Natural Language Processing

Meeting 2025-03-12

Marco Kuhlmann
Department of Computer and Information Science

LINKÖPING UNIVERSITY

Agenda for this meeting

18:00	Introduction & announcements
18:15	Language modelling (Q&A)
18:30	Exercise on language modelling
18:45	Tokenisation and embeddings (Q&A)
19:00	Break
19:15	Bias in word representations
19:30	Transformer-based models (Q&A)
19:45	Outlook on Units 3–4

Introduction and announcements

Date	Activity
2025-01-22	Meeting 1
2025-03-12	Meeting 2
2025-05-07	Meeting 3
2025-06-05	Last day to take the oral exam
2025-08-30	Additional examination 1
2026-01-09	Additional examination 2

"Route card"

https://forms.office.com/e/Kj7AfjCQCs

Checking in (10 minutes)

Language modelling (Q&A)

Quiz 0.2, question 5

5. What is the MLE-estimated probability of a word that is not in the model vocabulary? (1 point)

More details

60% of respondents answered this question correctly.

0	6
1	0
not defined	9 ✓

Formal definition of an n-gram model

n	the model's order (1 = unigram, 2 = bigram, …)
V	a finite set of possible words; the vocabulary
$P(w\|u)$	a probability that specifies how likely it is to observe the word w after the context $(n-1)$-gram u
	one value for each combination of a word w and a context u

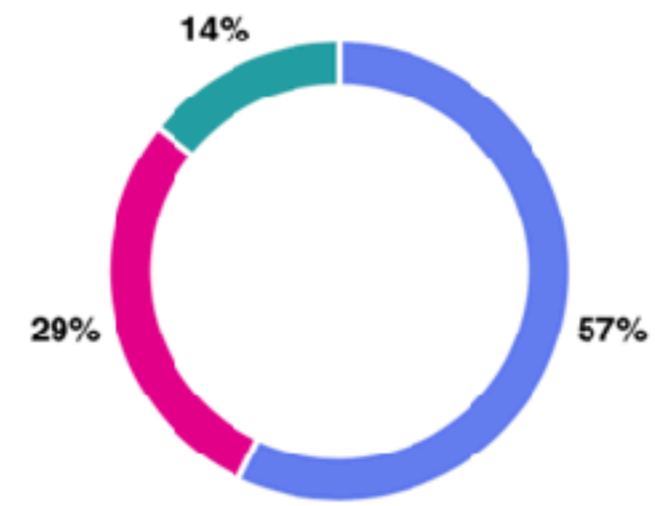
Quiz 0.3, question 1

1. What is meant by n-grams "sharing statistical strength"? (1 point)

More details

57% of respondents answered this question correctly.

n-grams with similar words have similar probabilities	8	✓
frequent n-grams give some of their counts to rare n-grams	4	
n-grams containing unknown words receive probability from the remaining n-grams	2	



Limitations of statistical n-gram models

Goldberg § 9.3.2

- Scaling to larger *n*-gram sizes is problematic, both for computational reasons and because of data sparsity.
- Techniques for mitigating these issues require careful engineering and are not sufficiently flexible.

  smoothing, interpolation
- Without additional effort, *n*-gram models are unable to share statistical strength across “similar” words.

  Observations of *a red apple* do not affect estimates for *the yellow apples*.

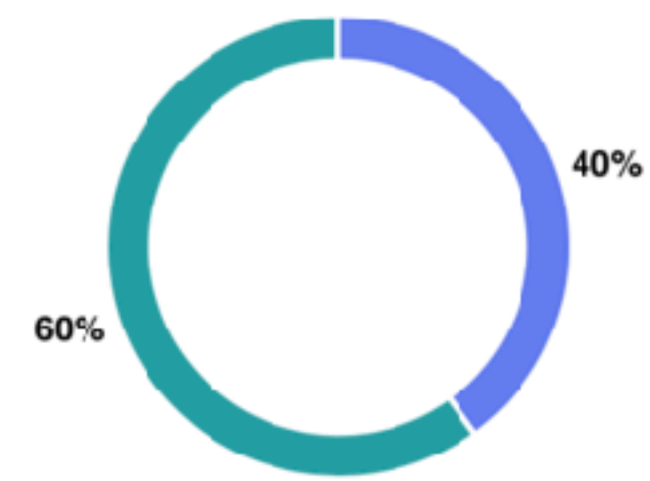
Quiz 0.5, question 4

4. In the LSTM architecture, what is the output of the forget gate? (1 point)

More details

60% of respondents answered this question correctly.

● a floating-point number between 0 and 1	6	
● a vector of zeros and ones	0	
● a vector of floating-point numbers between 0 and 1	9	✓



Gating mechanism

$$\begin{bmatrix}1\\2\\3\\4\end{bmatrix} \odot \begin{bmatrix}0\\1\\1\\0\end{bmatrix} + \begin{bmatrix}5\\6\\7\\8\end{bmatrix} \odot \begin{bmatrix}1\\0\\0\\1\end{bmatrix} = \begin{bmatrix}5\\2\\3\\8\end{bmatrix}$$

$$\boldsymbol{h}_{t-1} \qquad \boldsymbol{g} \qquad \boldsymbol{x}_t \qquad 1-\boldsymbol{g} \qquad \boldsymbol{h}_t$$

The gating masks **g** are learned values between 0 and 1.

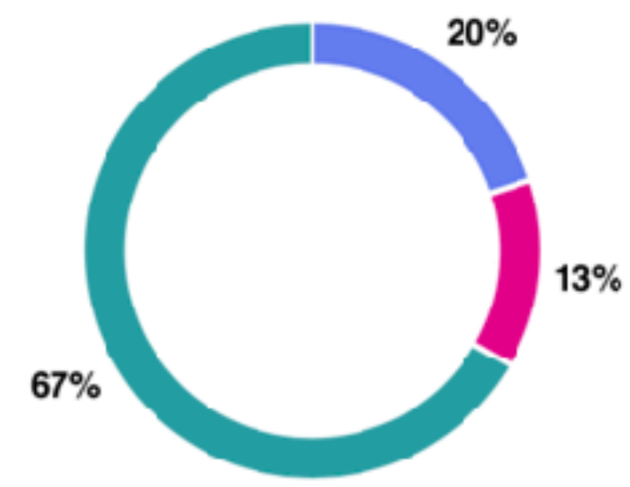
Quiz 0.6, question 3

3. When training RNN language models, how would we expect the training loss to change when we double the backpropagation-through-time horizon? (1 point)

