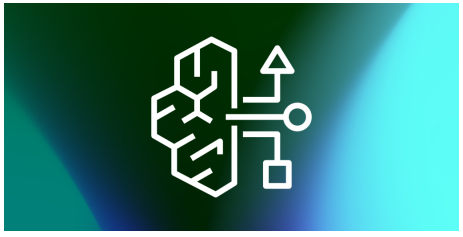
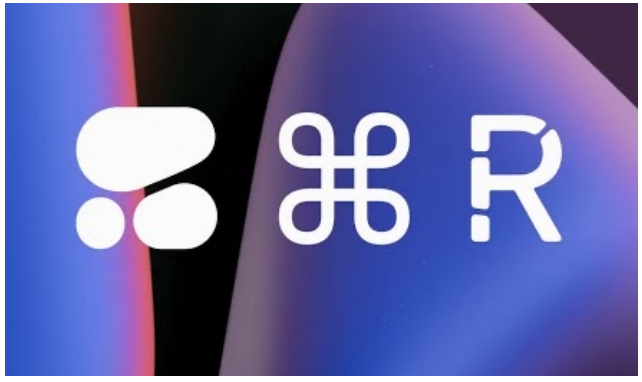


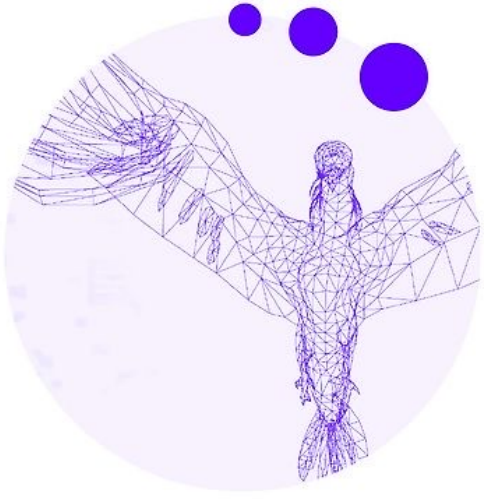
Claude

a BigScience initiative

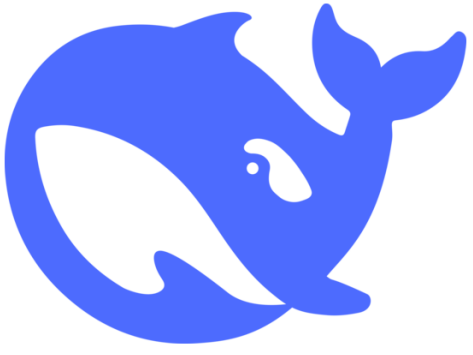
BLOOM

176B params · 59 languages · Open-access





Gemini



Grok

LUCIE

VALDOM - NLP 2 LLM

LARGE LANGUAGE MODELS

JOSEBA DALMAU

LLM Comparison

Task: search online for a common set of criteria to compare LLMs against each other.

LLM Comparison

Homework: Pick a LLM from the list and research about it online in order to fill in the comparison table.

LLM NEWS

Gradient Updates

How has DeepSeek improved the Transformer architecture?

Published

Jan 17, 2025

Authors

Ege Erdil

Share

Twitter LinkedIn

DeepSeek has recently released DeepSeek v3, which is currently state-of-the-art in benchmark performance among open-weight models, alongside a technical report describing in some detail the training of the model. Impressively, they've achieved this SOTA performance by only using 2.8 million H800 hours of training hardware time—equivalent to about 4e24 FLOP if we assume 40% MFU. This is about ten times less training compute than the similarly performing Llama 3.1 405B.

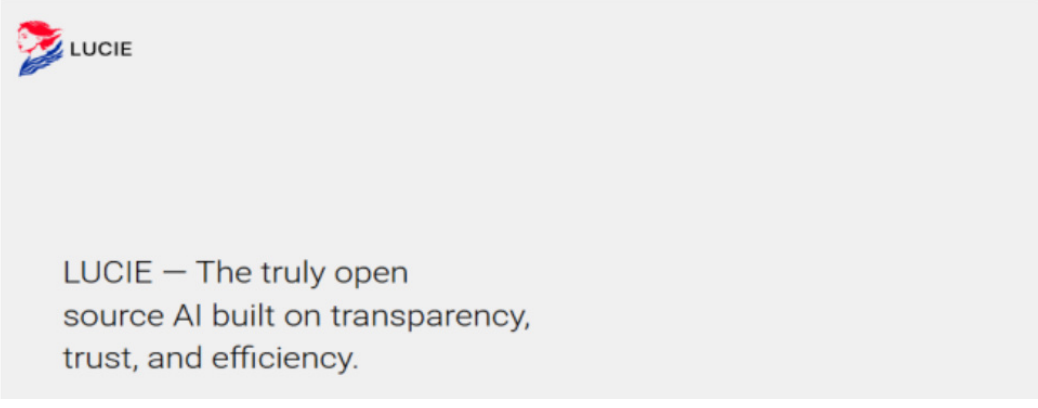
In this issue, I'll cover some of the important architectural improvements that DeepSeek highlight in their report and why we should expect them to result in better performance compared to a vanilla Transformer. The full technical report contains plenty of non-architectural details as well, and I strongly recommend reading it if you want to get a better idea of the

LLM News

Face aux erreurs, Linagora suspend temporairement son assistant GenAI Lucie

Jacques Cheminat, publié le 27 Janvier 2025

Lancé en test la semaine dernière, l'assistant GenAI nommé Lucie a été suspendu au début du week-end. Plusieurs réponses erronées ont provoqué un déferlement de critiques sur l'initiative portée par Linagora. Ce dernier a reconnu une erreur de communication et plaide un travail de recherche en phase initiale.



SUIVRE TOUTE L'ACTUALITÉ

Newsletter

Recevez notre newsletter comme plus de 50 000 professionnels de l'IT!

JE M'ABONNE

GC A **télécharger** sur guidescomparatifs.com

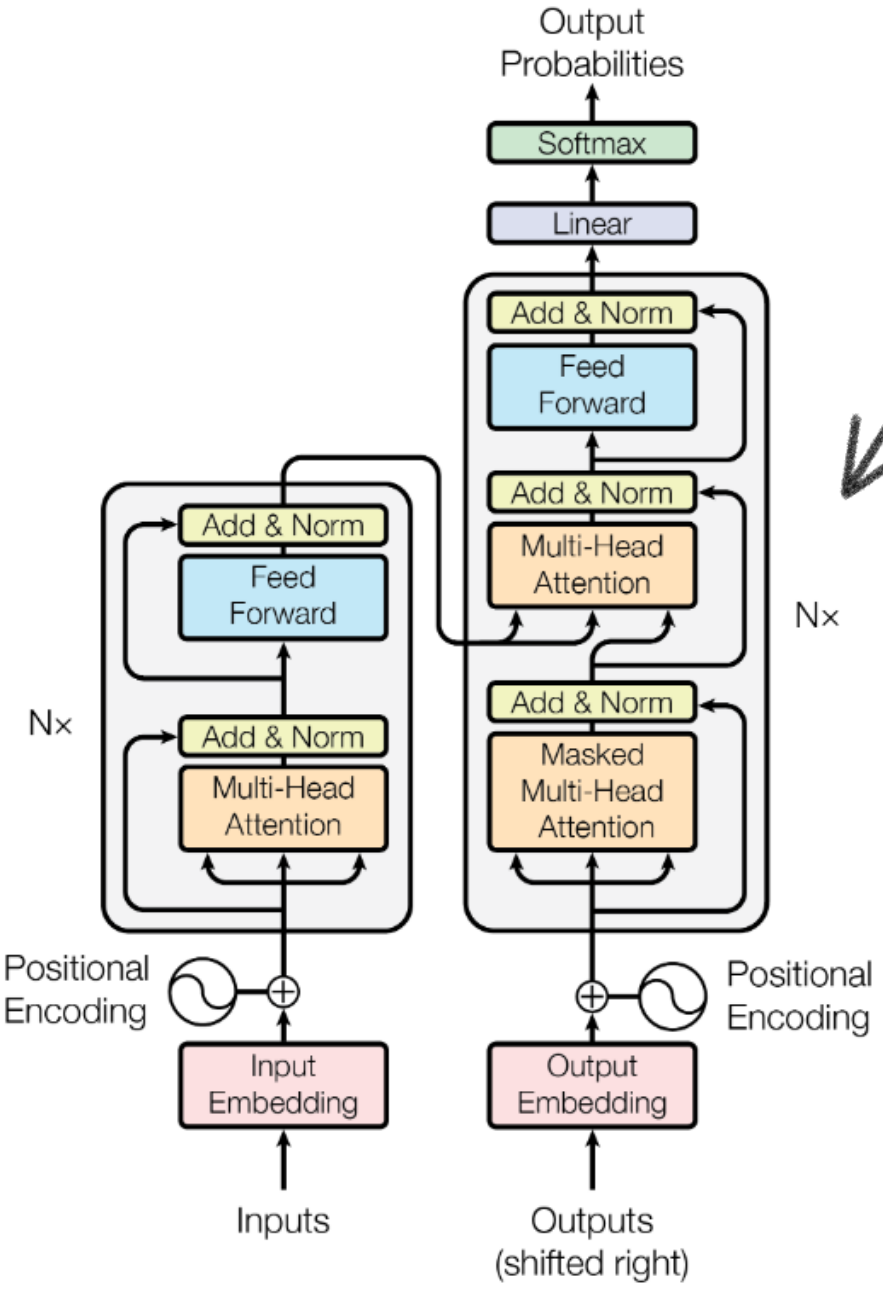
ERP - 454 critères

40 pages de **cahier des charges** pour préparer votre projet ERP [PDF] [EXCEL]

