Large language models

Inference, tokenization and small models

Dr. ing. Pieter Delobelle - 30/04/2025

Pieter.ai

Pieter Delobelle

LLM engineer at Aleph Alpha, prev. KU Leuven & Apple

Postdoc and PhD from KU Leuven's DTAI research group

Working on fairness issues in language models

e.g. trying to remove gender biases

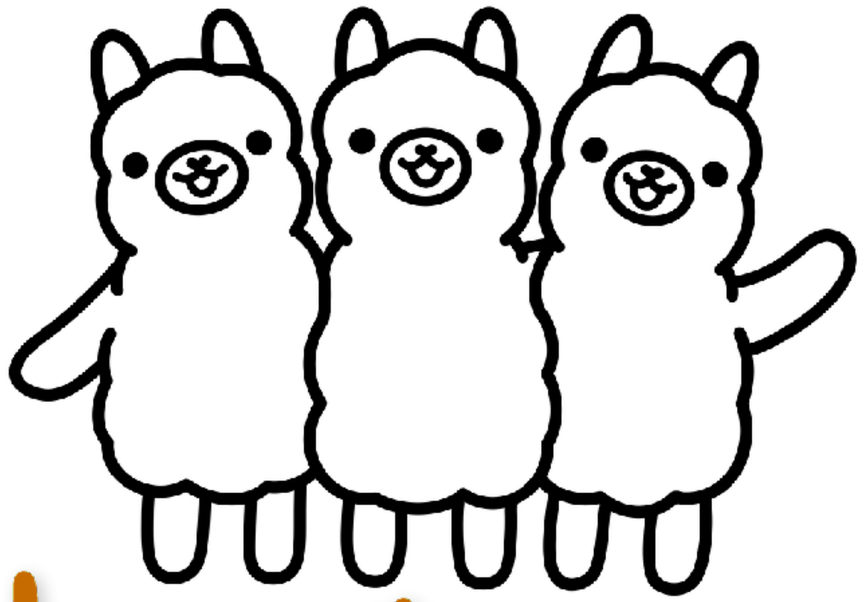
First author of the Dutch RobBERT model in 2020

state-of-the-art Dutch BERT language model

Expert advisor for the EU's AI Act Code of Practice

and prev. member of the KU Leuven GenAI board
and technical advisor in a strategic litigation case against companion AIs