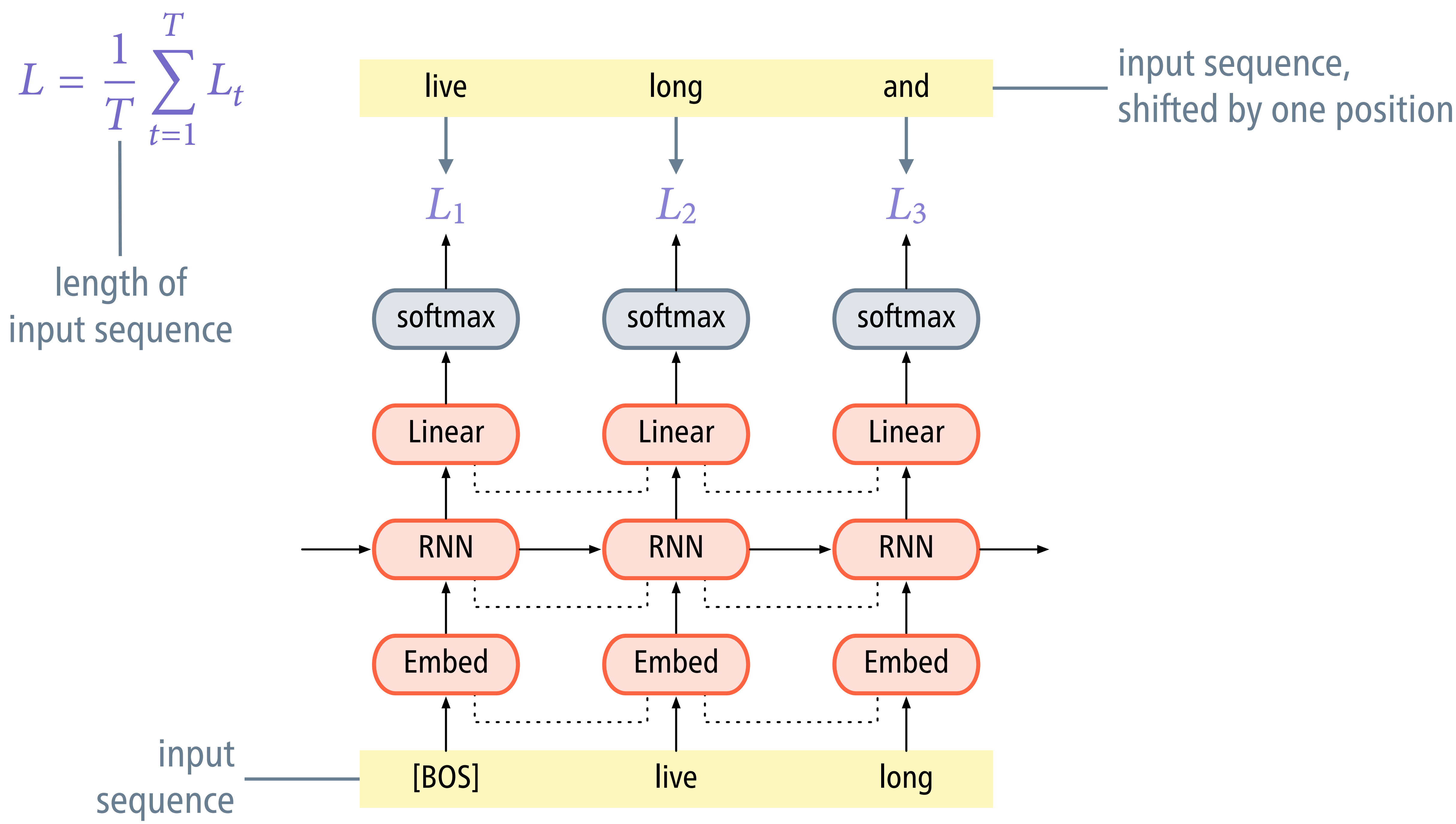
More details

67% of respondents answered this question correctly.