OBJECTIVE

Master the Transformer architecture for various different NLP tasks

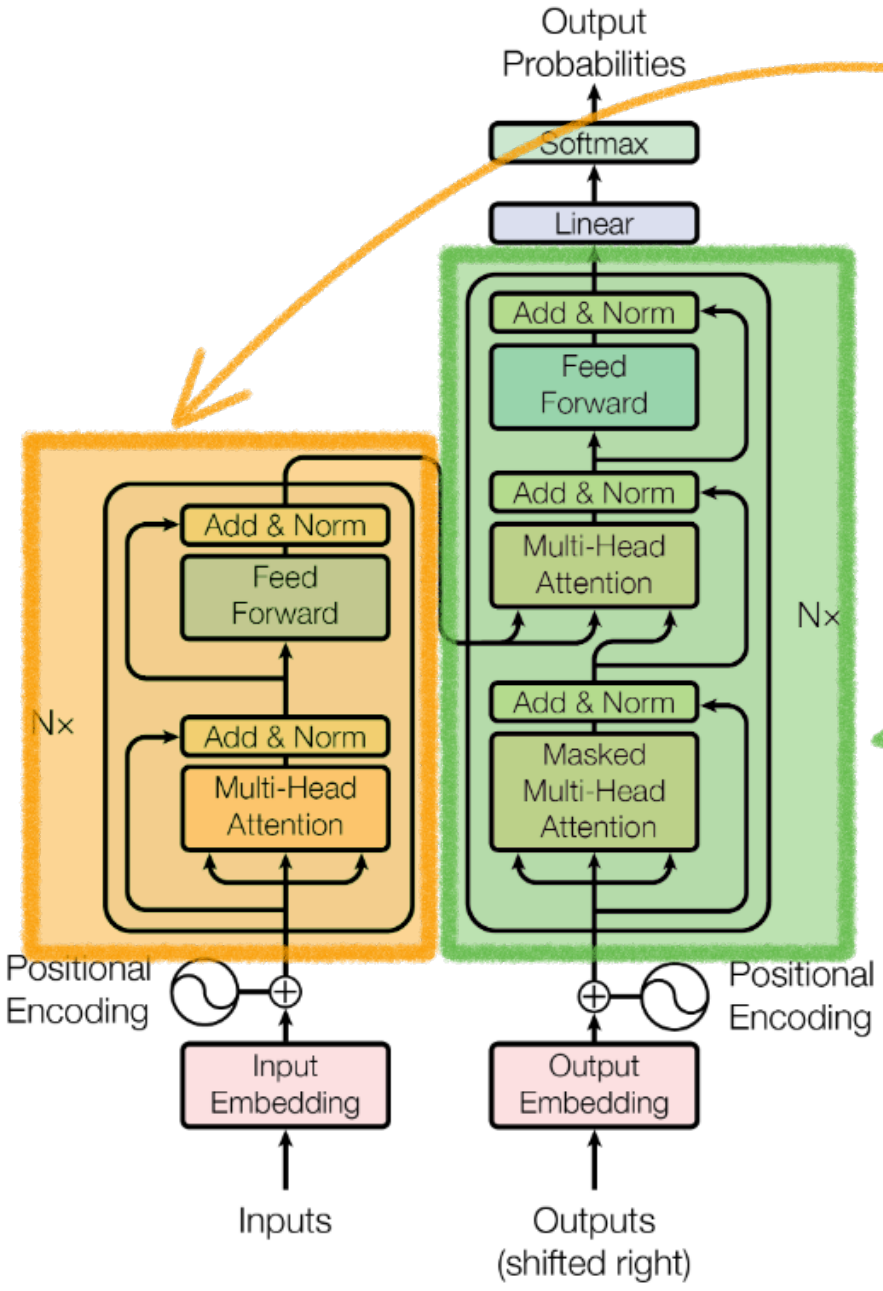
TRANSFORMERS



Main figure of the transformer architecture from the original paper "Attention is all you need"

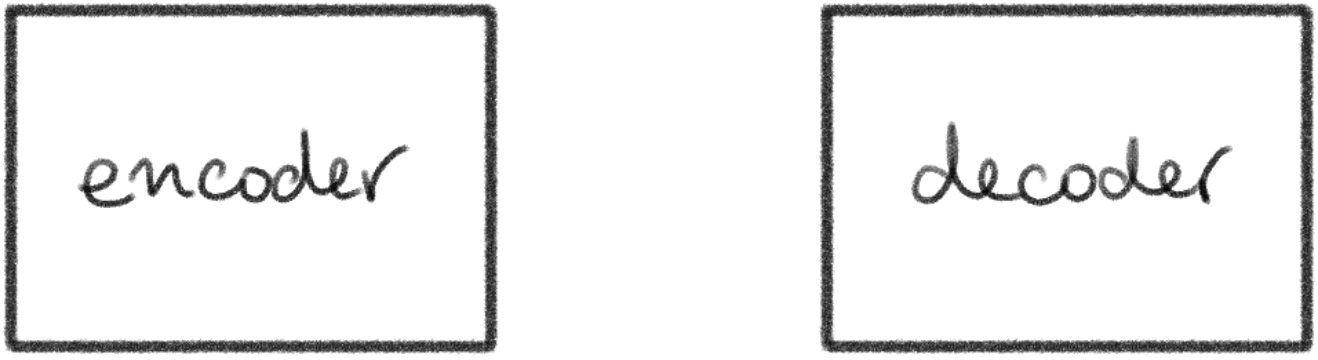
Vaswani et al. 2017

Encoder - Decoder Architecture



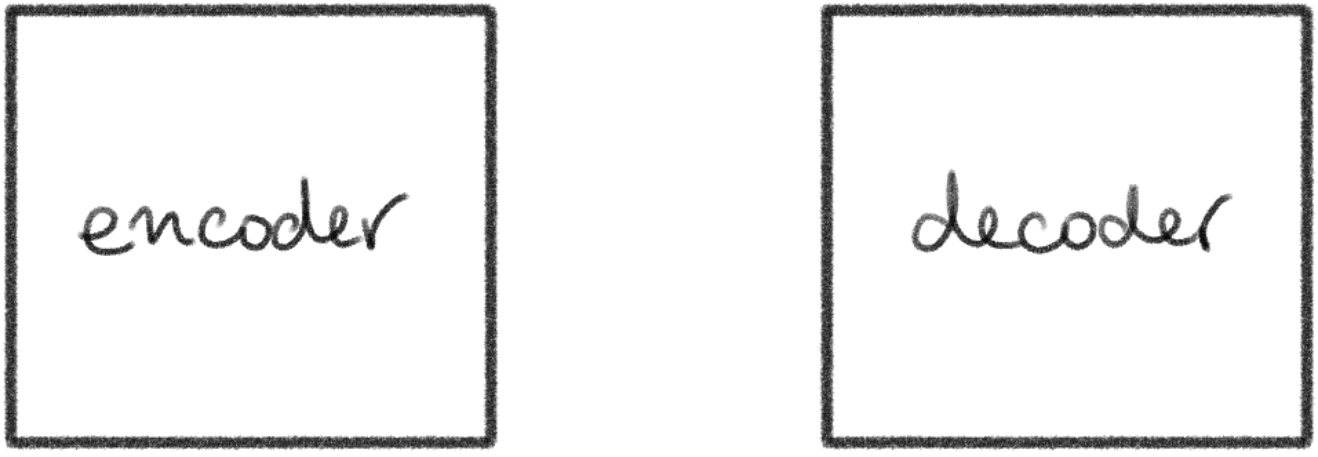
INFERENCE

Je suis enseignant



I am a teacher

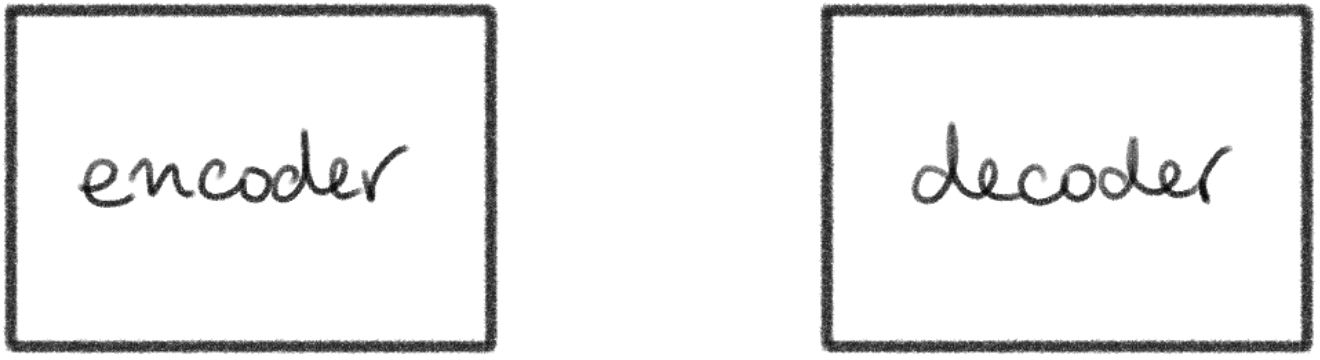
Inference



"input sentence"