ChatGPT



llama n llama3

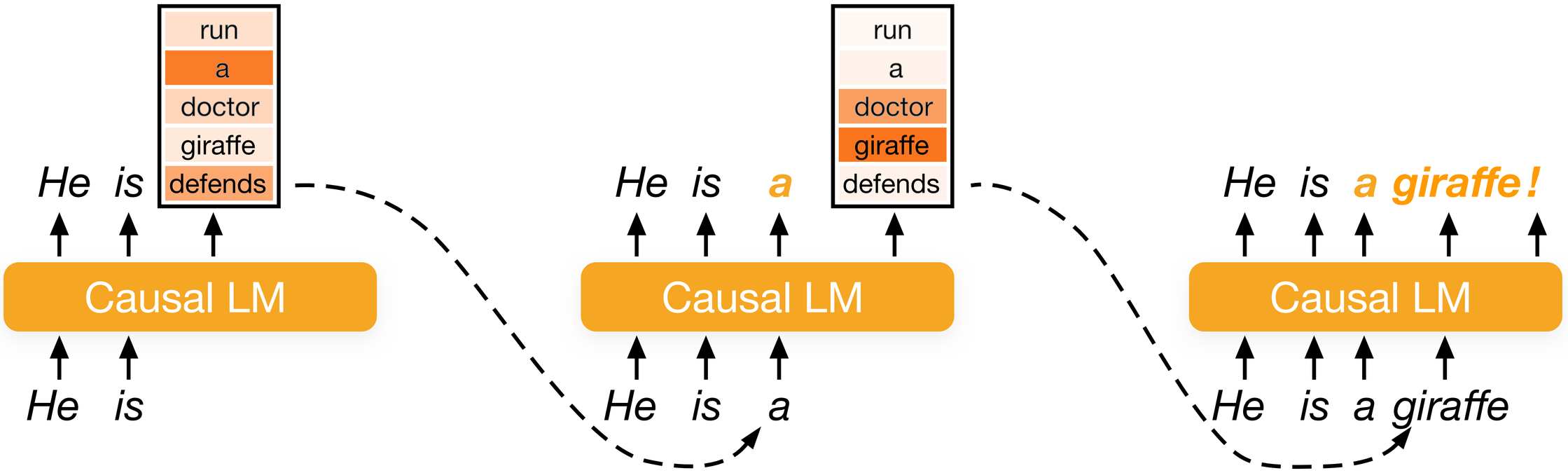
Agenda

Tokenization

Language-specific models

Inference: how to run an LLM efficiently

Generating text with LMs



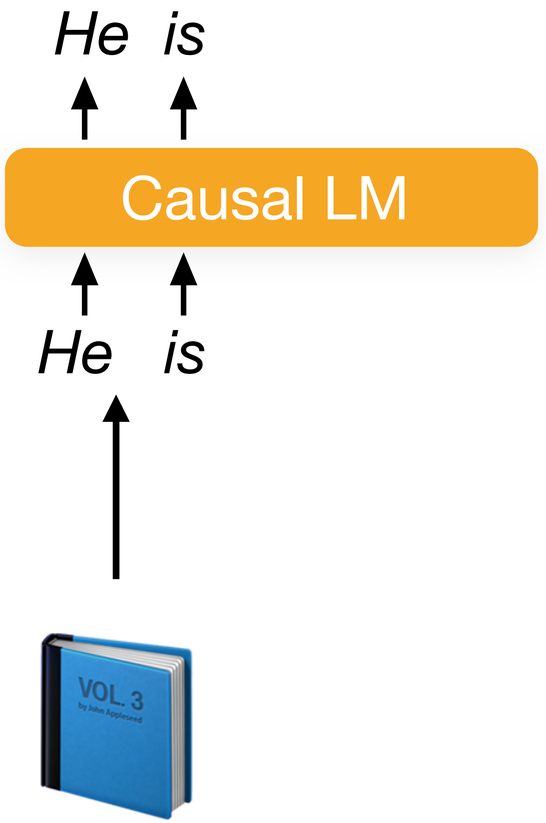
Parts of a language models

‘Heads’ of a language model
How a model predicts the next word

Attention mechanism
Each word affects the other words

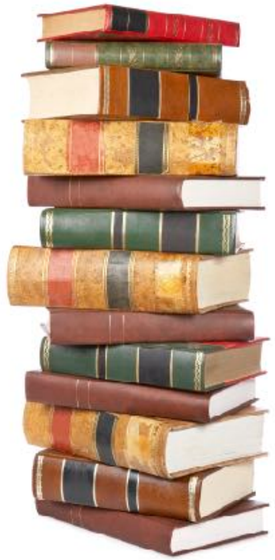
Tokenizer
How a model understands text

Training data
What a model learns



Training data

wikipedia



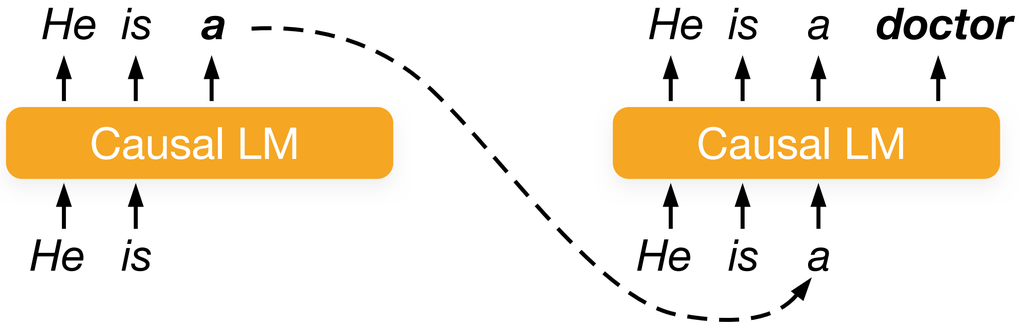
(copyright free) books

scraped data

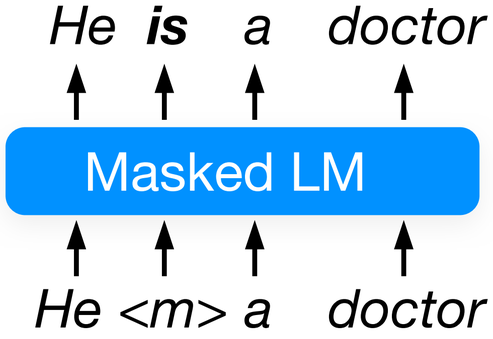
Oscar corpus

Language modeling

1. Autoregressive language modeling

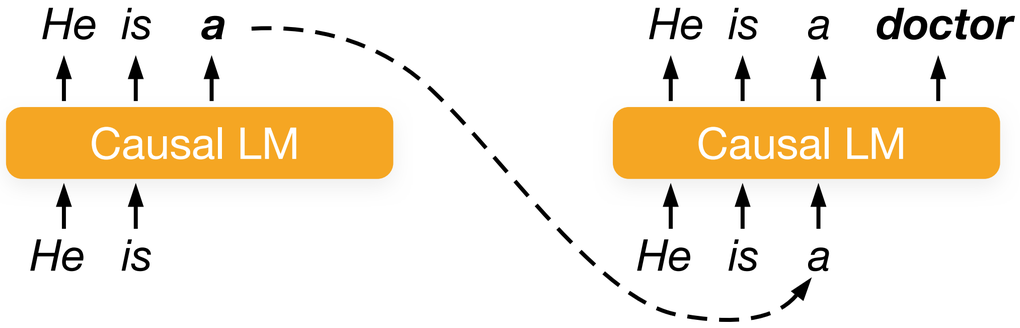


2. Masked language modeling



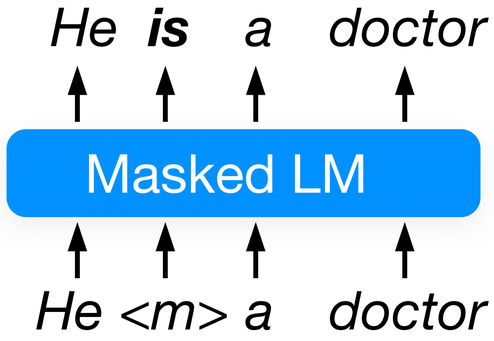
Language modeling

1. Autoregressive language modeling



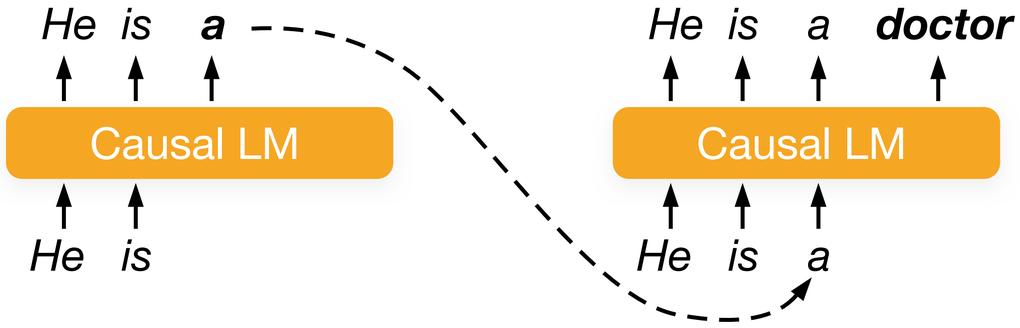
RobBERT

2. Masked language modeling

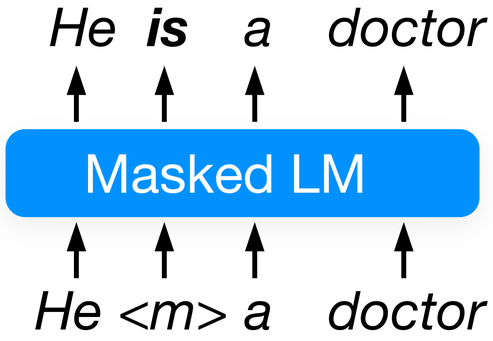


Language modeling

1. Autoregressive language modeling



2. Masked language modeling



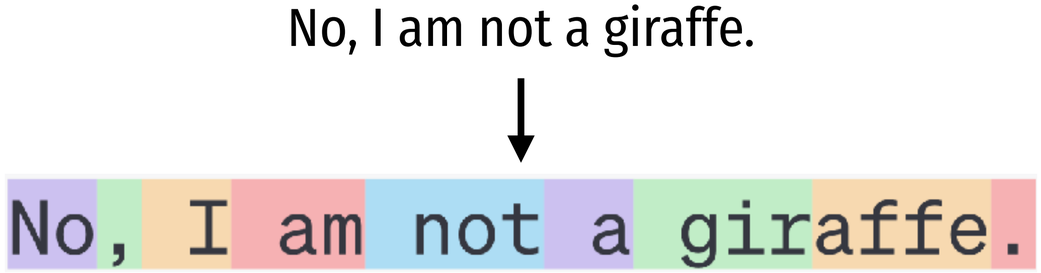
Tokenizing the training data

an example

No, I am not a giraffe.