It should double	3	
It should half	2	
It should not change	10	✓

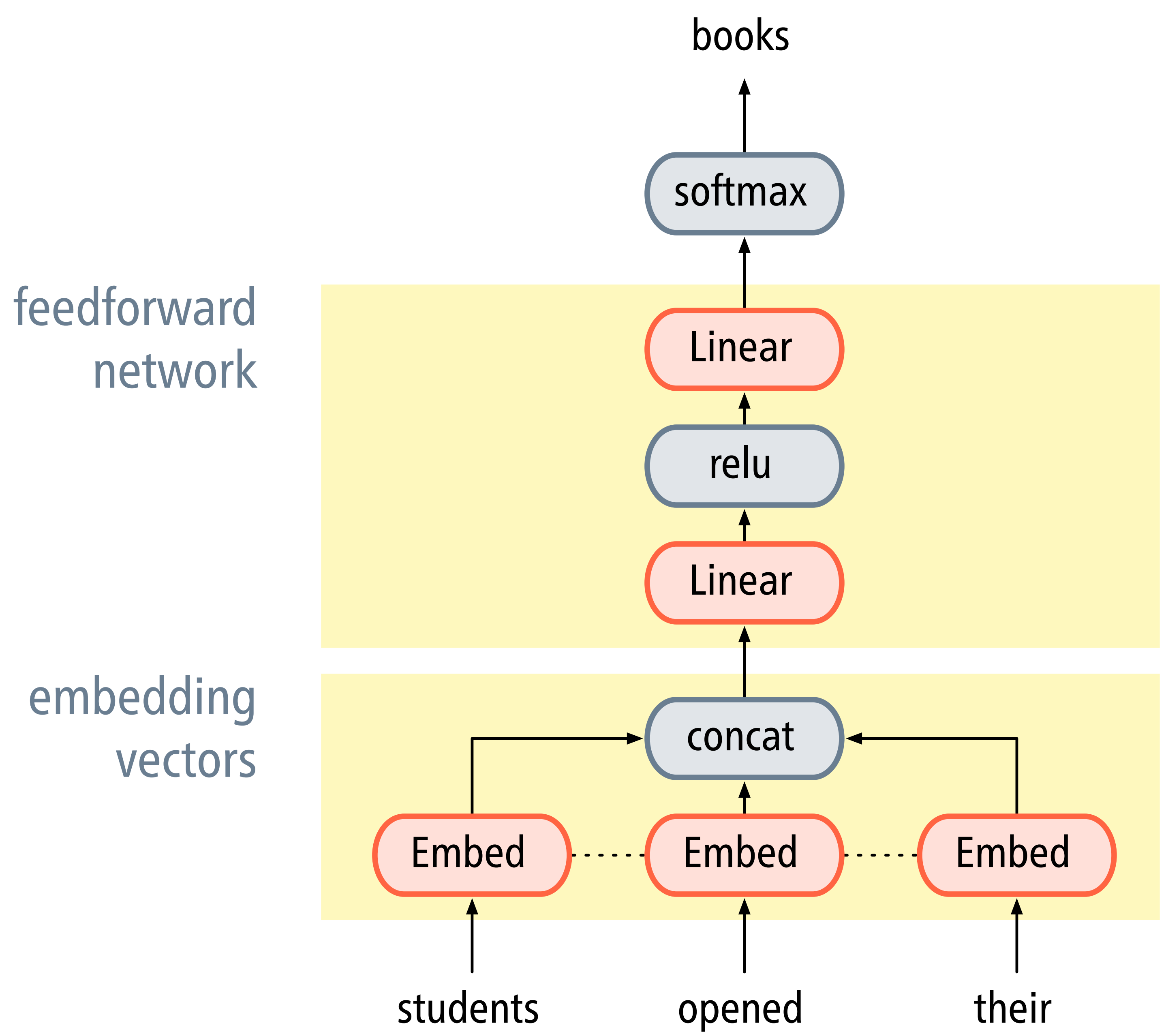


Training RNN language models

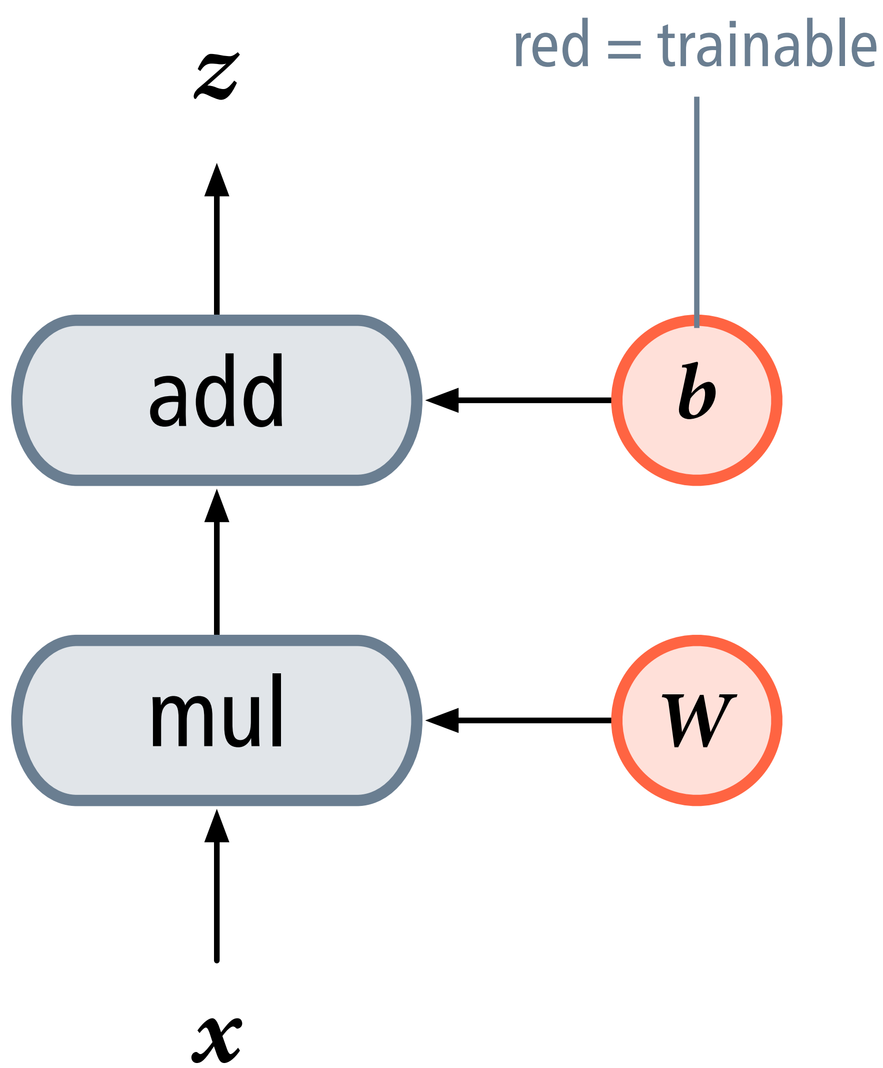


Exercise: Language modelling

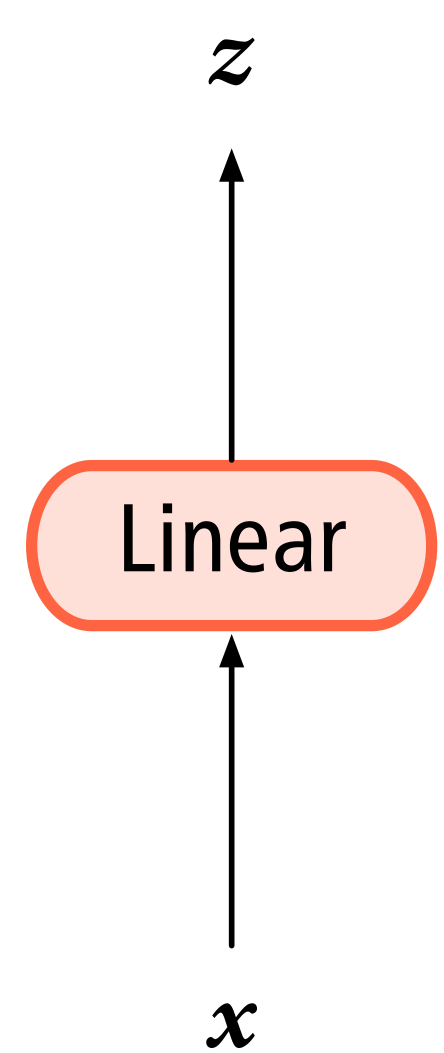
A neural n-gram model



Graphical notation



computation graph



shorthand notation

Linear layers

```
>>> import torch

>>> # Create a linear model
>>> model = torch.nn.Linear(784, 10)

>>> # Inspect the shapes of the model parameters
>>> [p.shape for p in model.parameters()]
[torch.Size([10, 784]), torch.Size([10])]

>>> # Feed random data and inspect the shape of the output
>>> model.forward(torch.rand(784)).shape
torch.Size([10])
```

Embedding layers

```
s2i = {'great': 0, 'monster': 1, 'movie': 2}

import torch
```

number of words to embed

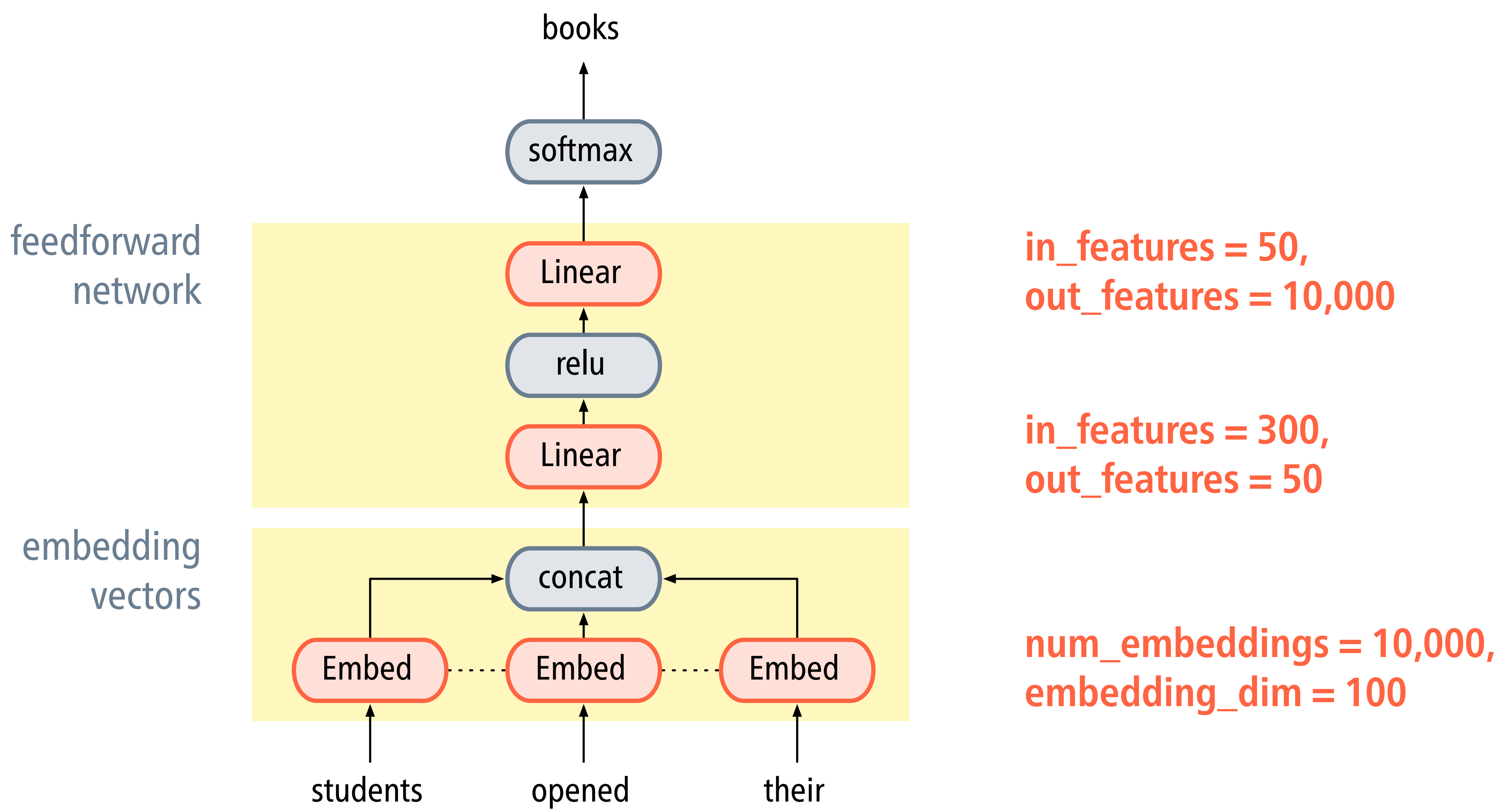
```
emb = torch.nn.Embedding(3, 2)
```

size of each embedding vector

```
emb(torch.tensor(s2i['monster'], dtype=torch.long))
# tensor([0.6399, 0.1779], grad_fn=<EmbeddingBackward>)

emb(torch.tensor([s2i[s] for s in s2i], dtype=torch.long))
tensor([[ 0.4503, -0.1549],
        [ 0.6399,  0.1779],
        [-0.6537, -0.5875]], grad_fn=<EmbeddingBackward>)
```

A neural n-gram model



Tokenisation and embeddings (Q&A)

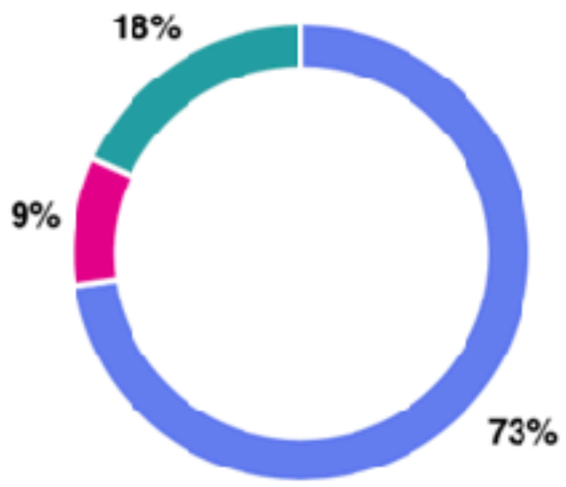
Quiz 1.2, question 4

4. Suppose we apply the BPE algorithm to a very long English text and do a lot of merge rules. Which token would be expect not to see in our final vocabulary? (1 point)

More details

73% of respondents answered this question correctly.

●	[eaux]	8 ✓
●	[tion]	1
●	[thes]	2



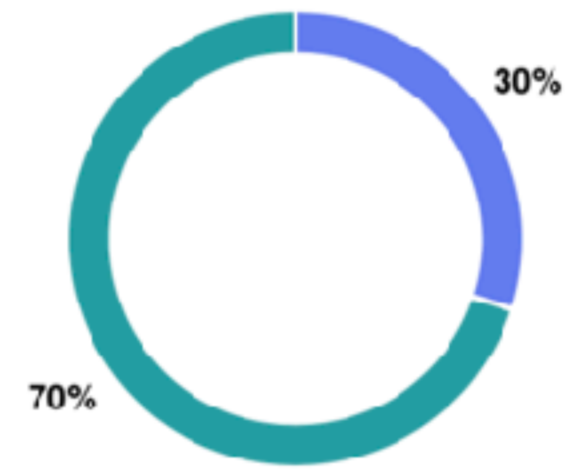
Quiz 1.3, question 3

3. True or false? "The embeddings learned for the words *university* and *school* are similar." (1 point)

More details

70% of respondents answered this question correctly.