I am a teacher

Inference

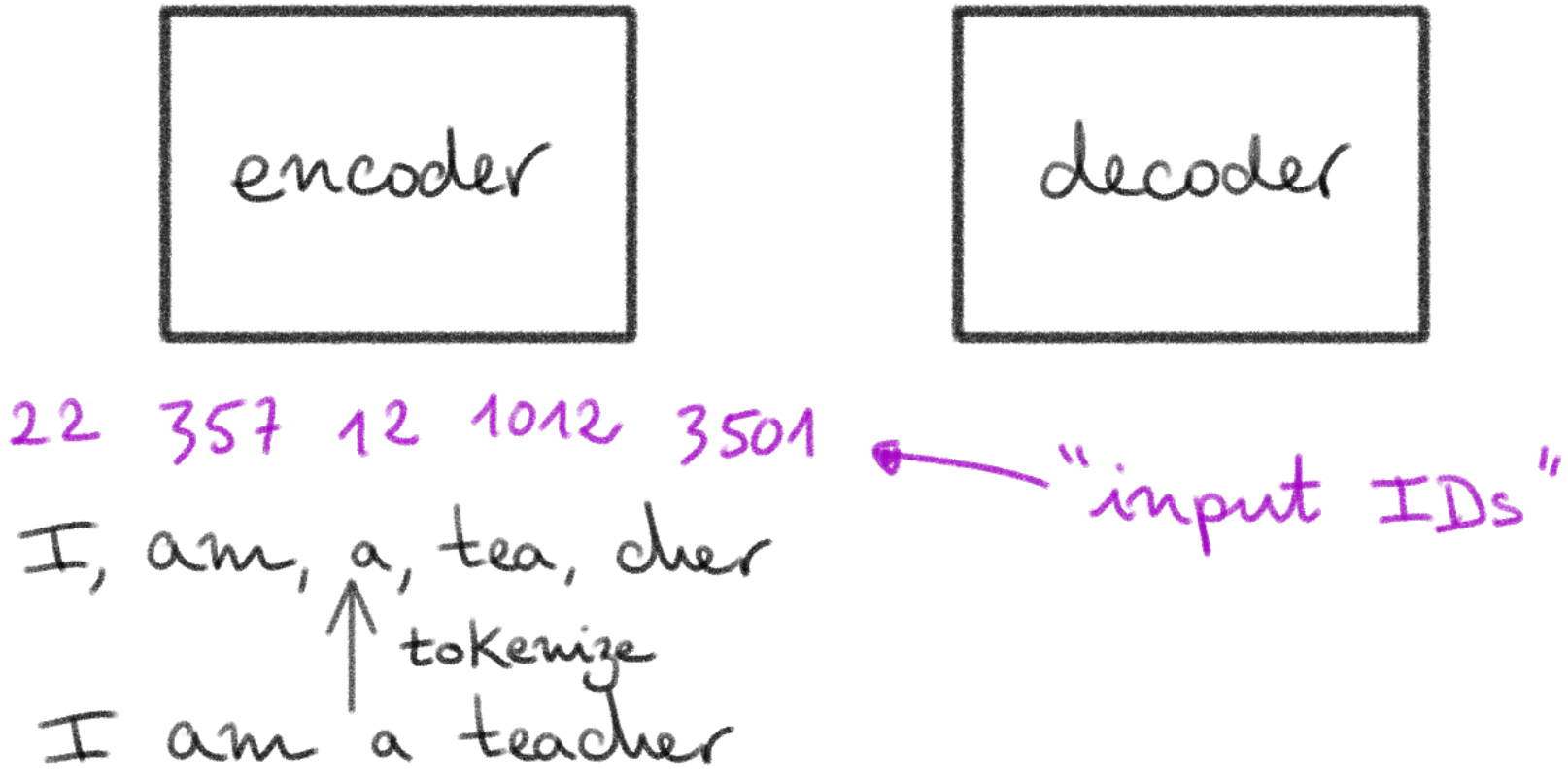


I, am, a, tea, cher ← "Input tokens"

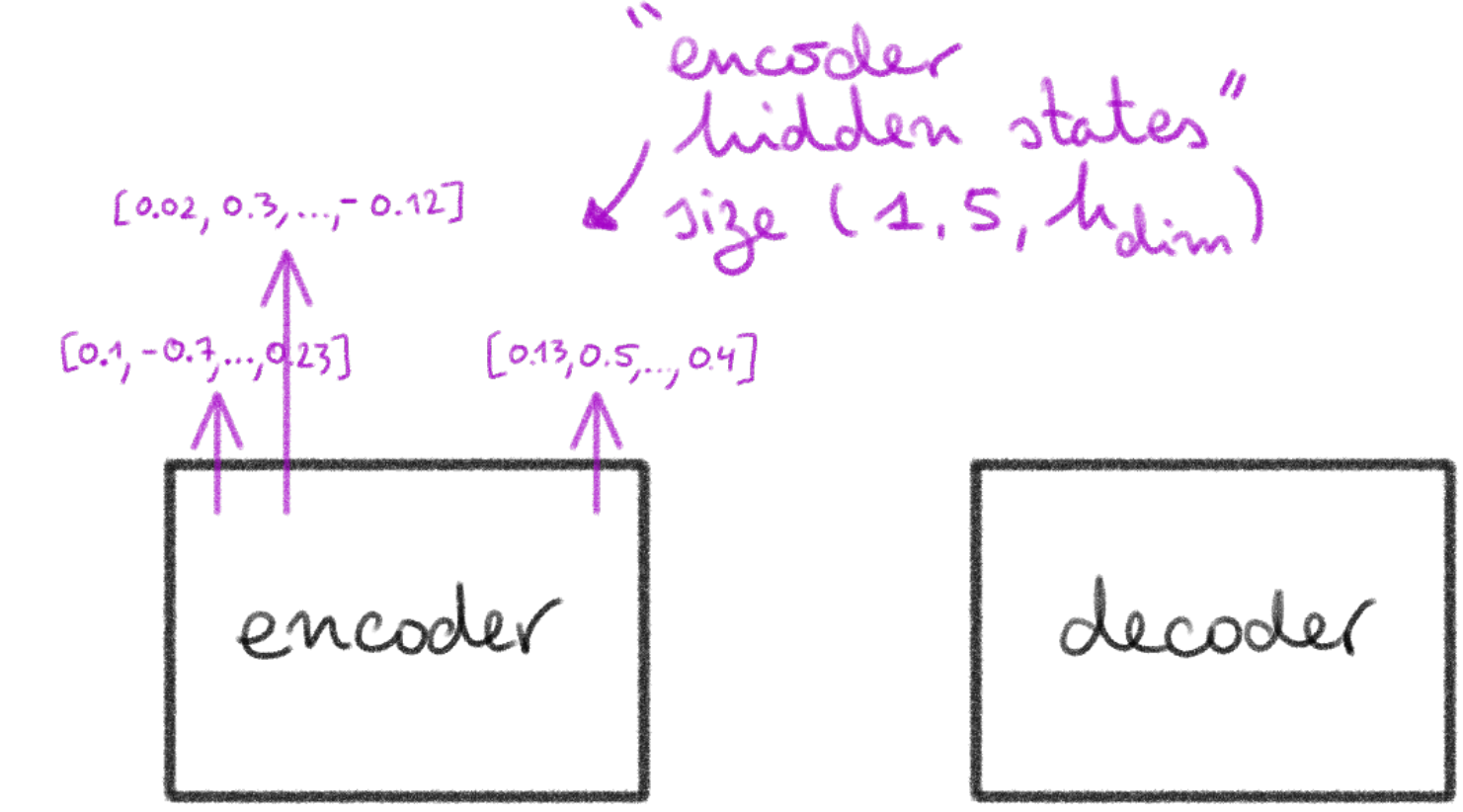
tokenize

I am a teacher

Inference


encoder
decoder
22 357 12 1012 3501
"input IDs"
I, am, a, tea, cher
tokenize
I am a teacher