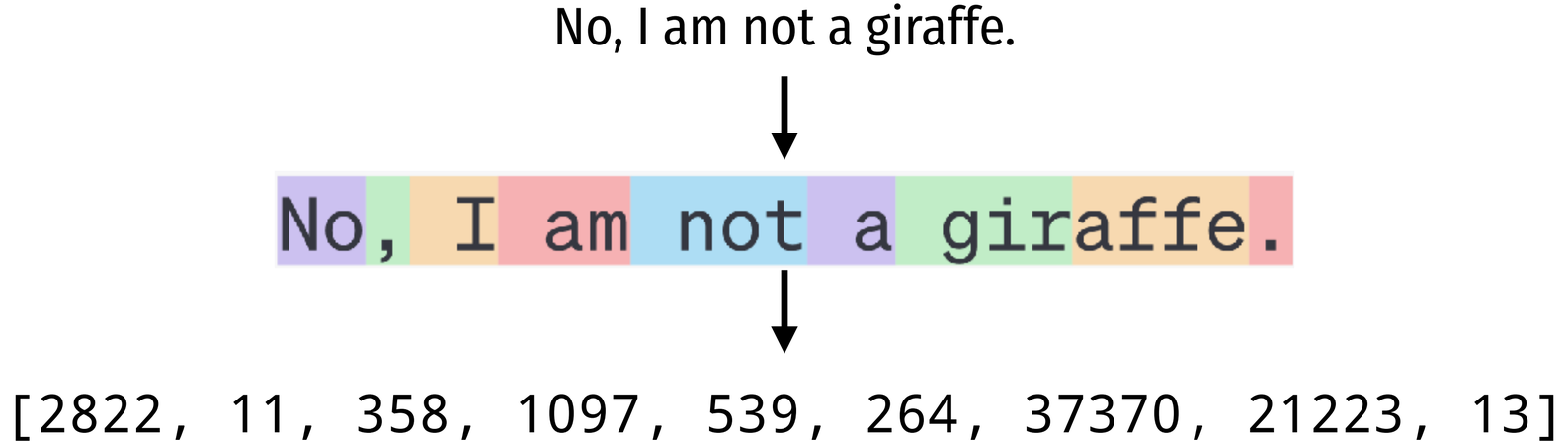
Tokenizing the training data

an example



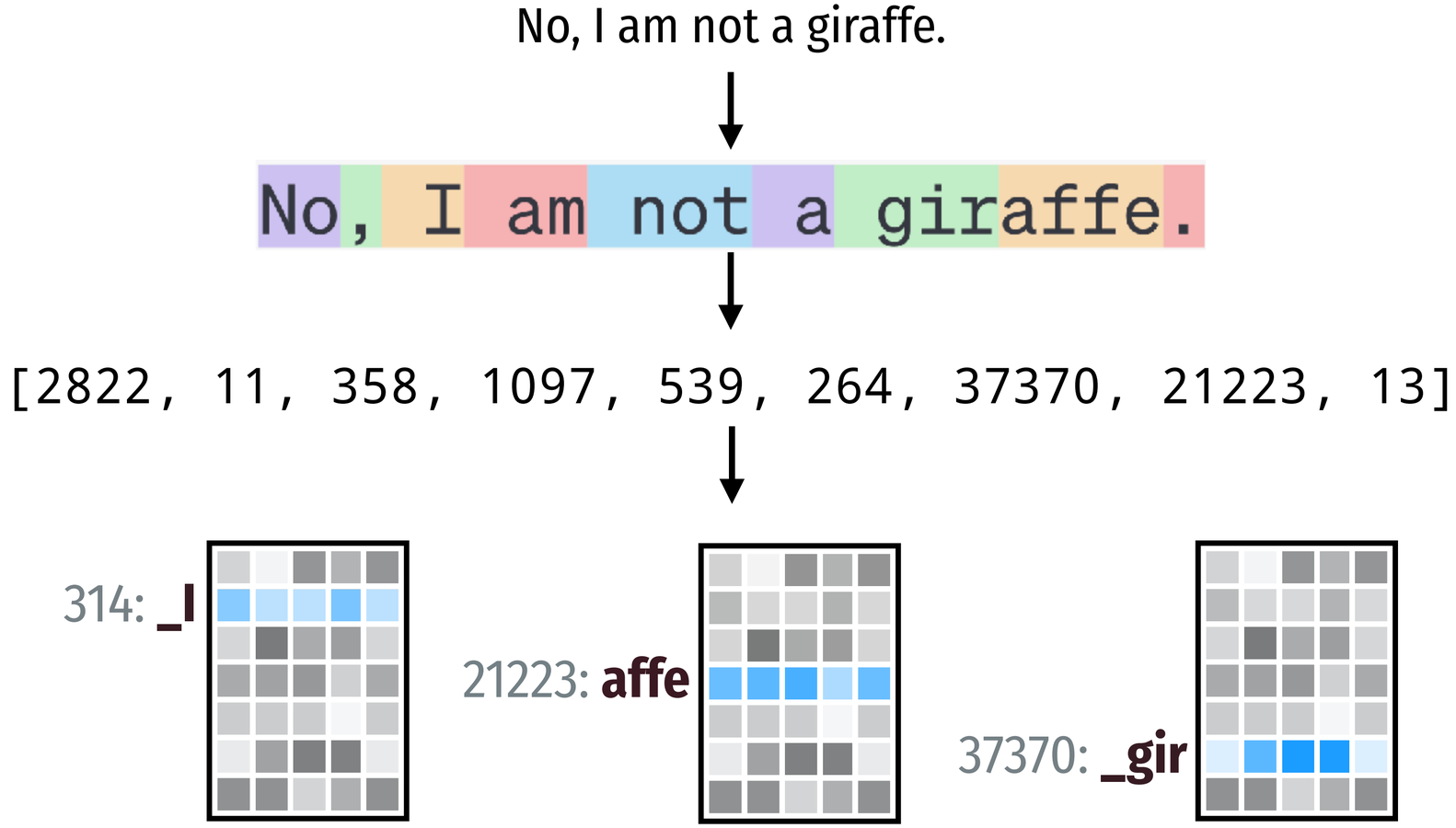
Tokenizing the training data

an example



Tokenizing the training data

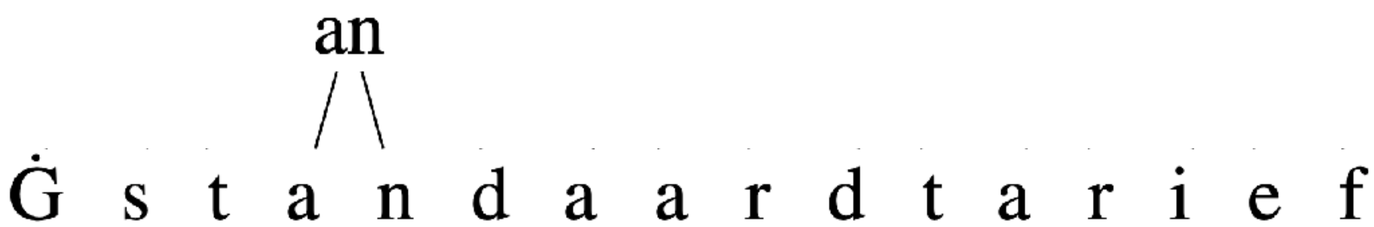
an example



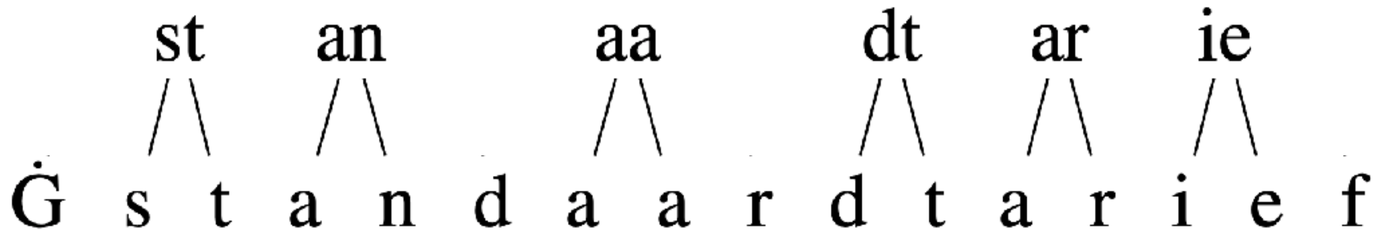
Byte-pair encoding (BPE) merges frequent tuples