True	3
False	0
Depends on the training task	7 ✓



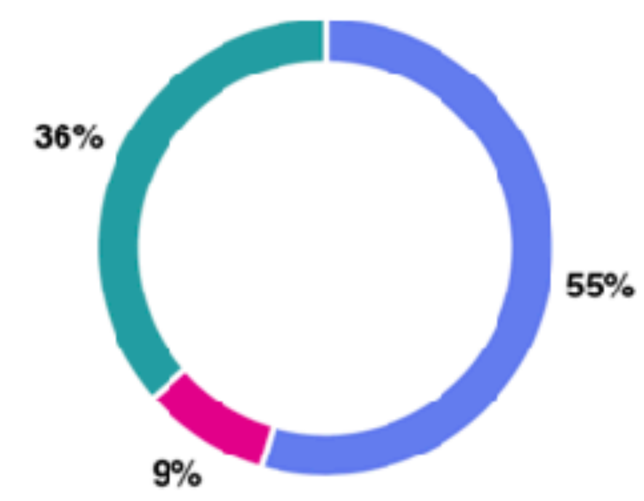
Quiz 1.6, question 1

1. The standard skip-gram model (without negative sampling) is a k-class classification problem. What would be a realistic value for k? (1 point)

More details

36% of respondents answered this question correctly.

2	6	
2,000	1	
20,000	4	✓



The skip-gram model in detail (1)

- We maintain two separate vector representations: one for target words and one for context words. Initially, they are random.
- The probability of a context word c given a target word w is defined using the softmax function:

vector representation for context words

vector representation for target words

$$P(c \mid w\,; \boldsymbol{\theta}) = \frac{\exp\left(\boldsymbol{v}'^{\top}_{c} \boldsymbol{v}_{w}\right)}{\sum_{x \in V} \exp\left(\boldsymbol{v}'^{\top}_{x} \boldsymbol{v}_{w}\right)}$$

all parameters of the model

Bias in word embeddings

Embedding bias and occupation participation

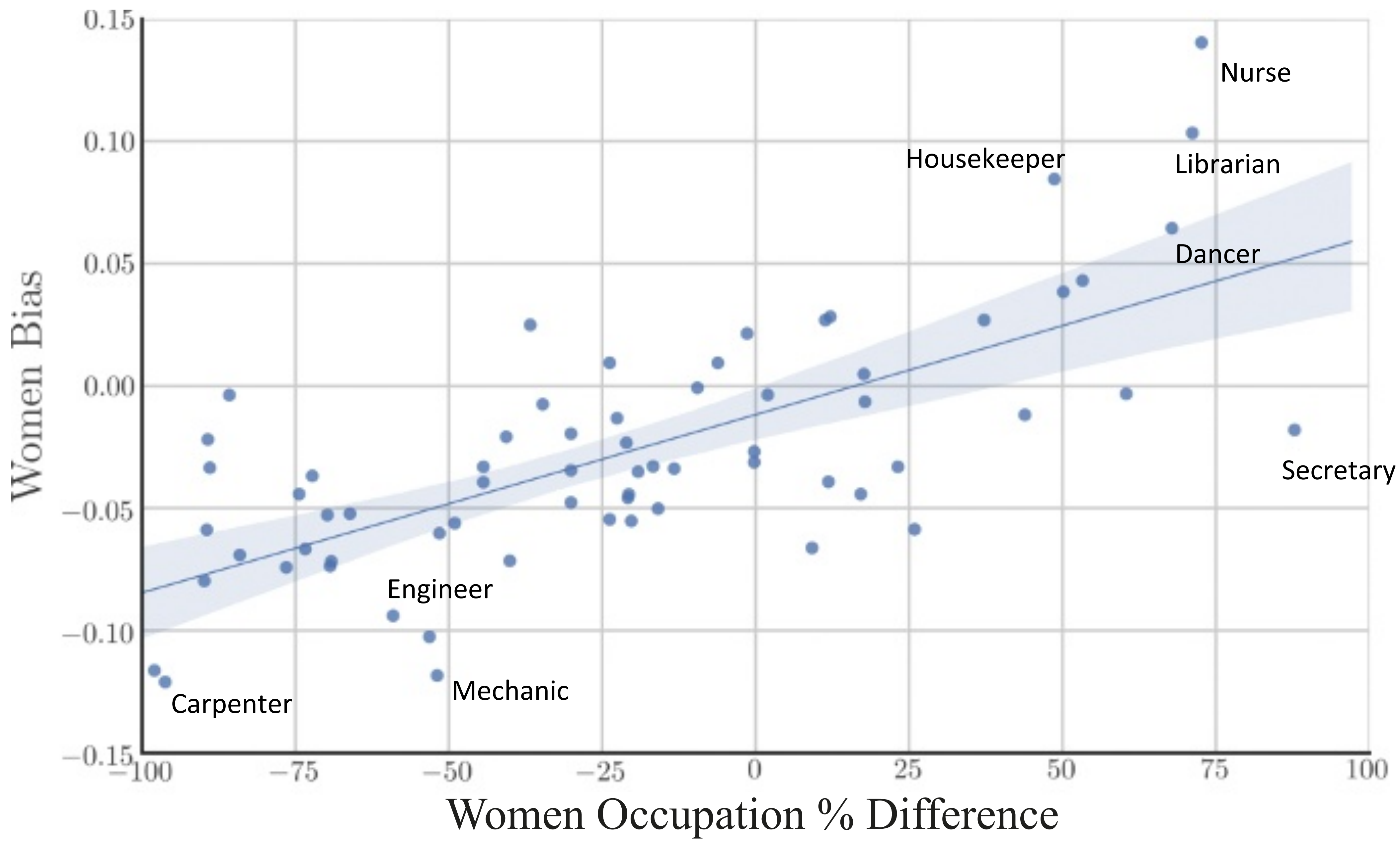


Figure 1 from Garg et al. (2018)

Vem styr debatten om migrationen?

20 november 2018

Mikael Sönne

Hur har det offentliga samtalet om invandring förändrats i Sverige? Och vem ligger bakom den förändringen - politikerna, medierna eller allmänheten i sociala medier? Det ska ett nytt forskningsprojekt vid LiU försöka ta reda på.

"Att kunna analysera fritt skriven text frigör forskningen", säger Marc Keuschnigg. B   ild: Mikael Sönne

Partner discussion

- **Partner A:** “The results of Garg et al. clearly show that word embeddings contain harmful biases. There is a risk that we build these biases into our models. We should therefore develop methods for de-biasing embeddings.”
- **Partner B:** “The results of Garg et al. simply show statistical correlations in the data; I would not call them harmful biases. The results suggest that word embeddings make an interesting tool for data-driven research in the social sciences.”

Transformer-based models (Q&A)

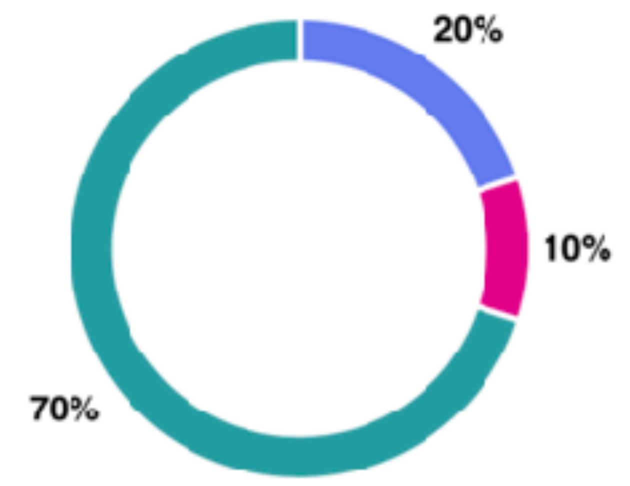
Quiz 2.2, question 1

1. Which of the following tasks do not usually lend themselves to the use of autoregressive language models? (1 point)

More details

70% of respondents answered this question correctly.