INFERENCE



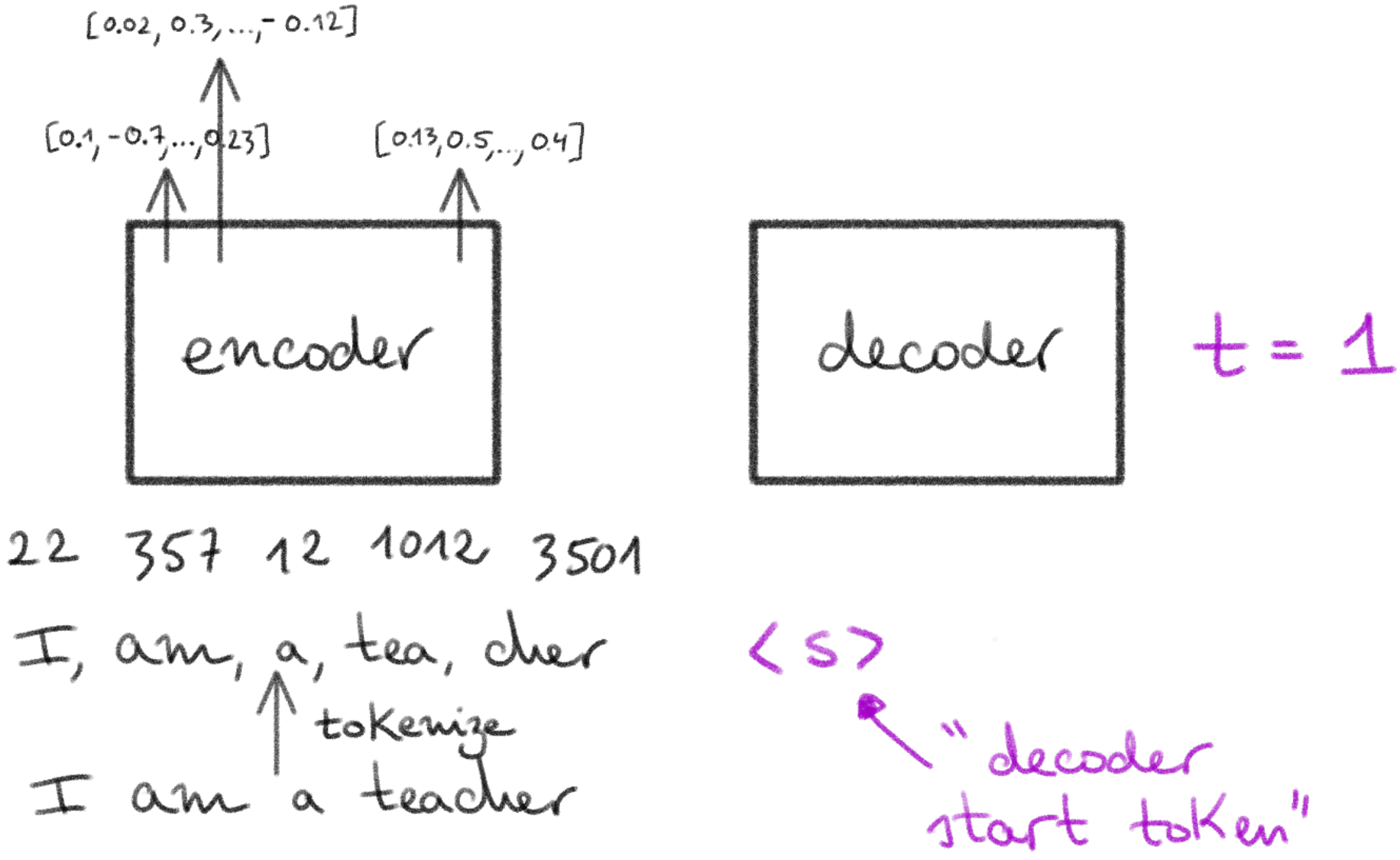
22 357 12 1012 3501

I, am, a, tea, cher

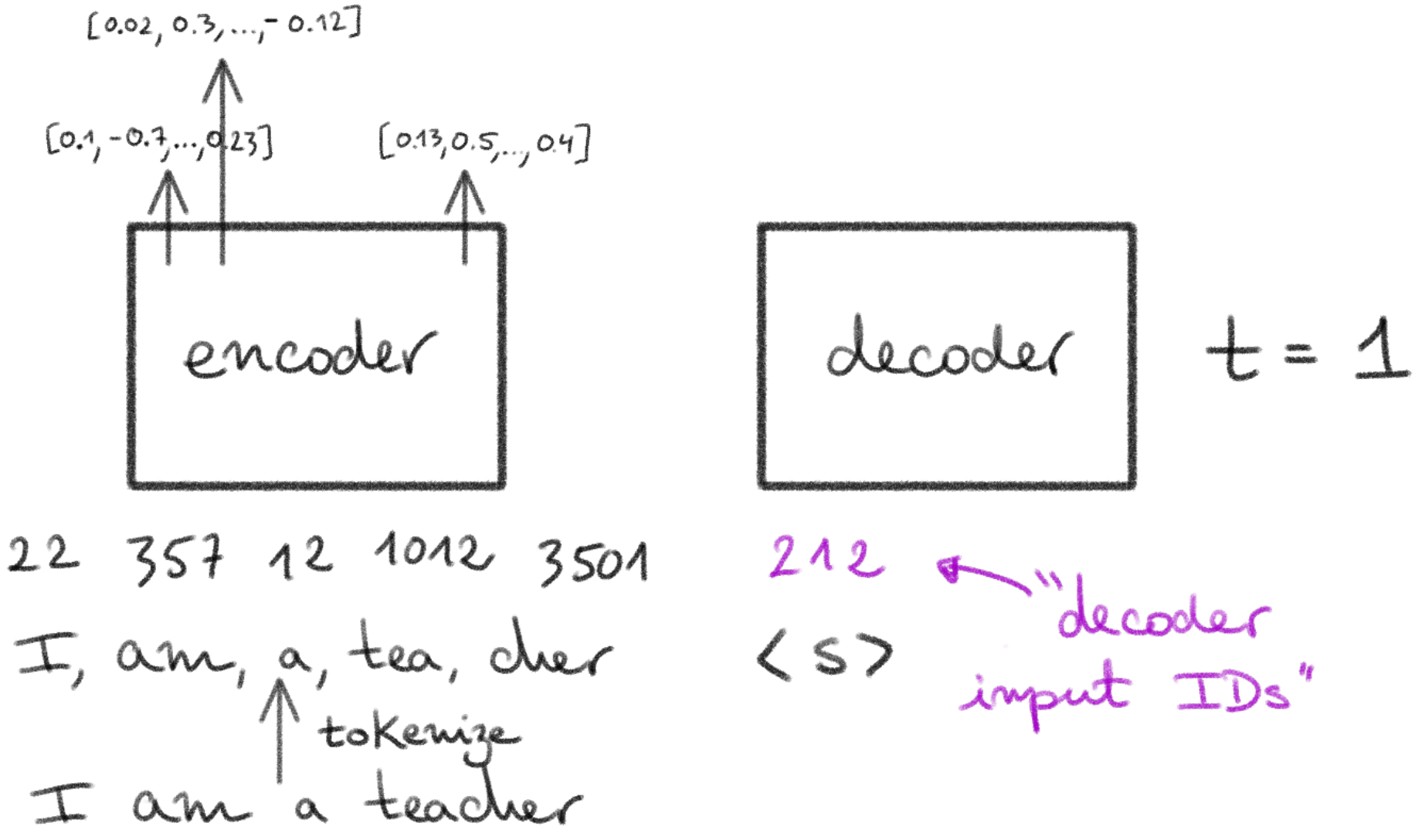
tokenize

I am a teacher

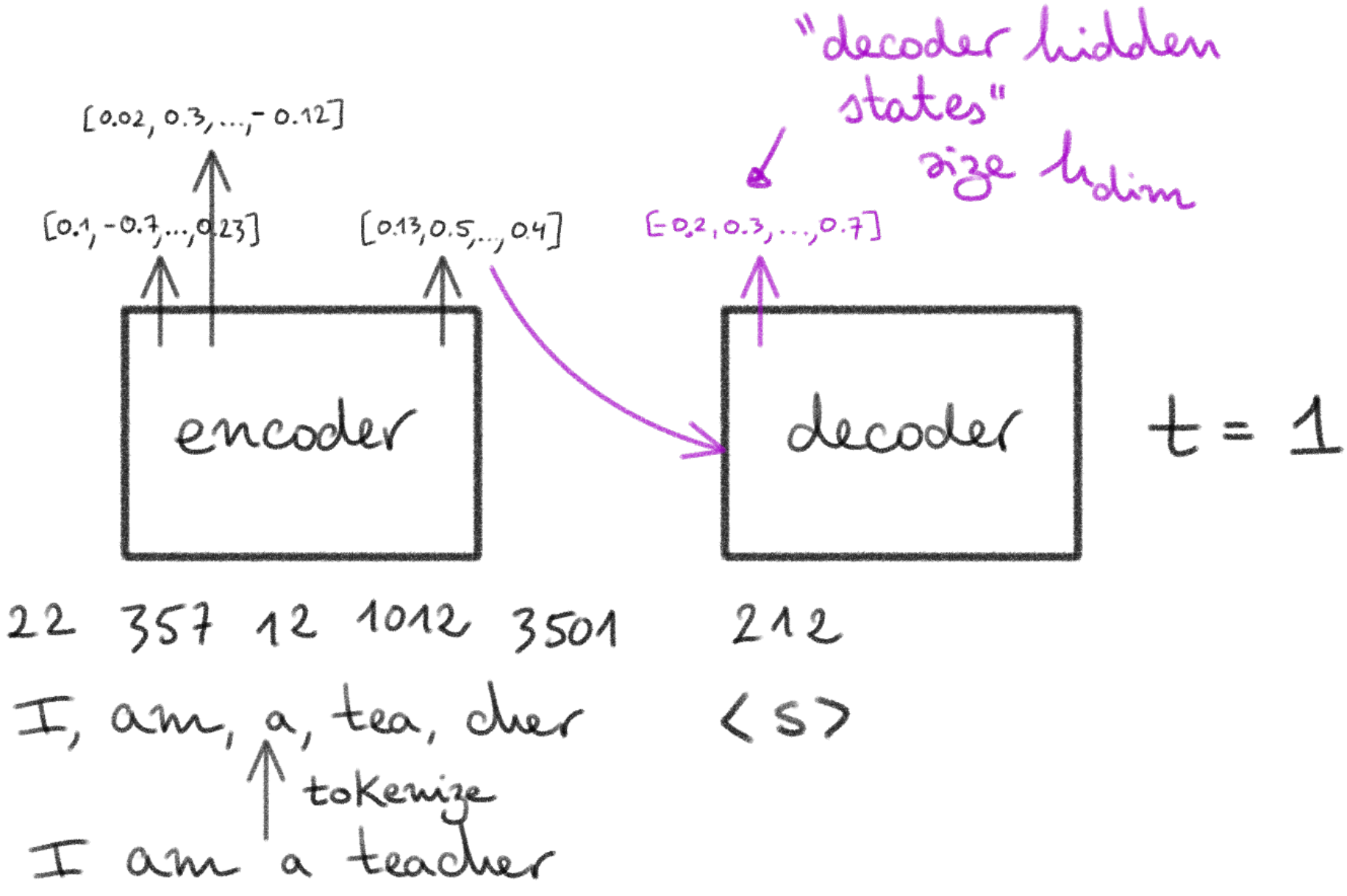
Inference


[0.02, 0.3, ..., -0.12]
[0.1, -0.7, .., 0.23]
[0.13, 0.5, .., 0.4]
encoder
decoder
t = 1
22 357 12 1012 3501
I, am, a, tea, cher
tokenize
I am a teacher
<s>
" decoder start token"

