard:

Помни, что самая опасная тюрьма – в твоей голове.

MT1:

Помните, что самая опасная тюрьма находится в вашей голове.

MT2:

Помни, что самая опасная тюрьма – это тюрьма в твоей голове.

BLEU (bilingual evaluation understudy)

Quote:

Remember that the most dangerous prison is the one in your head.

Gold standard:

Помни, что самая опасная тюрьма – в твоей голове.

MT1:

Помните, что самая опасная тюрьма находится в вашей голове.

MT2:

Помни, что самая опасная тюрьма – это тюрьма в твоей голове.

BLEU (bilingual evaluation understudy)

Quote:

Remember that the most dangerous prison is the one in your head.

Gold standard:

Помни, что самая опасная тюрьма – в твоей голове.

MT1:

Помните, что самая опасная тюрьма находится в вашей голове.

MT2:

Помни, что самая опасная тюрьма – это тюрьма в твоей голове.

BLEU (bilingual evaluation understudy)

Quote:

Remember that the most dangerous prison is the one in your head.

Gold standard:

Помни, что самая опасная тюрьма – в твоей голове.

MT1:

Помните, что самая опасная тюрьма находится в вашей голове.

MT2:

Помни, что самая опасная тюрьма – это тюрьма в твоей голове.

	MT1	MT2
Precision, n=1		
Precision, n=2		
Precision, n=3		
Precision, n=4		
Brevity penalty		
BLEU		

BLEU (bilingual evaluation understudy)

Quote:

Remember that the most dangerous prison is the one in your head.

Gold standard:

Помни, что самая опасная тюрьма – в твоей голове.

MT1:

Помните, что самая опасная тюрьма находится в вашей голове.

MT2:

Помни, что самая опасная тюрьма – это тюрьма в твоей голове.

	MT1	MT2
Precision, n=1	5/9	
Precision, n=2	3/8	
Precision, n=3	2/7	
Precision, n=4	1/6	
Brevity penalty		
BLEU		

BLEU (bilingual evaluation understudy)

Quote:

Remember that the most dangerous prison is the one in your head.

Gold standard:

Помни, что самая опасная тюрьма – в твоей голове.

MT1:

Помните, что самая опасная тюрьма находится в вашей голове.

MT2:

Помни, что самая опасная тюрьма – это тюрьма в твоей голове.

	MT1	MT2
Precision, n=1	5/9	
Precision, n=2	3/8	
Precision, n=3	2/7	
Precision, n=4	1/6	
Brevity penalty	1	
BLEU		

BLEU (bilingual evaluation understudy)

Quote:

Remember that the most dangerous prison is the one in your head.

Gold standard:

Помни, что самая опасная тюрьма – в твоей голове.

MT1:

Помните, что самая опасная тюрьма находится в вашей голове.

MT2:

Помни, что самая опасная тюрьма – это тюрьма в твоей голове.

	MT1	MT2
Precision, n=1	5/9	
Precision, n=2	3/8	
Precision, n=3	2/7	
Precision, n=4	1/6	
Brevity penalty	1	
BLEU	0.3303	

BLEU (bilingual evaluation understudy)

Quote:

Remember that the most dangerous prison is the one in your head.

Gold standard:

Помни, что самая опасная тюрьма – в твоей голове.

MT1:

Помните, что самая опасная тюрьма находится в вашей голове.

MT2:

Помни, что самая опасная тюрьма – это тюрьма в твоей голове.

	MT1	MT2
Precision, n=1	5/9	7/10
Precision, n=2	3/8	5/9
Precision, n=3	2/7	3/8
Precision, n=4	1/6	2/7
Brevity penalty	1	1
BLEU	0.3303	**0.6389**

WER (word error rate)

- The minimum number of operations required for transforming MT result into gold standard
- Operations: replacement, deletion, adding words

$$WER = \frac{\#\text{deletion} + \#\text{adding words} + \#\text{replacement}}{\#\text{ (gold standard)}}$$

translate.google.com

WER (word error rate)

Quote:

Remember that the most dangerous prison is the one in your head.

Gold standard:

Помни, что самая опасная тюрьма – в твоей голове.

MT1:

Помните, что самая опасная тюрьма находится в вашей голове.

MT2:

Помни, что самая опасная тюрьма – это тюрьма в твоей голове.

WER (word error rate)

Quote:

Remember that the most dangerous prison is the one in your head.