- machine translation 2
- text summarisation 1
- document classification 7 ✓

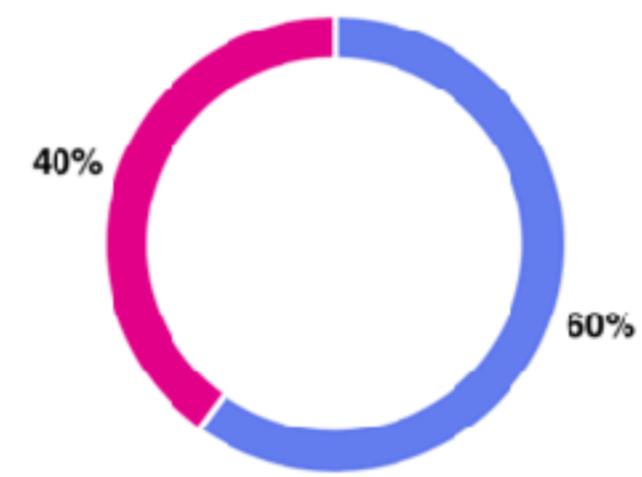


Quiz 2.2, question 6

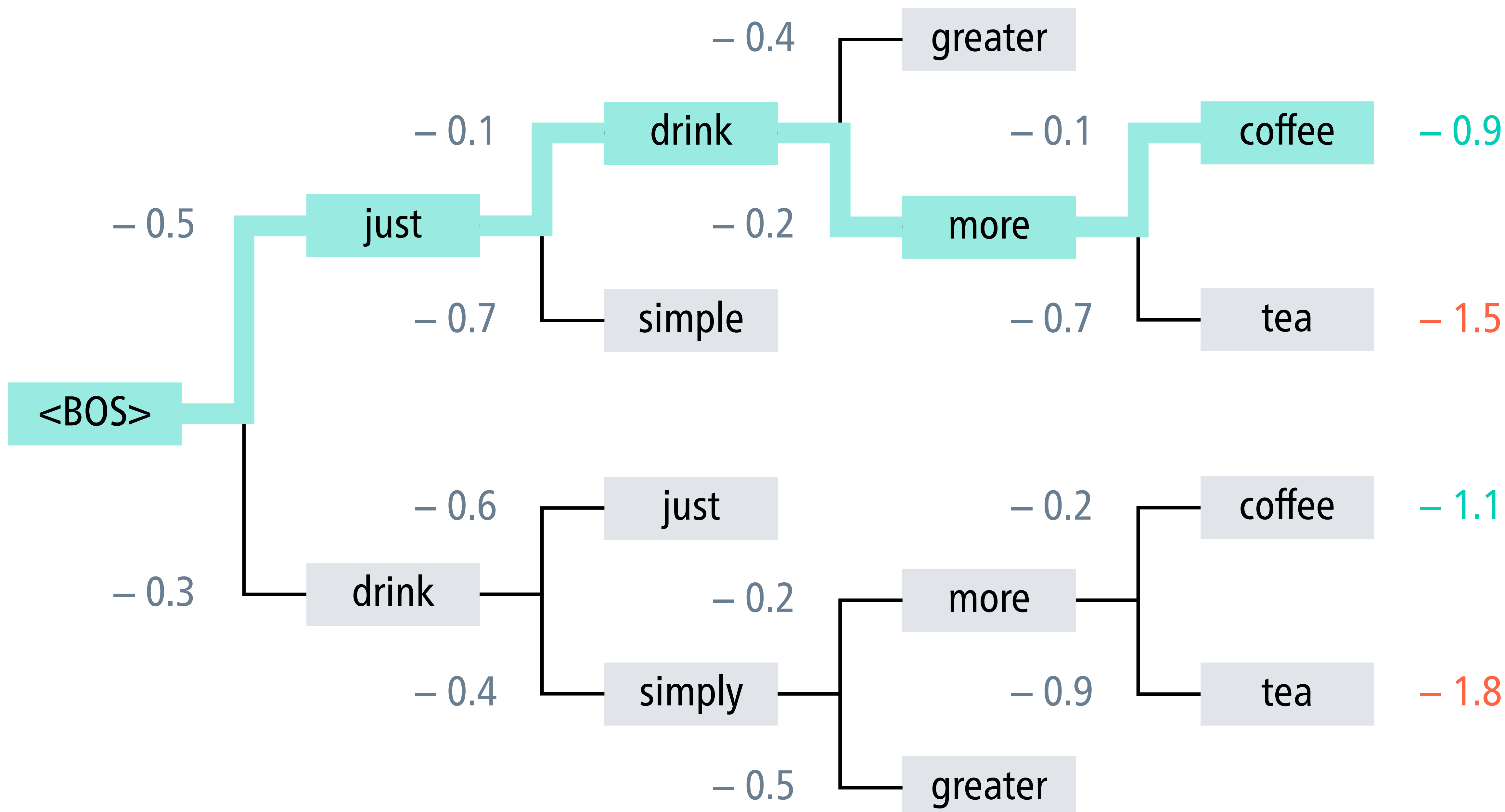
5. Why do we use length normalisation together with beam search in decoding? (1 point) More details

60% of respondents answered this question correctly.

- We do not want to penalise long translations. 6 ✓
- We do not want to penalise short translations. 4
- We want to avoid numerical overflow. 0



Beam search example



Quiz 2.3, question 1

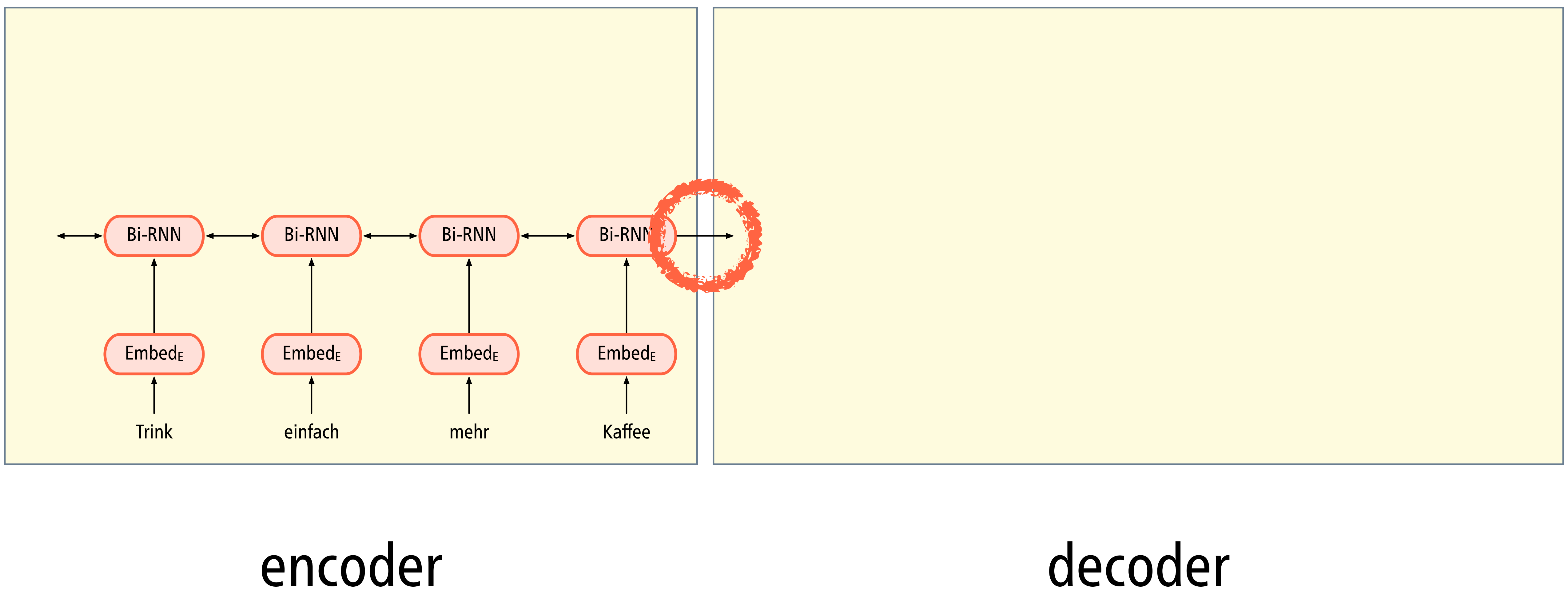
1. Suppose we encode the sentence "Gold is heavier than silver" using a bi-directional recurrent neural network. What issue does th e **recency bias** cause? (1 point)

More details

10% of respondents answered this question correctly.

- The final hidden state contains more information about *Gold* than about *heavier*. 1 ✓
- The final hidden state contains more information about *Gold* than about *silver*. 0
- The final hidden state contains more information about *silver* than about *Gold*. 9

Recency bias in recurrent neural networks



Sutskever et al. (2014)

Quiz 2.3, question 3

3. Consider following values for the example in slide 6 ff.:

s = [0.4685, 0.9785], h1 = [0.5539, 0.7239], h2 = [0.4111, 0.3878], h3 = [0.2376, 0.1264]