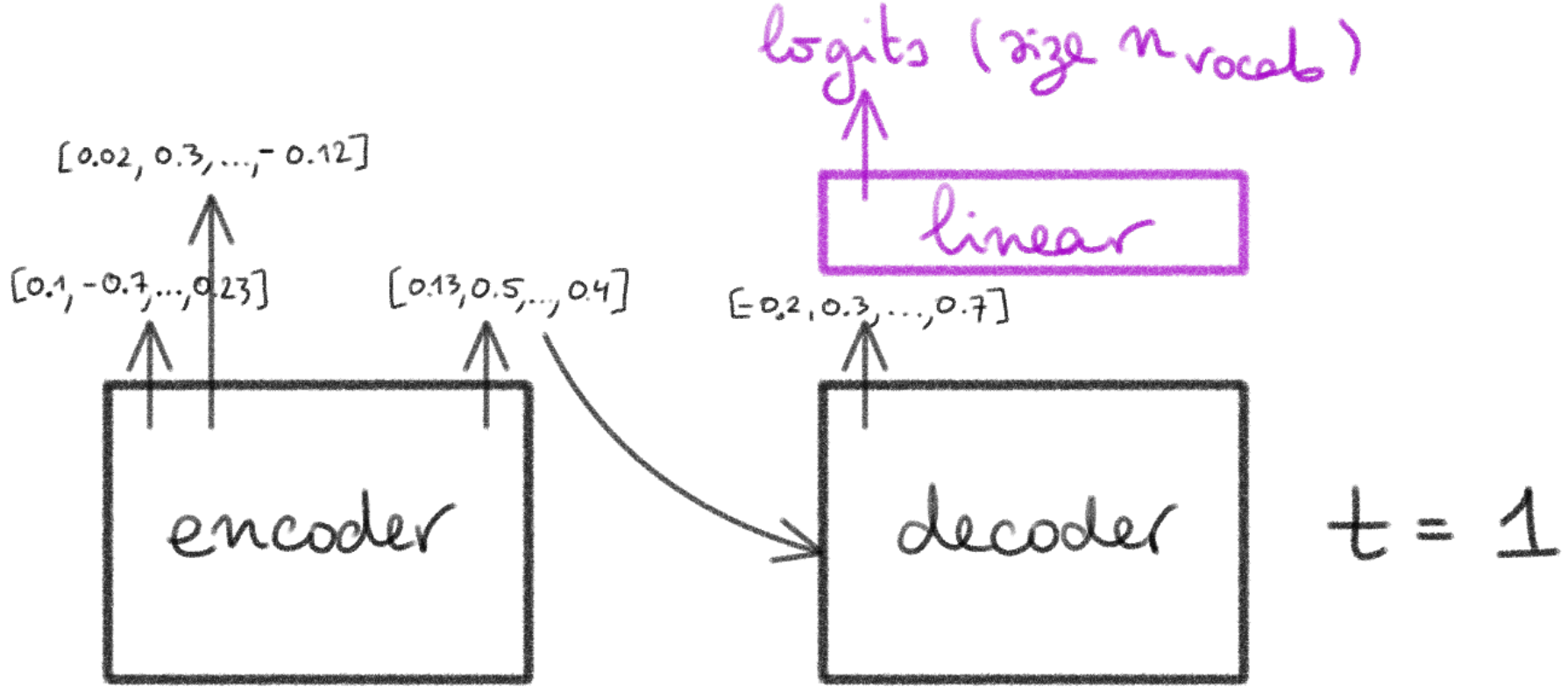
Inference


[0.02, 0.3, ..., -0.12]
[0.1, -0.7, .., 0.23]
[0.13, 0.5, .., 0.4]
encoder
decoder
t = 1
22 357 12 1012 3501
I, am, a, tea, cher
tokenize
I am a teacher
212
<s>
"decoder input IDs"

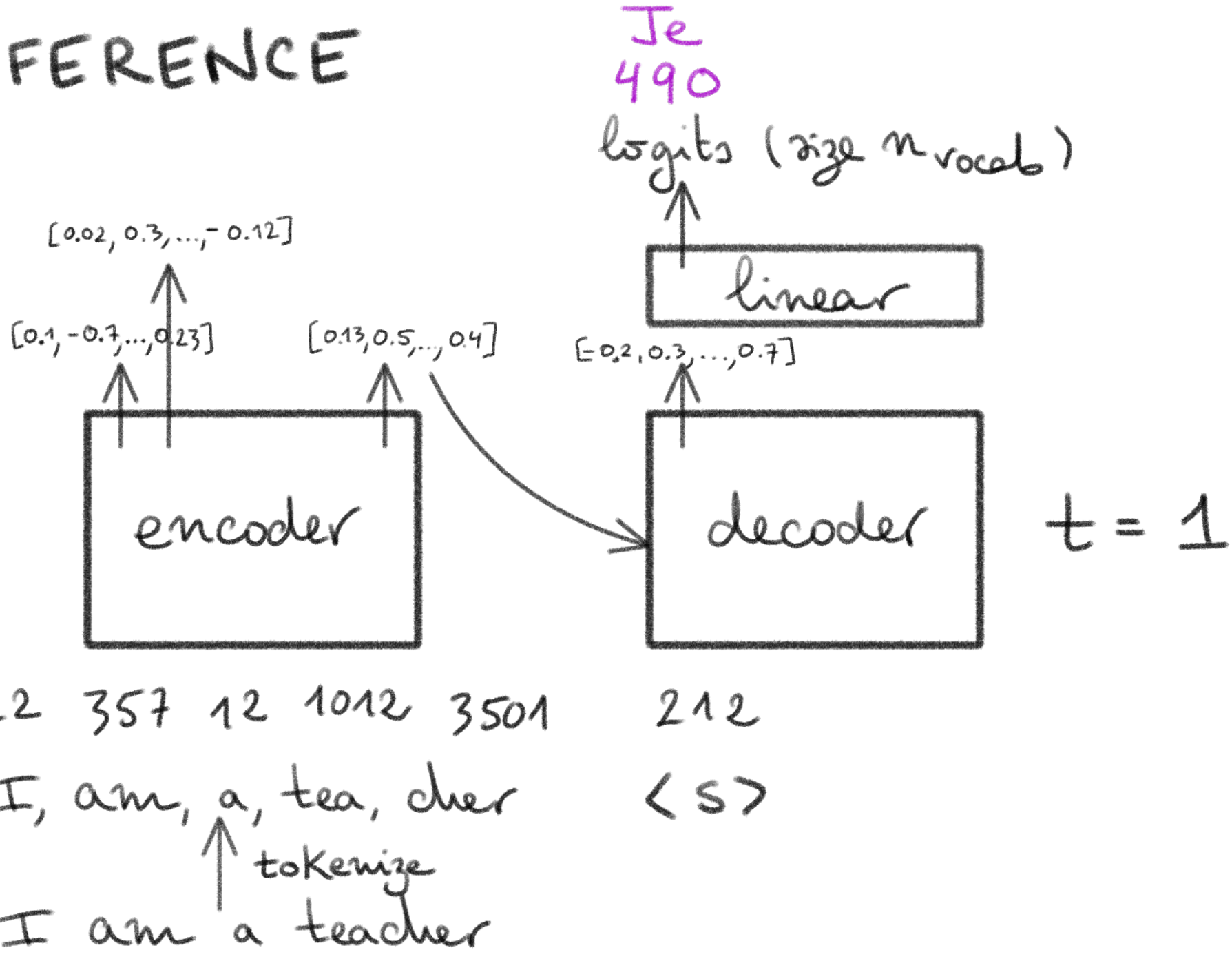
Inference


"decoder hidden states"
size hdim
[0.02, 0.3, ..., -0.12]
[0.1, -0.7, ..., 0.23]
[0.13, 0.5, ..., 0.4]
[-0.2, 0.3, ..., 0.7]
encoder
decoder
t = 1
22 357 12 1012 3501
212
I, am, a, tea, cher
<s>
tokenize
I am a teacher

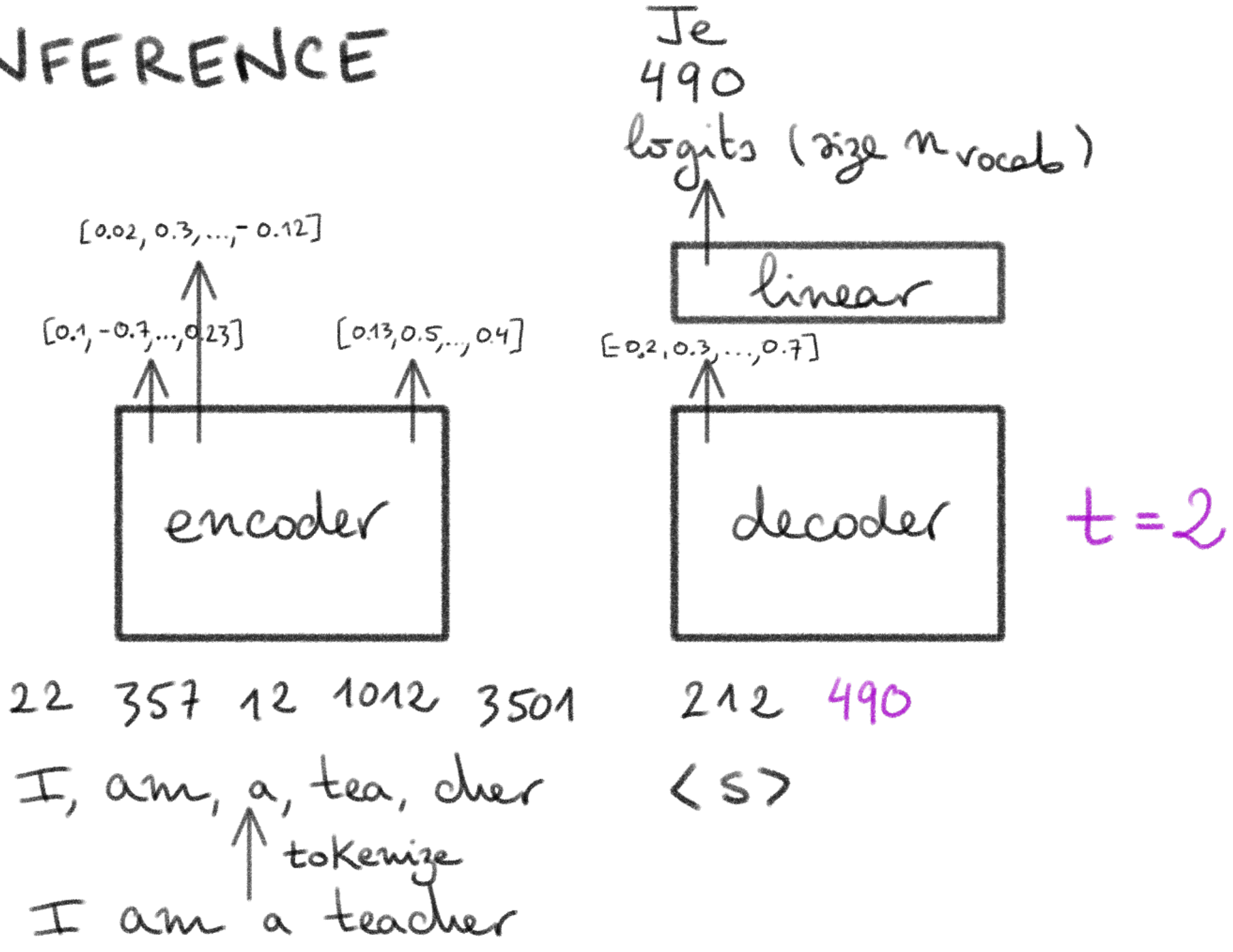
Inference


logits (size n_vocab)
linear
[0.02, 0.3, ..., -0.12]
[0.1, -0.7, .., 0.23]
[0.13, 0.5, .., 0.4]
[-0.2, 0.3, ..., 0.7]
encoder
decoder
t = 1
22 357 12 1012 3501
212
I, am, a, tea, cher
<s>
tokenize
I am a teacher