Ġ s t a n d a a r d t a r i e f

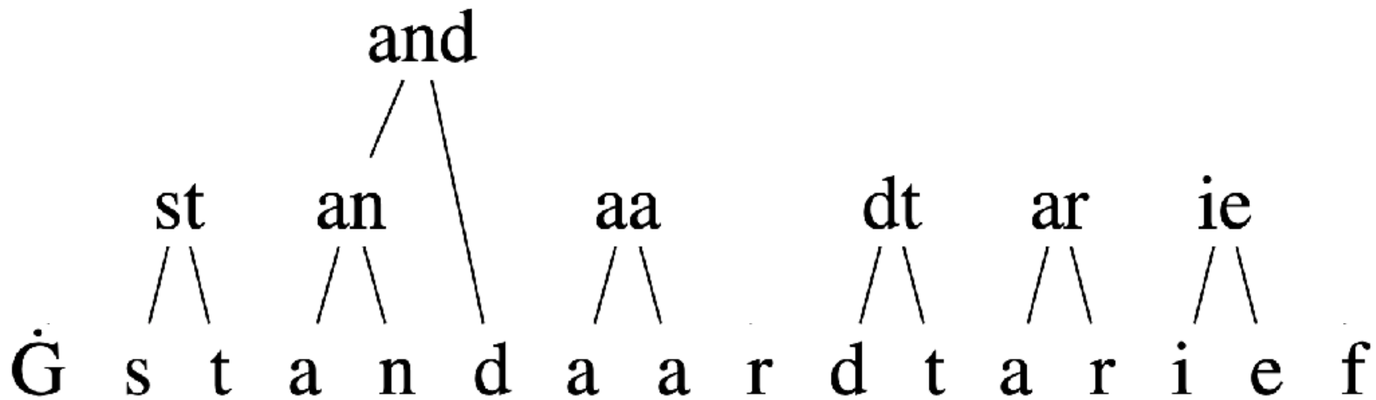
Byte-pair encoding (BPE) merges frequent tuples



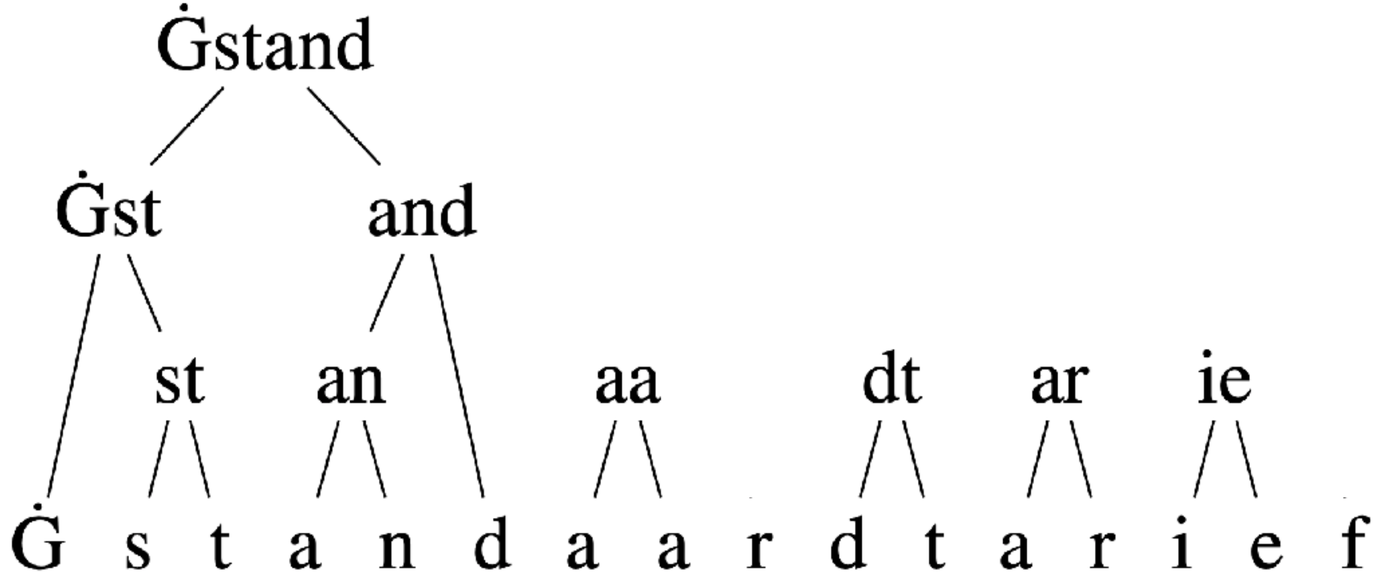
Byte-pair encoding (BPE) merges frequent tuples



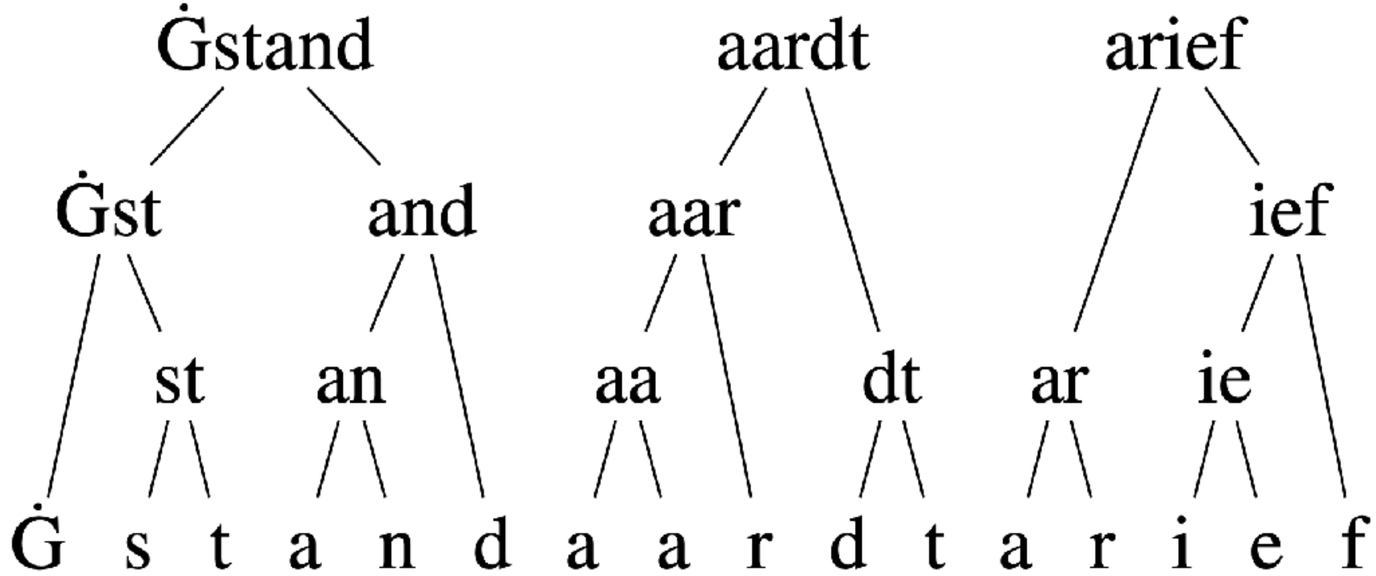
Byte-pair encoding (BPE) merges frequent tuples