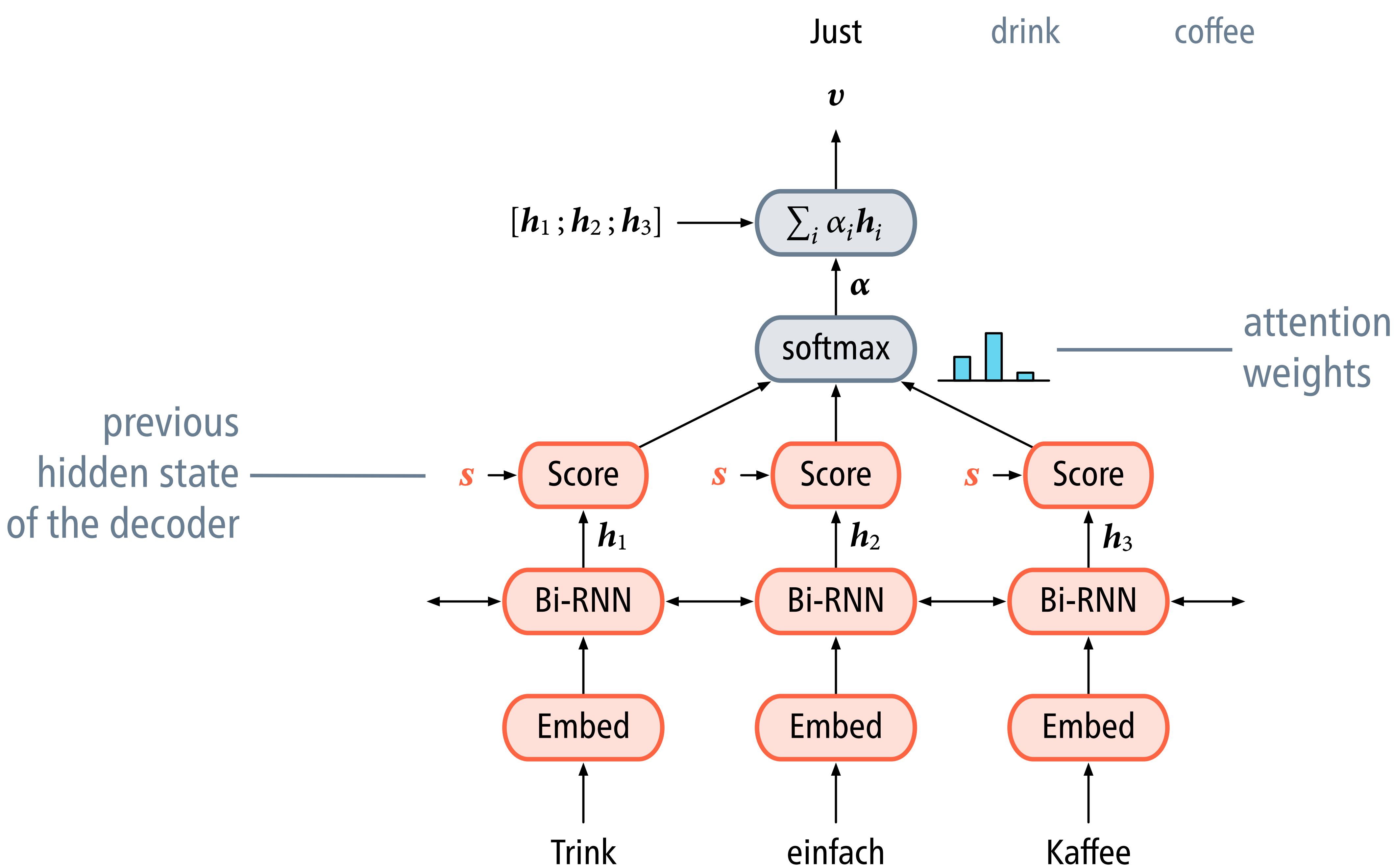
More details

Assuming that the attention score is computed using the dot product, what is v? (1 point)

40% of respondents answered this question correctly.

- [0.4387, 0.4855] 4 ✓
- [0.9678, 0.5721, 0.2350] 6
- [0.4643, 0.3126, 0.2231] 0

Attention for translation



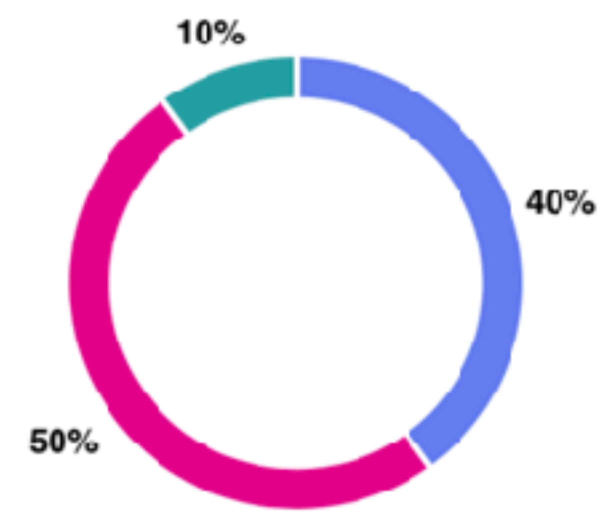
Bahdanau et al. (2015)

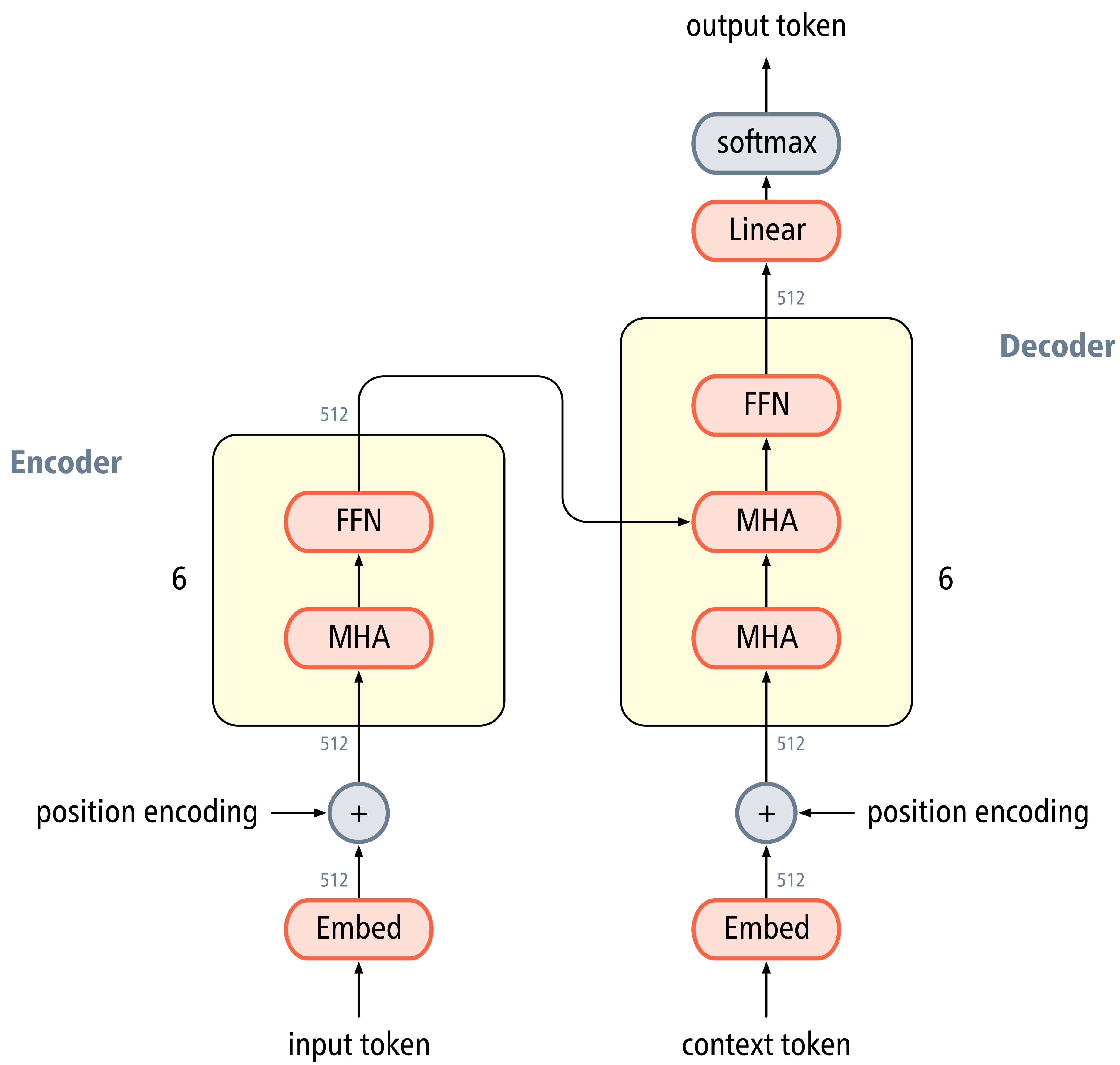
Quiz 2.4, question 5

5. True or false: Permuting the input tokens to a Transformer encoder does not change the final token representations. (1 point) More details

50% of respondents answered this question correctly.