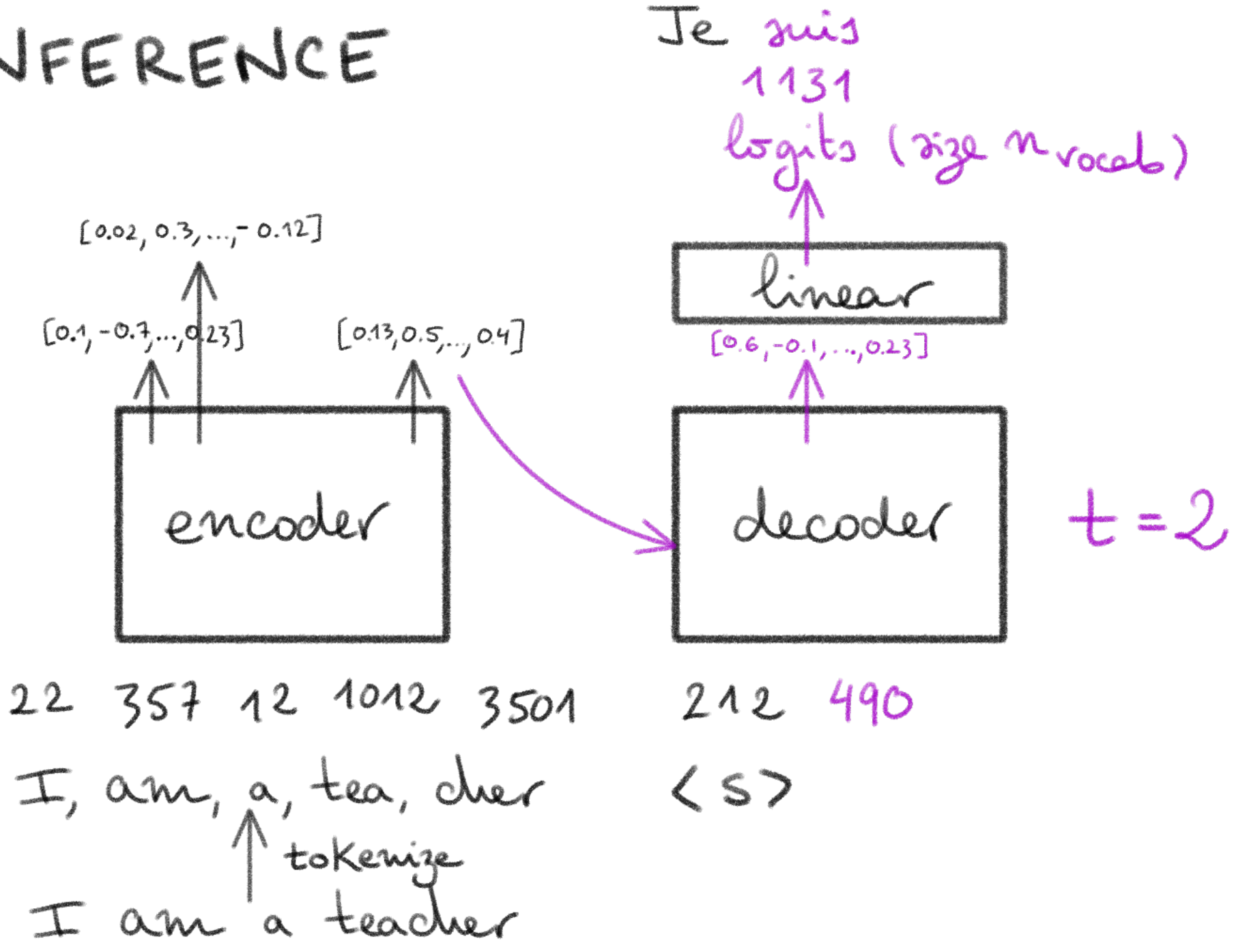
Inference


Je
490
logits (size n_vocab)
linear
[0.02, 0.3, ..., -0.12]
[0.1, -0.7, .., 0.23]
[0.13, 0.5, .., 0.4]
[-0.2, 0.3, ..., 0.7]
encoder
decoder
t = 1
22 357 12 1012 3501
212
I, am, a, tea, cher
<s>
tokenize
I am a teacher

INFERENCE


Je
490
logits (size n_vocab)
linear
[0.02, 0.3, ..., -0.12]
[0.1, -0.7, .., 0.23]
[0.13, 0.5, .., 0.4]
[-0.2, 0.3, ..., 0.7]
encoder
decoder
t = 2
22 357 12 1012 3501
212 490
I, am, a, tea, cher
<s>
tokenize
I am a teacher

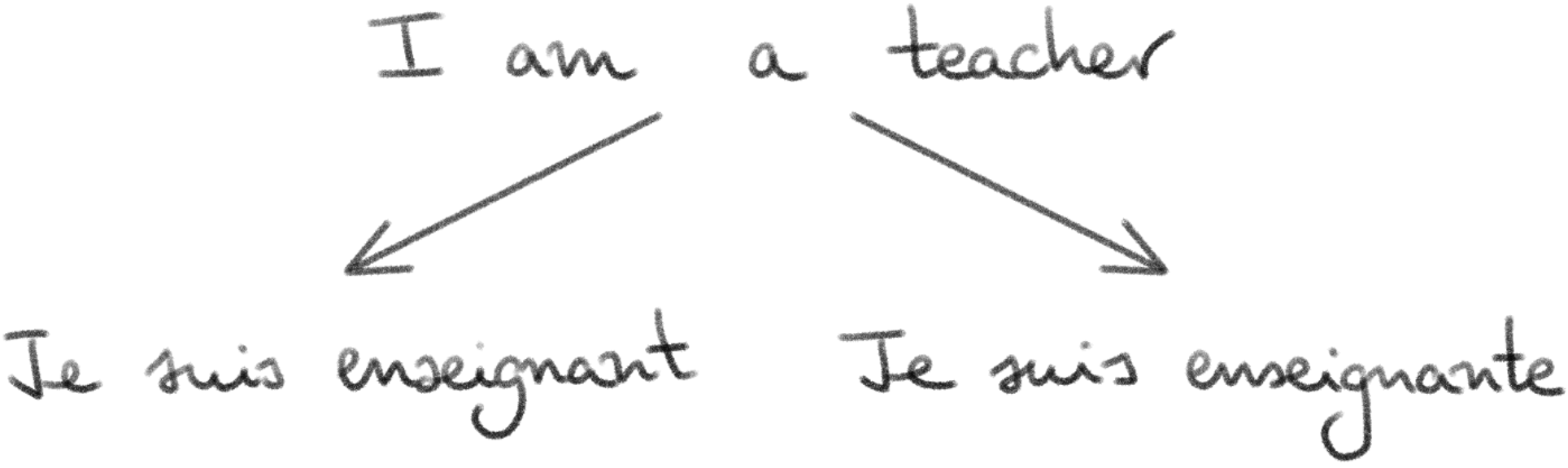
INFERENCE


Je suis
1131
logits (size n_vocab)
[0.02, 0.3, ..., -0.12]
[0.1, -0.7, .., 0.23]
[0.13, 0.5, .., 0.4]
linear
[0.6, -0.1, .., 0.23]
encoder
decoder
t = 2
22 357 12 1012 3501
212 490
I, am, a, tea, cher
<s>
tokenize
I am a teacher

Inference

We iterate through the decoding process until we hit the "end of sequence" (<EOS>) token!

Inference



⇒ We can randomize the output using softmax instead of argmax of logits!

Decoding Strategies

LLM gives:

$$p(x_t \mid x_{<t}) = \text{softmax}(z) = \left(\frac{e^{z_1}}{\sum_{k=1}^{n_v} e^{z_k}}, \dots, \frac{e^{z_{n_v}}}{\sum_{k=1}^{n_v} e^{z_k}} \right)$$

sample = FALSE $\Rightarrow$ argmax(z)

sample = TRUE $\Rightarrow \mathbb{P}(h) = \frac{e^{z_h}}{\sum_{k=1}^{n_v} e^{z_k}}$

Decoding Strategies

Adding temperature:

$$p(x_t \mid x_{<t}) = \text{softmax}\left(\frac{z}{T}\right)$$

$T < 1 \Rightarrow$ sharper distribution $\xrightarrow{T \to 0} \left(\mathbb{1}_{k = \text{argmax}(z)}\right)_{k=1}^{n_v}$

$T = 1 \Rightarrow$ original distribution

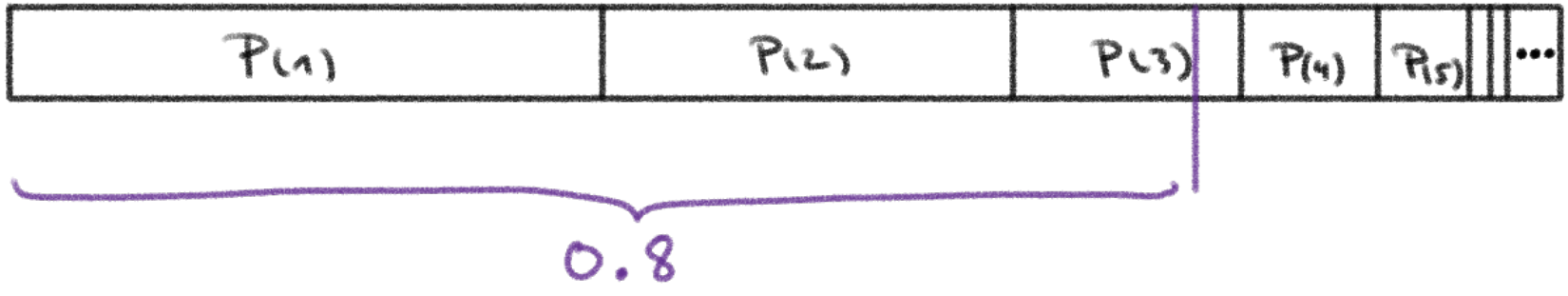
$T > 1 \Rightarrow$ flatter distribution $\xrightarrow{T \to \infty} \left(\frac{1}{n_v}, \dots, \frac{1}{n_v}\right)$