Byte-pair encoding (BPE) merges frequent tuples



Byte-pair encoding (BPE) merges frequent tuples

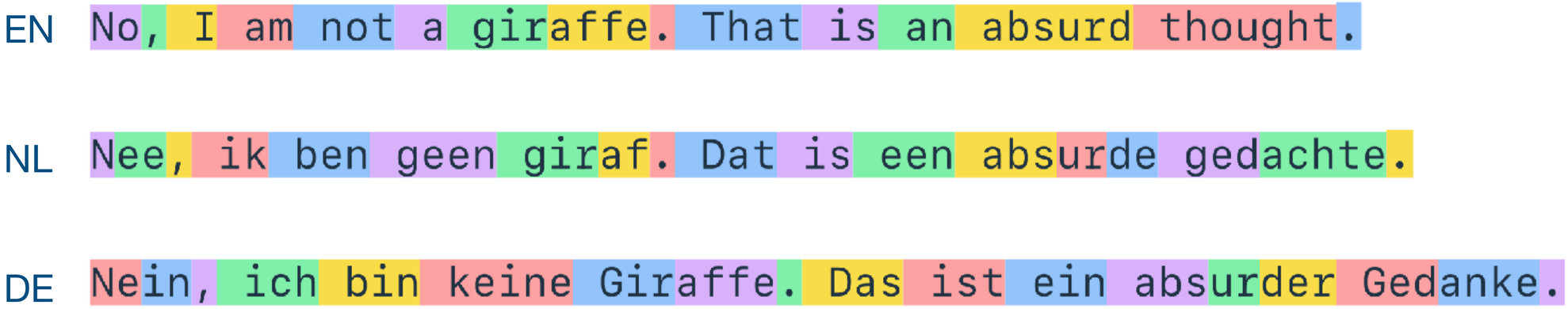


Few non-English words are tokens

Token types for words in English do not match, so the tokenizer falls back to non-representative tokens types.

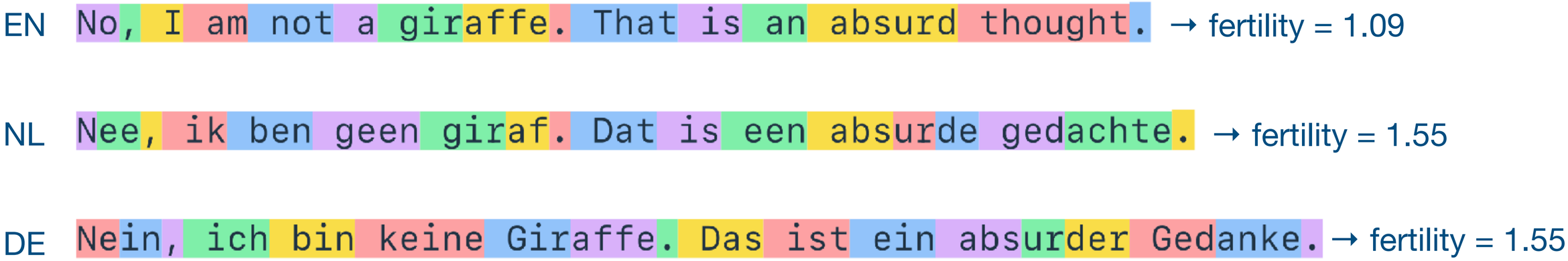
Few non-English words are tokens

Token types for words in English do not match, so the tokenizer falls back to non-representative tokens types.



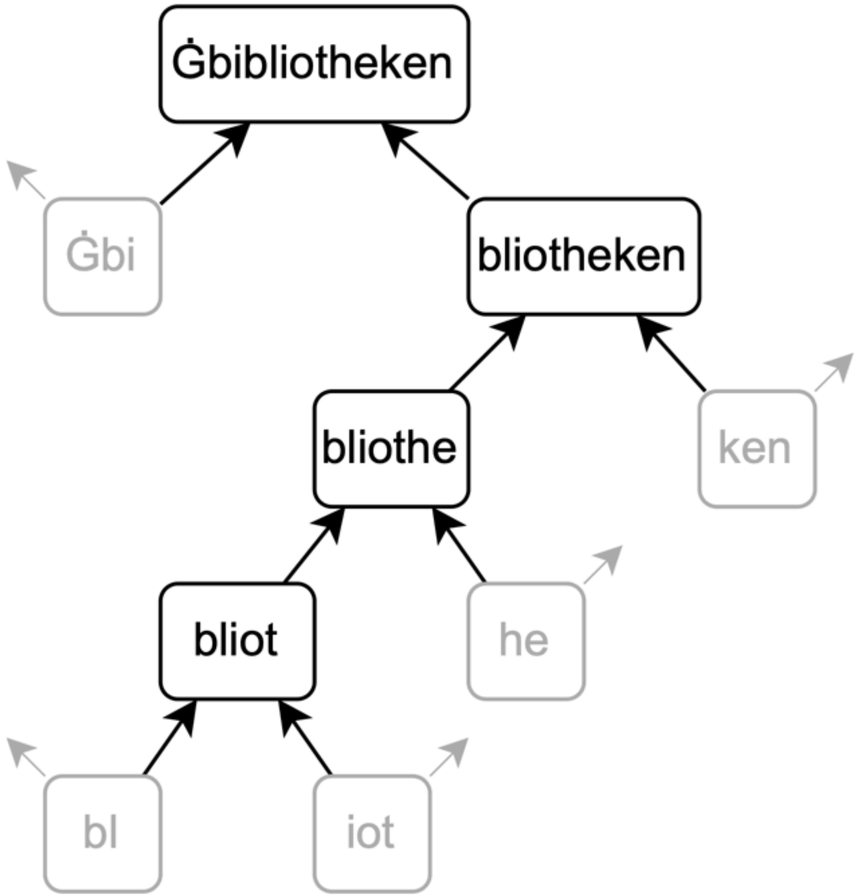
Few non-English words are tokens

Token types for words in English do not match, so the tokenizer falls back to non-representative tokens types.



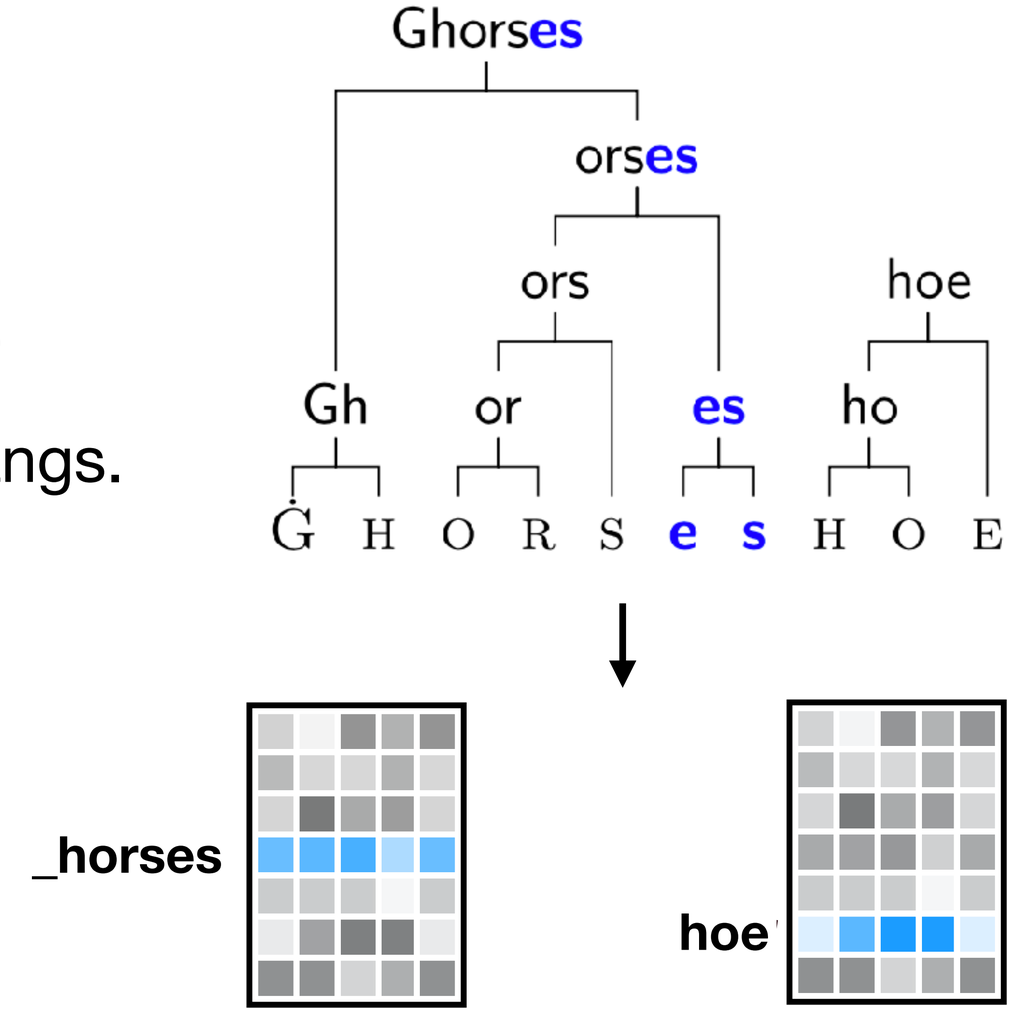
Chain effect: vocabulary is filled with unused tokens