True	4
False	5 ✓
Depends on the input tokens	1




output token
softmax
Linear
512
Decoder
512
FFN
MHA
Encoder
FFN
6
MHA
MHA
6
512
512
position encoding
+
+
position encoding
512
512
Embed
Embed
input token
context token

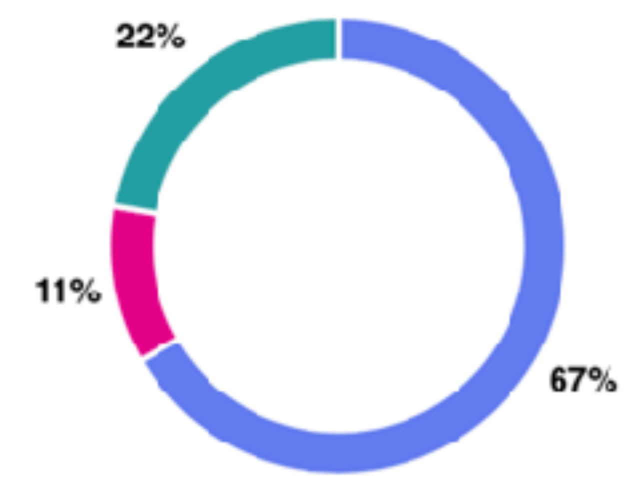
Quiz 2.5, question 2

2. Looking at the original GPT model architecture (Radford et al., 2018), what is the approximate number of trainable parameters in the FNN? (1 point)

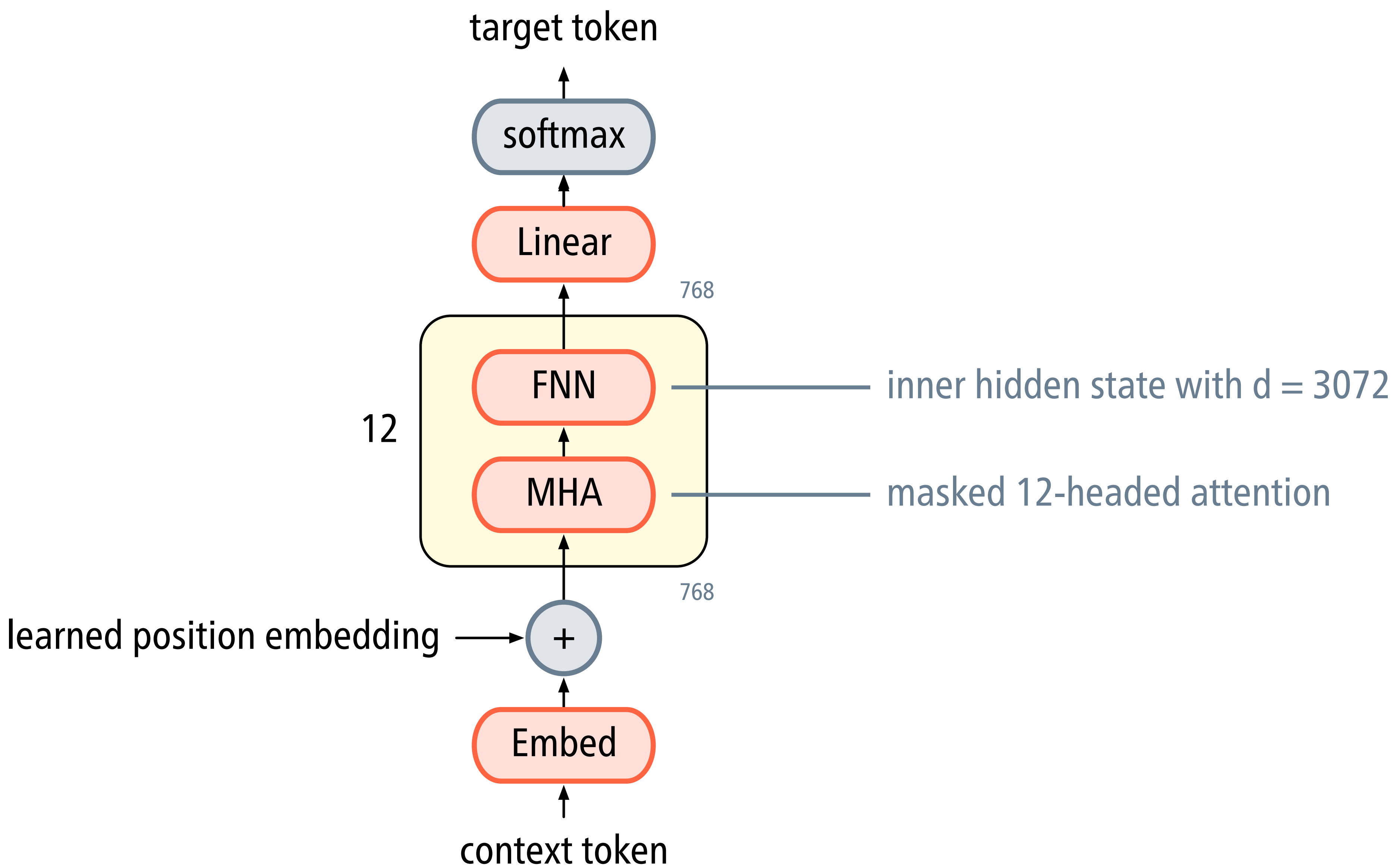
More details

67% of respondents answered this question correctly.

●	4718592	6 ✓
●	589824	1
●	9216	2

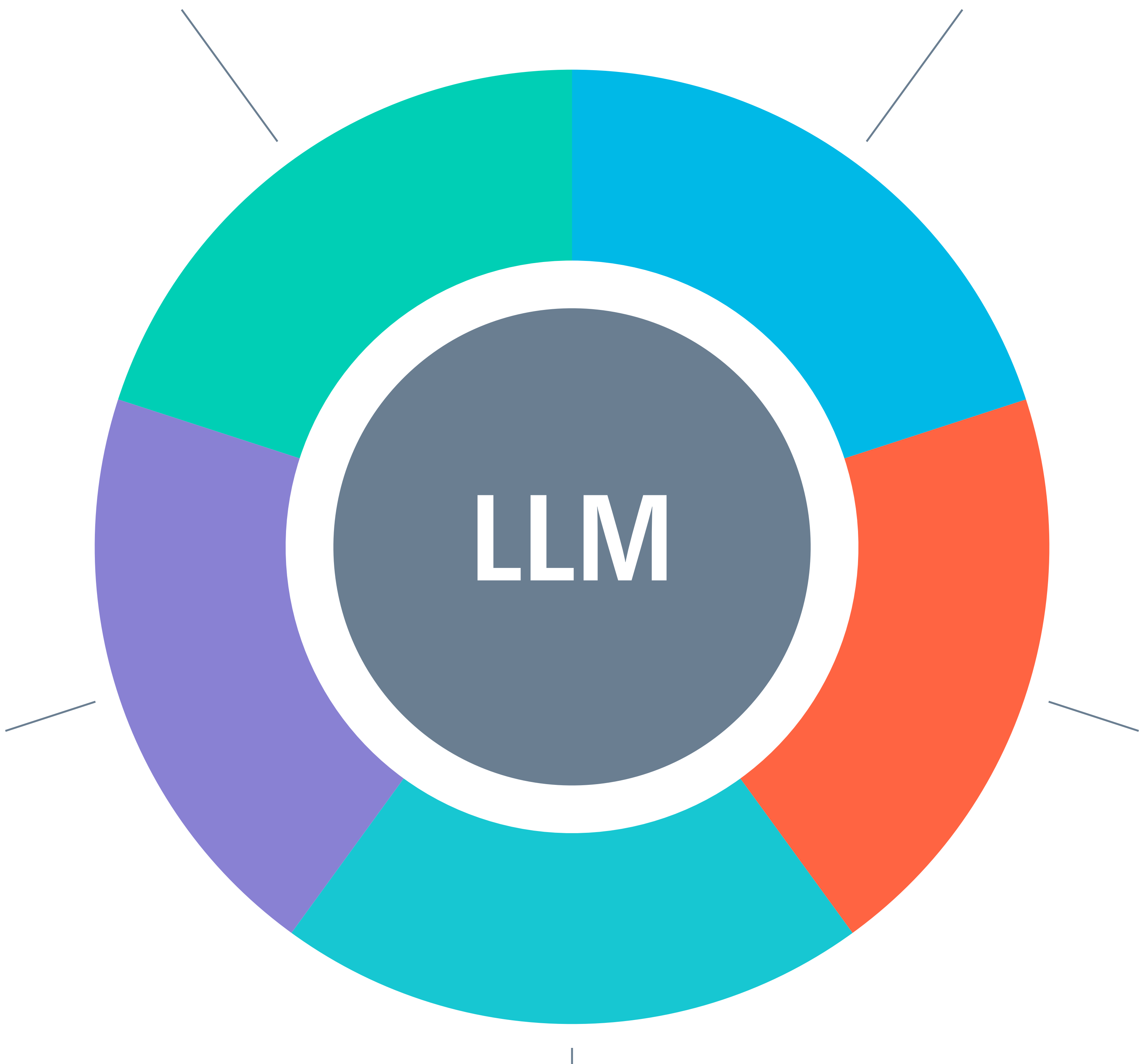


GPT model architecture



Radford et al. (2018)