Decoding Strategies

Adding top-p sampling ($p = 0.8$)

$$P(x_t \mid x_{<t}) = (p_1, \cdots, p_{n_v})$$

Rank $P_{(1)} > P_{(2)} > P_{(3)} > \cdots > P_{(n_v)}$



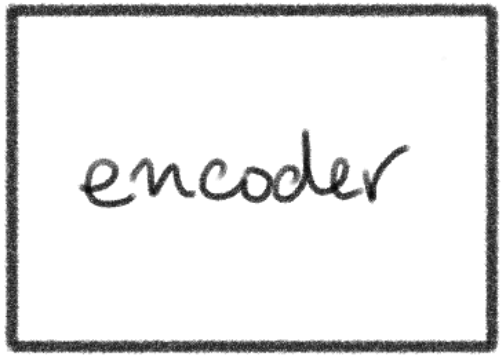
$\Rightarrow$ sample only between $K=(1)$, $K=(2)$, $K=(3)$

TRAINING

labels

Je, suis, ensei, gnant

490 1131 3212 2757



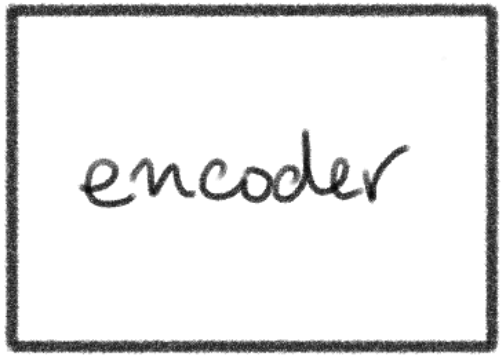
22 357 12 1012 3501

I, am, a, tea, cher

TRAINING

Je, suis, ensei, gnant

490 1131 3212 2757



22 357 12 1012 3501

I, am, a, tea, cher

212 490 1131 3212 2757

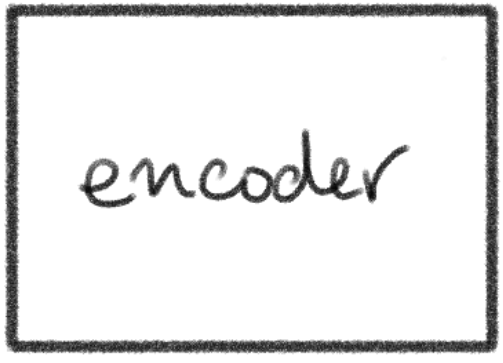
<s>

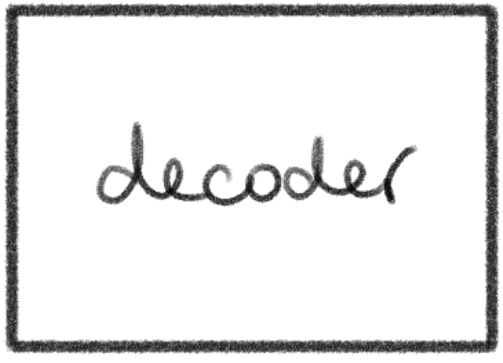
TRAINING

Je, suis, ensei, gnant <EOS>

490 1131 3212 2757 5000

l_1 l_2 l_3 l_4 l_5





22 357 12 1012 3501

I, am, a, tea, cher

212 490 1131 3212 2757

<S>

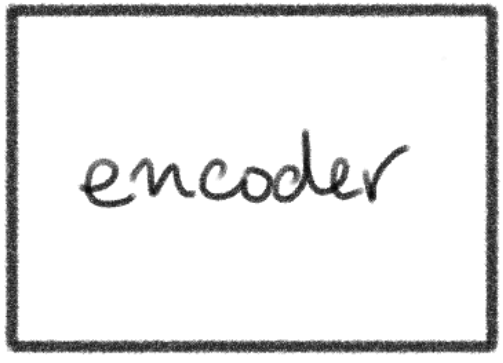
TRAINING

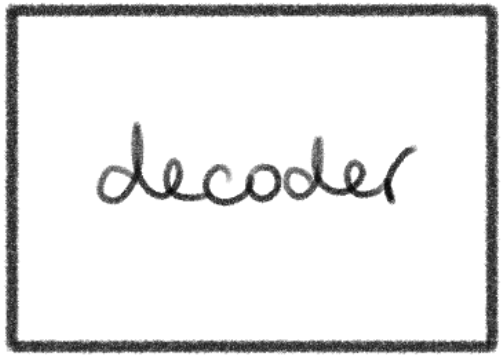
compute
cross-entropy loss
& back-propagation

Je, suis, ensei, gnant <EOS>

490 1131 3212 2757 5000

ℓ_1 ℓ_2 ℓ_3 ℓ_4 ℓ_5





22 357 12 1012 3501

I, am, a, tea, cher

212 490 1131 3212 2757

<S>

Cross-Entropy Loss

Used for multi-class classification

$$L(y, \hat{y}) := -\sum_{k=1}^{C} y_k \log(\hat{y}_k)$$

C: nb. of classes

y: ground truth (one-hot)

$\hat{y}$: prediction probabilities

Transformers apply CE loss tokenwise!

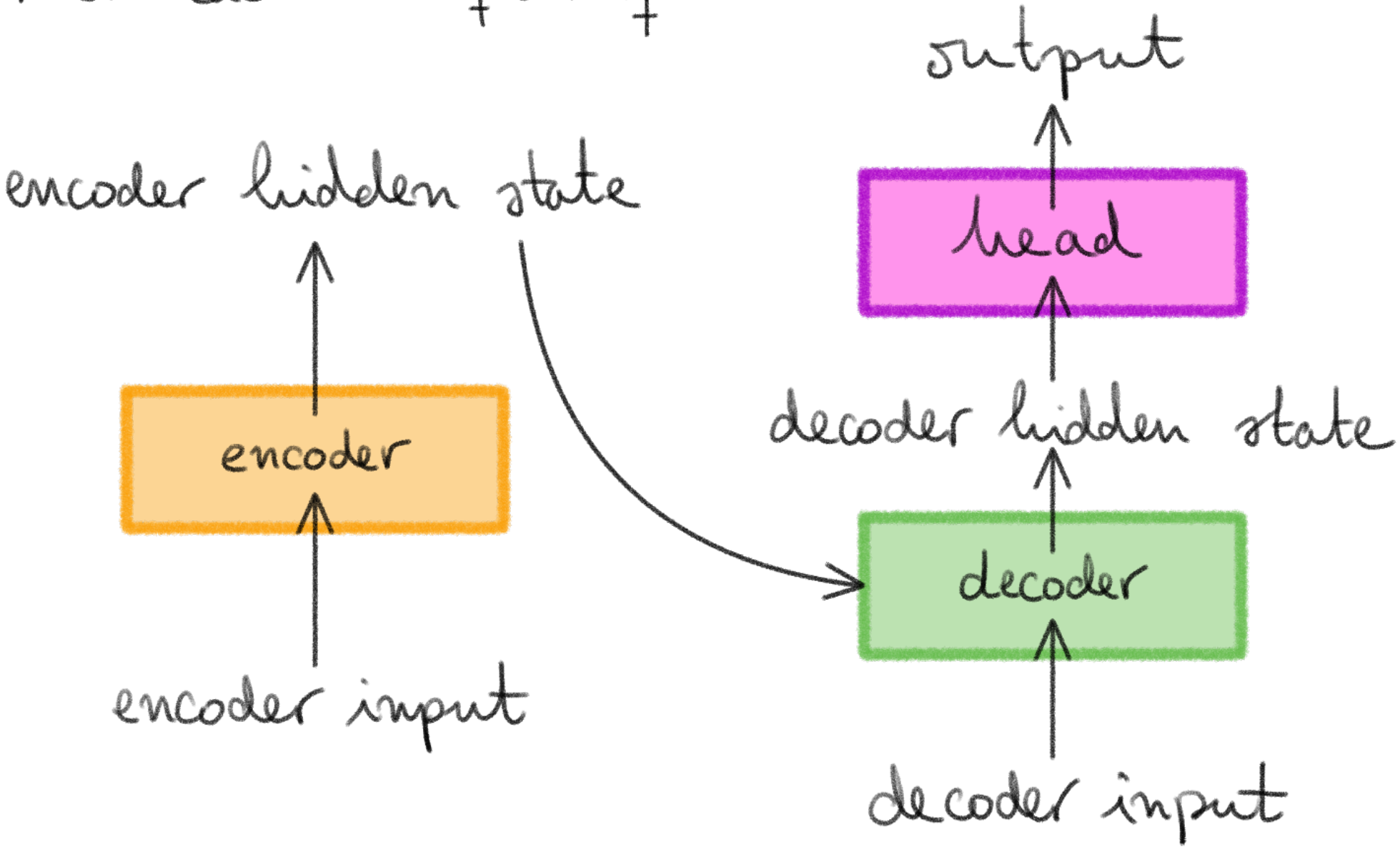
EXERCISE

Given a 3-class classification problem, and assuming a data point (x, y) where the one-hot encoded label y is $(1, 0, 0)$, $\hat{z} = \hat{f}(x) \in \mathbb{R}^3$ logits and $\hat{y} = \text{softmax}(\hat{z})$ i.e. $\hat{y}_k = \frac{e^{\hat{z}_k}}{e^{\hat{z}_1} + e^{\hat{z}_2} + e^{\hat{z}_3}}$.