- BPE merges tuples of 2 token types
- Token types need to exist first



… and morpheme boundaries are not respected

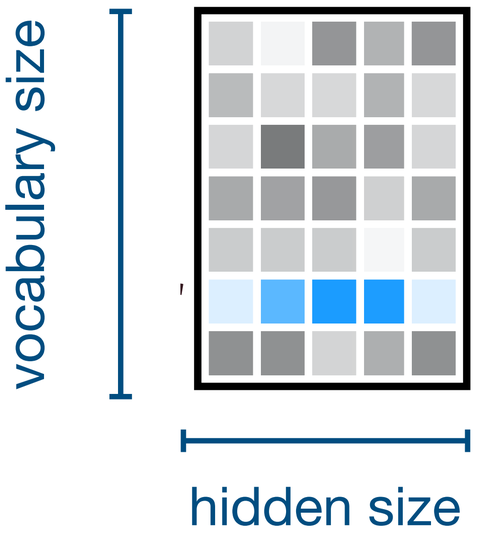
- Tokenization happens *eagerly*
- Representations are dependent on tokens
- Problematic for agglutinative or fusional langs.



Language-specific models

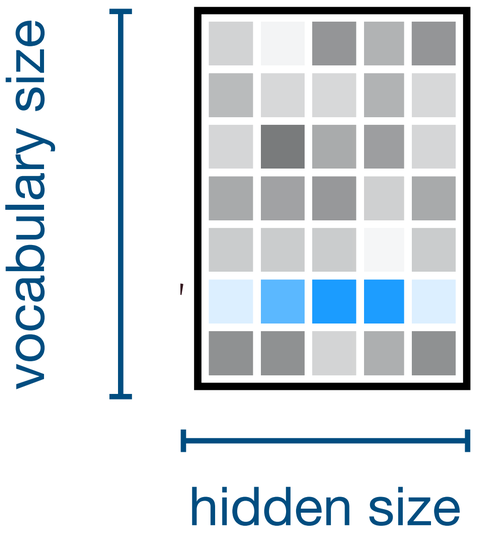
Why?

- Vocabulary size is limited
  - Not every word in every language can be a token
  - Embeddings need to be stored (16bits x vocab x h)



Why?

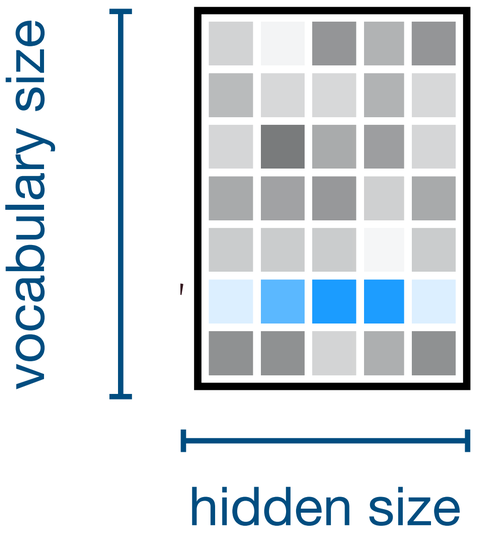
- Vocabulary size is limited
  - Not every word in every language can be a token
  - Embeddings need to be stored (16bits x vocab x h)
- But we want to represent our target domain well
  - More meaningful representations
  - Lower fertility = faster inference



Why?

- Vocabulary size is limited
  - Not every word in every language can be a token
  - Embeddings need to be stored (16bits x vocab x h)
- But we want to represent our target domain well
  - More meaningful representations
  - Lower fertility = faster inference

→ **domain-specific or language-specific models**



Geitje-7b

First Dutch LLM

Geitje-7b

First Dutch LLM that got taken down by Brein

- Mistral-7b finetune on ‘gigacorpus’
- A torrent with gigabytes of Dutch books
- Gigacorpus got taken down by Brein already

Ontwikkelaar haalt taalmodel GEITje offline na verzoek Stichting Brein - update

Het Nederlandse AI-taalmodel GEITje is offline gehaald op 'dringend verzoek' van Stichting Brein. GEITje zou volgens Brein deels getraind zijn op documenten uit de dienst Library Genesis, die afgelopen zomer is geblokkeerd.

Brein zegt dat het model is getraind met tienduizenden Nederlandstalige boeken die afkomstig zijn uit een illegale bron, namelijk Library Genesis, die afgelopen zomer op verzoek van Brein is geblokkeerd door Nederlandse accessproviders. De illegaal verkregen documenten en e-books waren waarschijnlijk terug te vinden in Gigacorpus, de dataset die afgelopen zomer door de maker zelf offline is gehaald. Gigacorpus bevatte naast boeken ook andere Nederlandstalige data, zoals wetsartikelen en uitspraken van Rechtspraak.nl.

"Brein is niet tegen het trainen van AI, maar vindt wel dat de auteurs van al die muziek, boeken etc. daarvoor een eerlijke vergoeding moeten krijgen. Indien de oorspronkelijke makers niet willen dat hun materiaal voor het trainen van AI wordt gebruikt, dan moet dat ook gerespecteerd worden", schrijft de stichting.

De ontwikkelaar van GEITje verweerde dat tekstdatamining is toegestaan voor wetenschappelijke doeleinden en dat het model door wetenschappers wordt gebruikt, volgens Brein. De stichting wijst er echter op dat het model ook voor commercieel gebruik openbaar werd aangeboden op Huggingface.co. "De AI Act schrijft voor dat wetenschappers rechtmatig toegang moeten hebben tot materiaal om het te mogen gebruiken voor het trainen van AI. Dat is niet het geval als bij het trainen van een model gebruik is gemaakt van evident illegale bronnen", aldus Brein.