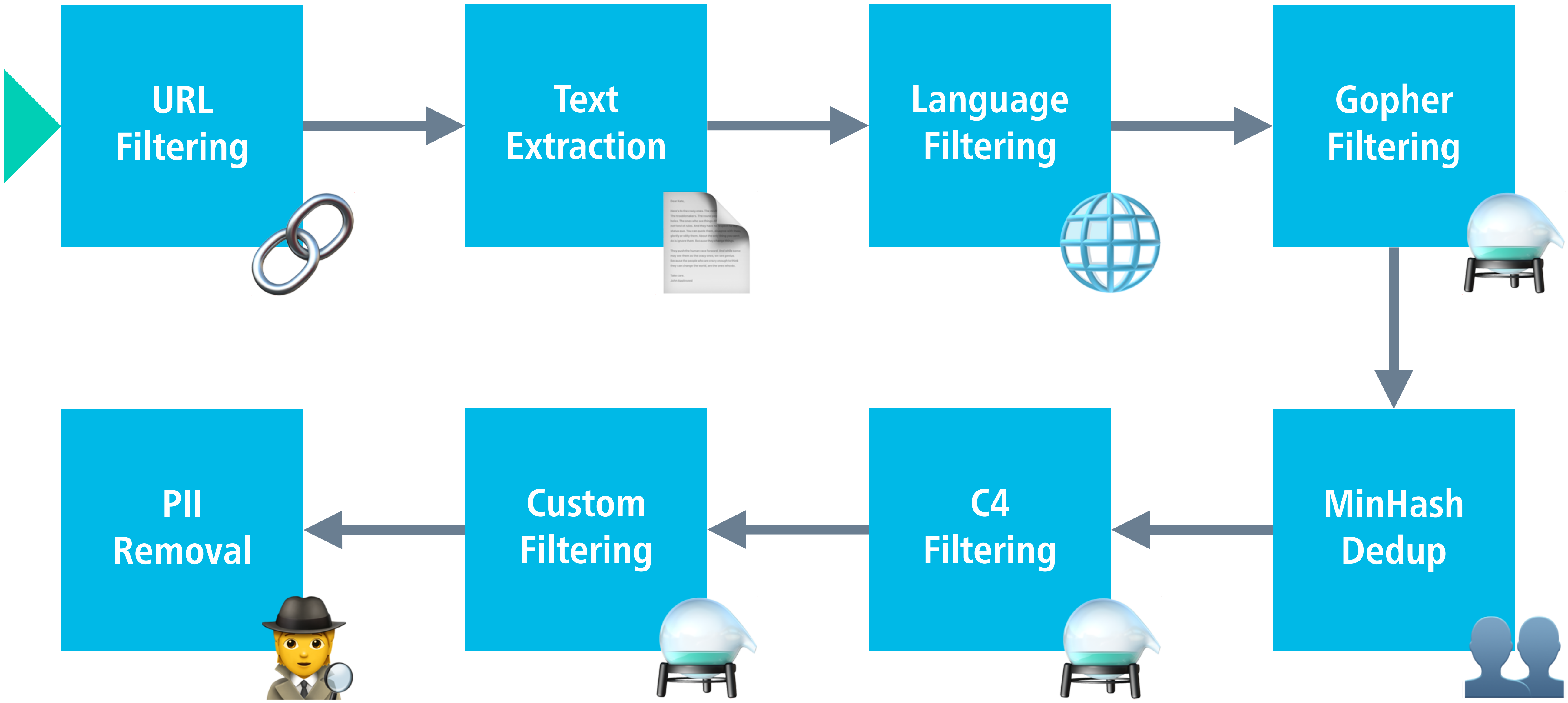
Outlook on Units 3–4


LLM

	unsupervised pre-training	instruction fine-tuning	reward modelling	reinforcement learning
data	**raw text from the Internet** trillions of words low quality, high quantity	**ideal dialogues** 10k–100k low quantity, high quality	**annotated dialogues** 100k–1M low quantity, high quality	**generated dialogues** 10k–100k low quantity, high quality
algorithm	**language modelling** predict the next word	**language modelling** predict the next word	**binary classification** reward consistent with preferences?	**reinforcement learning** generate text for maximal reward
resources	1000s of GPUs several months of training time GPT, Llama	1–100 GPUs several days of training time	1–100 GPUs several days of training time	1–100 GPUer several days of training time ChatGPT, Claude

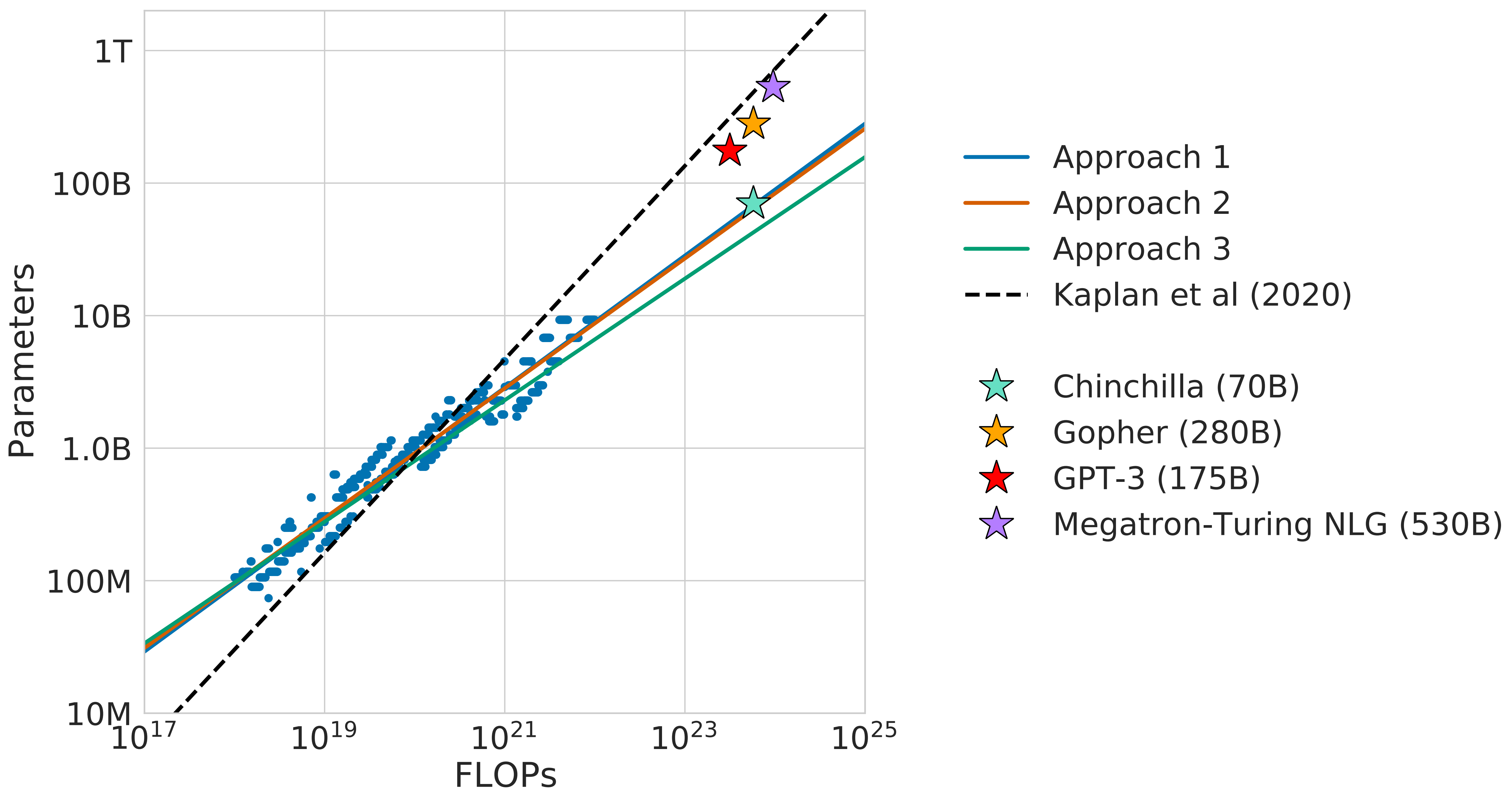
language model → assistant model

The FineWeb pipeline



Penedo et al. (2024)

Large language models can be too large



Chain-of-thought prompting

Wei et al. (2022)

Standard prompting

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 balls. How many tennis balls does he have now?

A: The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

A: **The answer is 11.**

Chain-of-thought prompting

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 balls each is 6 balls. 5 + 6 = 11. The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

A: **The cafeteria had 23 apples originally. They used 20 to make lunch. So they had 23 – 20 = 3. They bought 6 more apples, so they had 3 + 6 = 9. The answer is 9.**