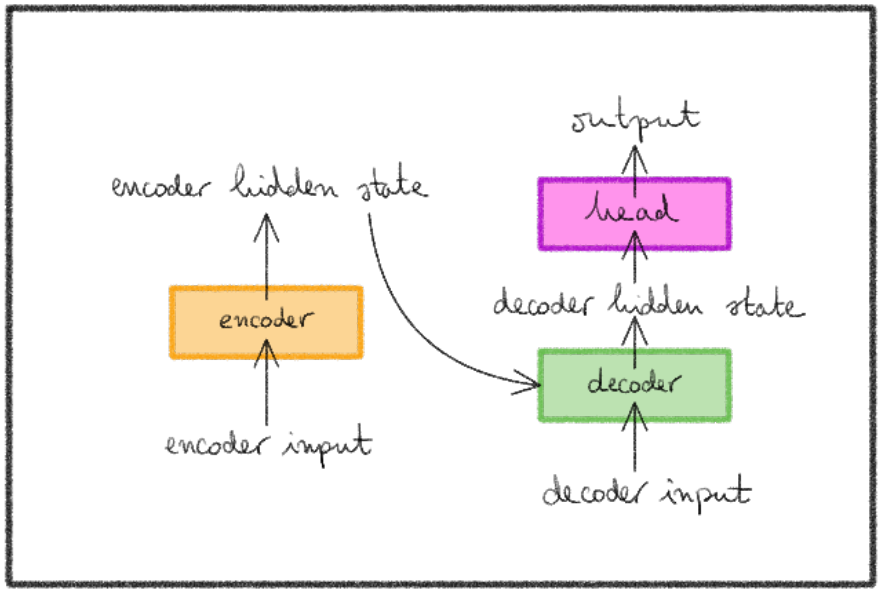
Study the behavior (increasing/decreasing) of the CE with respect to $\hat{z}$.

Encoder-Decoder Models

Also called Seq2Seq



Encoder-Decoder Models

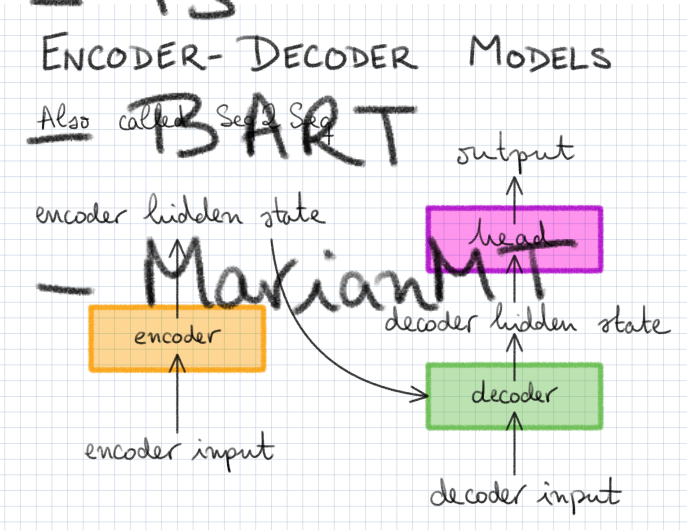


Tasks:

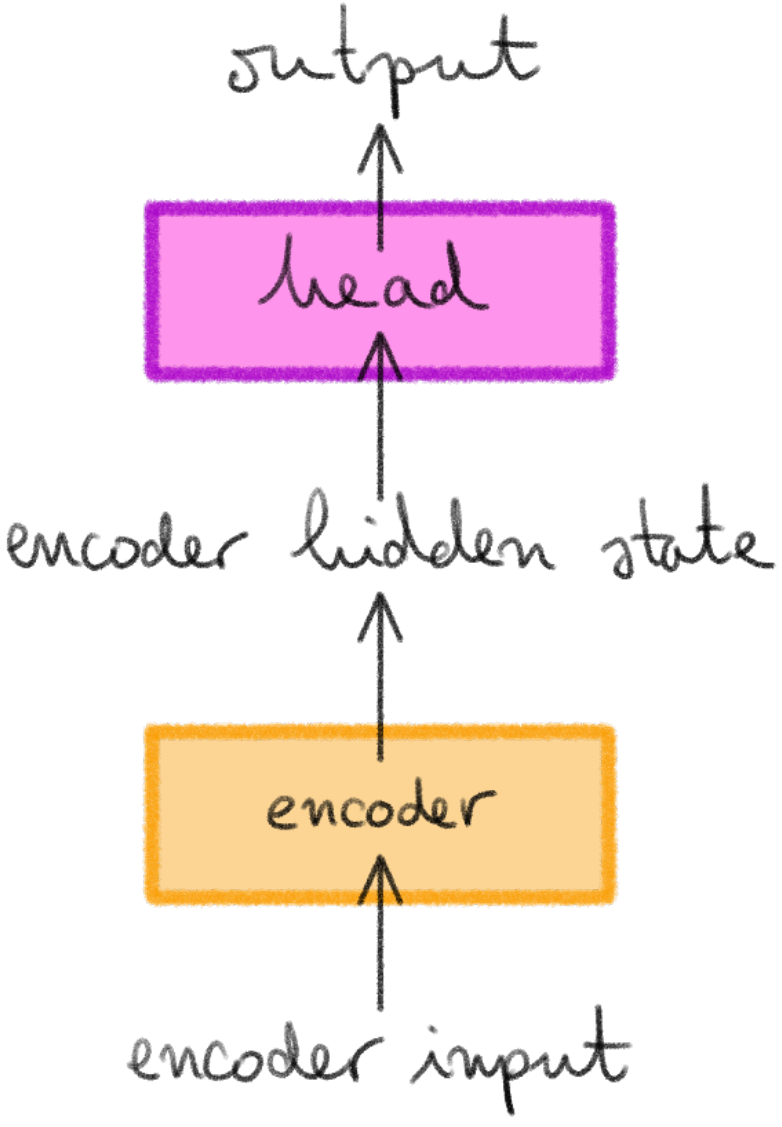
- translation
- summarization
- question answering

Model examples:

- T5
- BART
- MarianMT



Encoder-Only Models



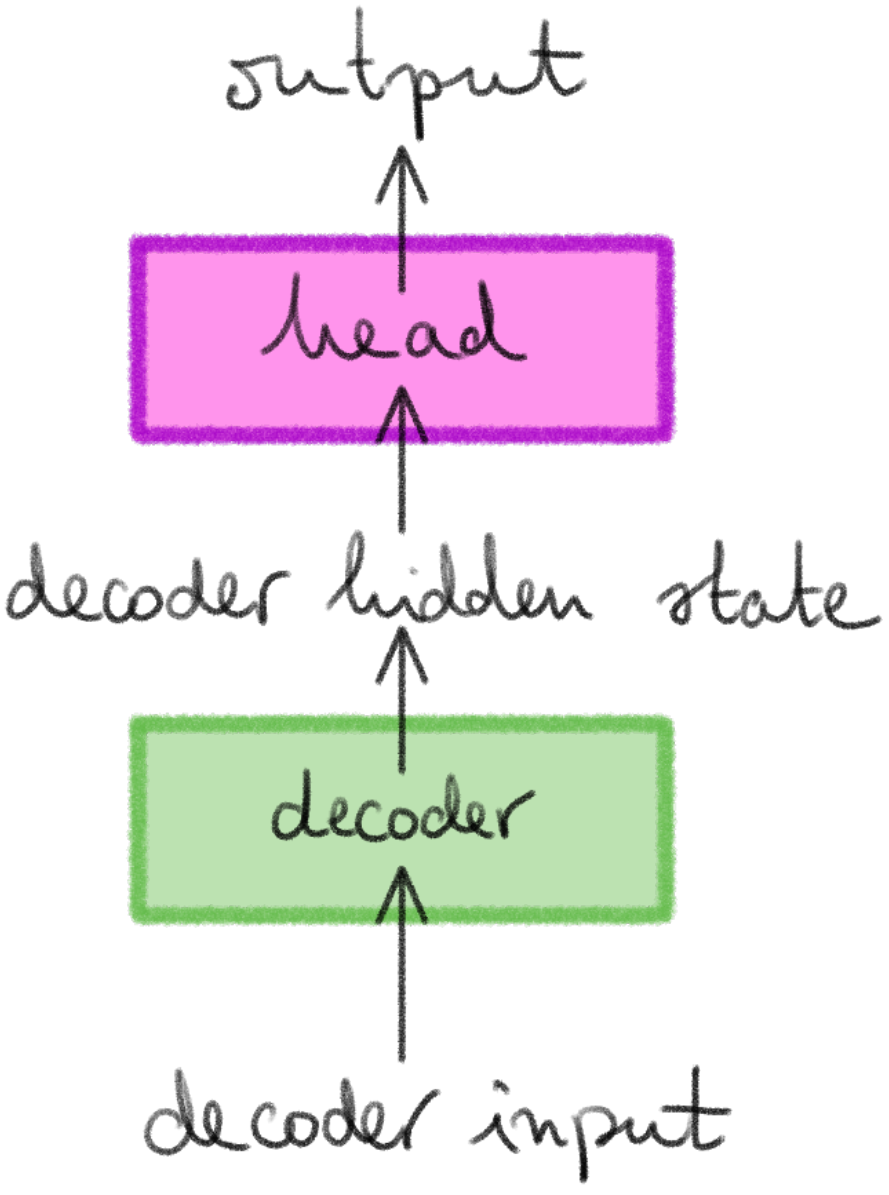
Tasks:

- classification
- named entity recognition

Models:

- BERT family

Decoder-Only Models



Tasks:

- Text generation
- Chatbots

Models:

- GPT family

Kahoot time!

Question Time

- Write down something you understood well

- Write down something you did not fully understand

- Write something you would like to know more about

SUMMARY

- What are the three main types of transformer models? What tasks is each of them used for?

- In pairs: discuss the difference between the inference / training phases of a transformer model for the machine translation task.