GEITje-maker Edwin Rijgersberg, op Tweakers bekend als E_Rijgersberg, bevestigt in een eigen post dat het taalmodel eind 2023 getraind is op gedeelten van het Nederlandse Gigacorpus. Brein heeft tegen Rijgersberg gezegd dat volgens de geldende wet- en regelgeving GEITje daarom offline gehaald moet worden.

ChocoLlama

More effort to curate high-quality data

- OSCAR: 93 GB (28.6B tokens) - Common Crawl dump
- Open Subtitles: 5 GB (1.54B tokens)
- Wikipedia: 2.5 GB (769M tokens)
- Job Descriptions: 1.5 GB (462M tokens) - **TechWolf**
- Staatsblad: 1.4 GB (431M tokens) - **Bizzy**
- Project Gutenberg: 0.3 GB (92M tokens) - 970 books
- Legislation: 0.2 GB (62M tokens) - **ML6**

Meeus, Rathé, Remy, Delobelle, Decorte, Demeester.
"ChocoLlama: Lessons Learned From Teaching Llamas Dutch" (2023)

ChocoLlama

More effort to curate high-quality data

- OSCAR: 93 GB (28.6B tokens) - Common Crawl dump
- Open Subtitles: 5 GB (1.54B tokens)
- Wikipedia: 2.5 GB (769M tokens)
- Job Descriptions: 1.5 GB (462M tokens) - **TechWolf**
- Staatsblad: 1.4 GB (431M tokens) - **Bizzy**
- Project Gutenberg: 0.3 GB (92M tokens) - 970 books
- Legislation: 0.2 GB (62M tokens) - **ML6**

Model	ARC	HellaSwag	MMLU	TruthfulQA	Avg.
Llama-3-ChocoLlama-instruct	**0.48**	**0.66**	**0.49**	**0.49**	**0.53**
llama-3-8B-rebatch	0.44	0.64	0.46	0.48	0.51
llama-3-8B-instruct	0.47	0.59	0.47	0.52	0.51
llama-3-8B	0.44	0.64	0.47	0.45	0.5
Reynaerde-7B-Chat	0.44	0.62	0.39	0.52	0.49
Llama-3-ChocoLlama-base	**0.45**	**0.64**	**0.44**	**0.44**	**0.49**
zephyr-7b-beta	0.43	0.58	0.43	0.53	0.49
geitje-7b-ultra	0.40	0.66	0.36	0.49	0.48
ChocoLlama-2-7B-tokentrans-instruct	**0.45**	**0.62**	**0.34**	**0.42**	**0.46**
mistral-7b-v0.1	0.43	0.58	0.37	0.45	0.46
ChocoLlama-2-7B-tokentrans-base	**0.42**	**0.61**	**0.32**	**0.43**	**0.45**
ChocoLlama-2-7B-instruct	**0.36**	**0.57**	**0.33**	**0.45**	**0.43
ChocoLlama-2-7B-base	**0.35**	**0.56**	**0.31**	**0.43**	**0.41**
llama-2-7b-chat-hf	0.36	0.49	0.33	0.44	0.41
llama-2-7b-hf	0.36	0.51	0.32	0.41	0.40

Meeus, Rathé, Remy, Delobelle, Decorte, Demeester.
"ChocoLlama: Lessons Learned From Teaching Llamas Dutch" (2023)

ChocoLlama

More effort to curate high-quality data

- OSCAR: 93 GB (28.6B tokens) - Common Crawl dump
- Open Subtitles: 5 GB (1.54B tokens)
- Wikipedia: 2.5 GB (769M tokens)
- Job Descriptions: 1.5 GB (462M tokens) - **TechWolf**
- Staatsblad: 1.4 GB (431M tokens) - **Bizzy**
- Project Gutenberg: 0.3 GB (92M tokens) - 970 books
- Legislation: 0.2 GB (62M tokens) - **ML6**

Model	ARC	HellaSwag	MMLU	TruthfulQA	Avg.
Llama-3-ChocoLlama-instruct	**0.48**	**0.66**	**0.49**	**0.49**	**0.53**
llama-3-8B-rebatch	0.44	0.64	0.46	0.48	0.51
llama-3-8B-instruct	0.47	0.59	0.47	0.52	0.51
llama-3-8B	0.44	0.64	0.47	0.45	0.5
Reynaerde-7B-Chat	0.44	0.62	0.39	0.52	0.49
Llama-3-ChocoLlama-base	**0.45**	**0.64**	**0.44**	**0.44**	**0.49**
zephyr-7b-beta	0.43	0.58	0.43	0.53	0.49
geitje-7b-ultra	0.40	0.66	0.36	0.49	0.48
ChocoLlama-2-7B-tokentrans-instruct	**0.45**	**0.62**	**0.34**	**0.42**	**0.46**
mistral-7b-v0.1	0.43	0.58	0.37	0.45	0.46
ChocoLlama-2-7B-tokentrans-base	**0.42**	**0.61**	**0.32**	**0.43**	**0.45**
ChocoLlama-2-7B-instruct	**0.36**	**0.57**	**0.33**	**0.45**	**0.43
ChocoLlama-2-7B-base	**0.35**	**0.56**	**0.31**	**0.43**	**0.41**
llama-2-7b-chat-hf	0.36	0.49	0.33	0.44	0.41
llama-2-7b-hf	0.36	0.51	0.32	0.41	0.40