Gold standard:

Помни, что самая опасная тюрьма – в твоей голове.

MT1:

Помните, что самая опасная тюрьма находится в вашей голове.

MT2:

Помни, что самая опасная тюрьма – это тюрьма в твоей голове.

	MT1	MT2
# adding words		
# replacement		
# deletion		
WER		

WER (word error rate)

Quote:

Remember that the most dangerous prison is the one in your head.

Gold standard:

Помни, что самая опасная тюрьма – в твоей голове.

MT1:

~~Помните~~ помни, что самая опасная тюрьма находится в вашей голове.

MT2:

Помни, что самая опасная тюрьма – это тюрьма в твоей голове.

	MT1	MT2
# adding words		
# замен	1	
# deletion		
WER		

WER (word error rate)

Quote:

Remember that the most dangerous prison is the one in your head.

Gold standard:

Помни, что самая опасная тюрьма – в твоей голове.

MT1:

~~Помните~~ помни, что самая опасная тюрьма находится в вашей голове.

MT2:

Помни, что самая опасная тюрьма – это тюрьма в твоей голове.

	MT1	MT2
# adding words		
# replacement	1	
# deletion		
WER		

WER (word error rate)

Quote:

Remember that the most dangerous prison is the one in your head.

Gold standard:

Помни, что самая опасная тюрьма – в твоей голове.

MT1:

~~Помните~~ помни, что самая опасная тюрьма ~~находится~~ в вашей голове.

MT2:

Помни, что самая опасная тюрьма – это тюрьма в твоей голове.

	MT1	MT2
# adding words		
# replacement	1	
# deletion	1	
WER		

WER (word error rate)

Quote:

Remember that the most dangerous prison is the one in your head.

Gold standard:

Помни, что самая опасная тюрьма – в твоей голове.

MT1:

~~*Помните*~~ *помни, что самая опасная тюрьма* ~~*находится*~~ *в* ~~*вашей*~~ *твоей голове.*

MT2:

Помни, что самая опасная тюрьма – это тюрьма в твоей голове.

	MT1	MT2
# adding words		
# replacement	2	
# deletion	1	
WER		

WER (word error rate)

Quote:

Remember that the most dangerous prison is the one in your head.

Gold standard:

Помни, что самая опасная тюрьма – в твоей голове.

MT1:

~~*Помните*~~ *помни, что самая опасная тюрьма* ~~*находится*~~ *в* ~~*вашей*~~ *твоей голове.*

MT2:

Помни, что самая опасная тюрьма – это тюрьма в твоей голове.

	MT1	MT2
# adding words	0	
# replacement	2	
# deletion	1	
WER		

WER (word error rate)

Quote:

Remember that the most dangerous prison is the one in your head.

Gold standard:

Помни, что самая опасная тюрьма – в твоей голове.

MT1:

~~*Помните*~~ *помни, что самая опасная тюрьма* ~~*находится*~~ *в* ~~*вашей*~~ *твоей голове.*

MT2:

Помни, что самая опасная тюрьма – это тюрьма в твоей голове.

	MT1	MT2
# adding words	0	
# replacement	2	
# deletion	1	
WER	0.33	

WER (word error rate)

Quote:

Remember that the most dangerous prison is the one in your head.

Gold standard:

Помни, что самая опасная тюрьма – в твоей голове.

MT1:

~~Помните~~ помни, что самая опасная тюрьма ~~находится~~ в ~~вашей~~ твоей голове.

MT2:

Помни, что самая опасная тюрьма – ~~это тюрьма~~ в твоей голове.

	MT1	MT2
# adding words	0	
# replacement	2	
# deletion	1	
WER	0.33	

WER (word error rate)

Quote:

Remember that the most dangerous prison is the one in your head.

Gold standard:

Помни, что самая опасная тюрьма – в твоей голове.

MT1:

~~*Помните*~~ *помни, что самая опасная тюрьма* ~~*находится*~~ *в* ~~*вашей*~~ *твоей голове.*

MT2:

Помни, что самая опасная тюрьма – ~~*это тюрьма*~~ *в твоей голове.*

	MT1	MT2
# adding words	0	0
# replacement	2	0
# deletion	1	2
WER	0.33	**0.2**

WER and BLEU

Advantages

- Simple and quick
- Correlate with expert review

Disadvantages

- Operate with short fragments
- Can’t estimate genre specifics
- The ratings are relative, difficult to compare with each other
- An important technical problem: estimates are not differentiable