DE TIJD

Computerwetenschappers bouwen Vlaams AI-model ChocoLlama

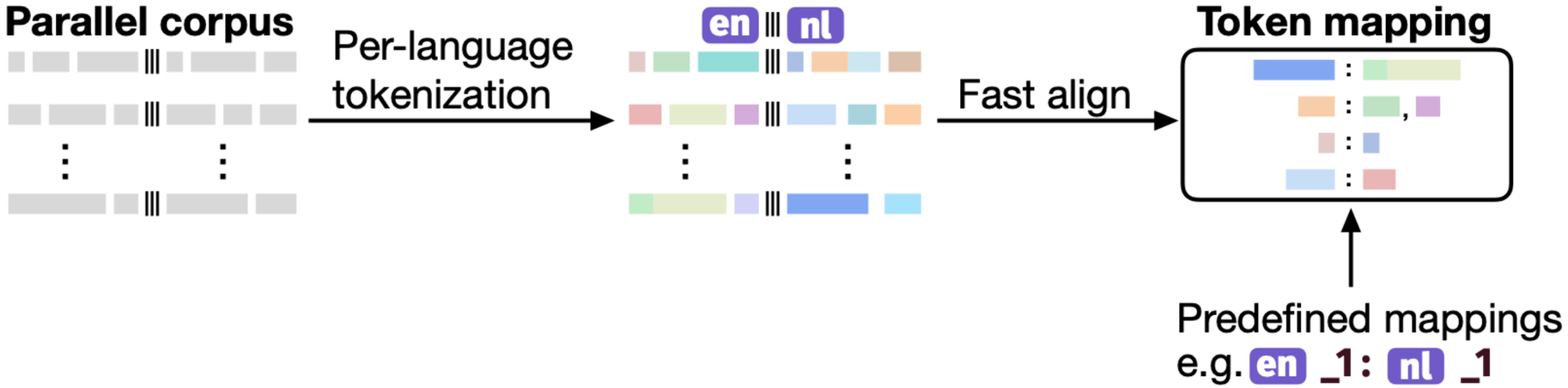
06 februari 2025 16:48

Tweety LLMs

A series of models with language-specific tokenizers

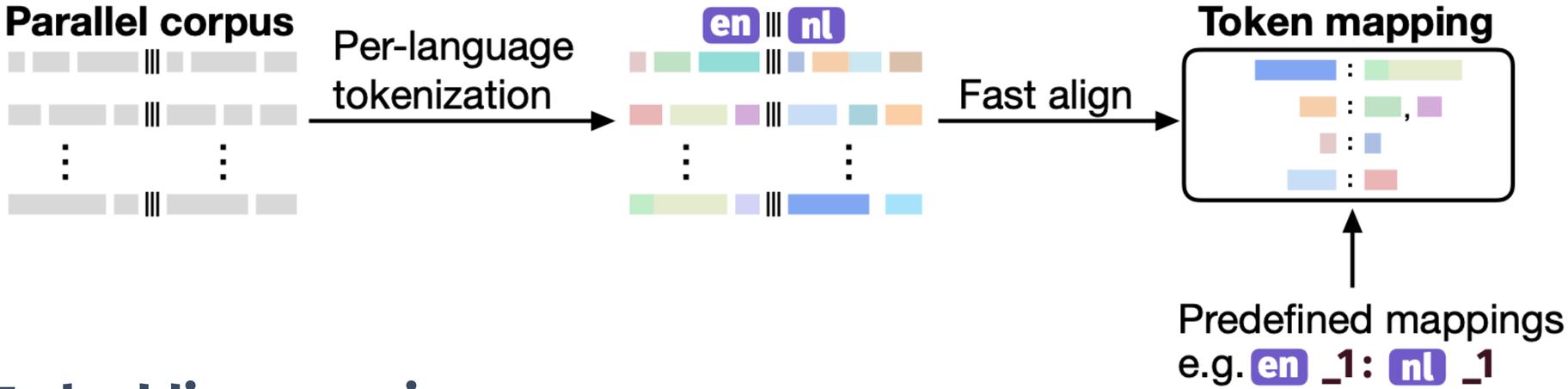
Trans-tokenization

1. Token alignment

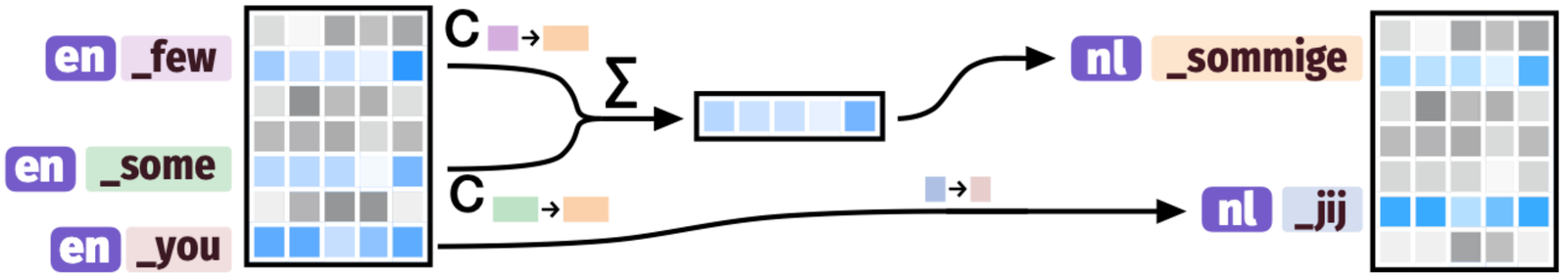


Trans-tokenization

1. Token alignment

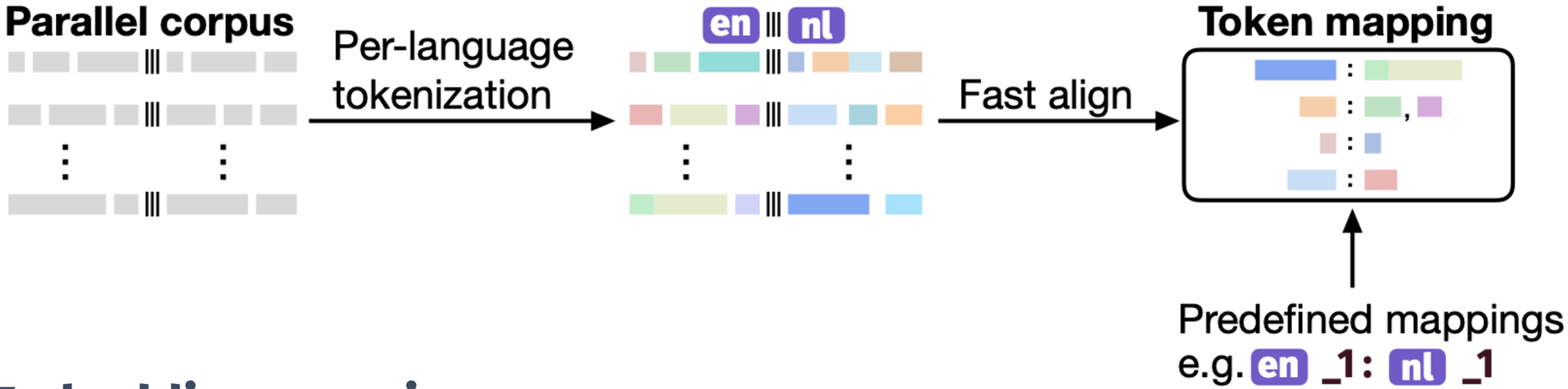


2. Embedding mapping

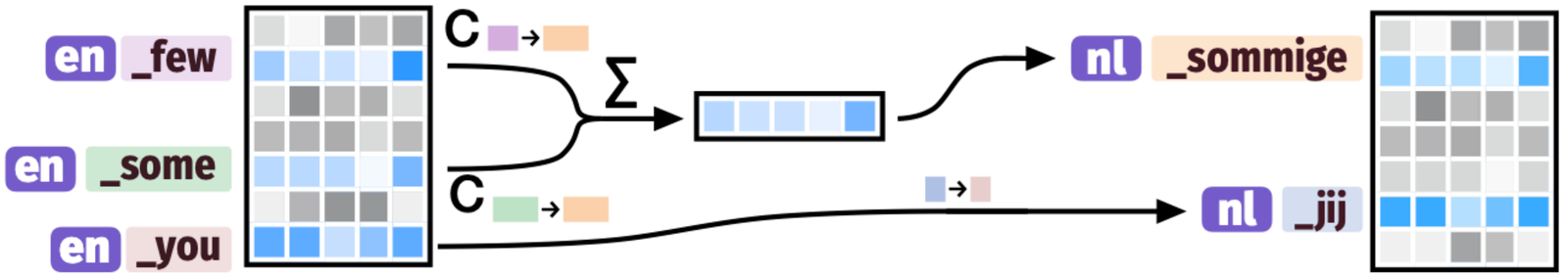


Trans-tokenization

1. Token alignment



2. Embedding mapping



3. **Model adaptation:** continue pretraining for a few GPU hours (e.g. 40h)

tweety-7b-dutch

tweety-7b-tatar

Community model

tweety-7b-italian

github.com/RiTA-nlp

Model	Training tokens	Normalized PPL
mistral-7b-v0.1	6-8T	*9.4*
WECHSEL (Minixhofer et al., 2022)	**+0.4B**	34.3
+ improved Dutch dictionary	**+0.4B**	27.1
FOCUS (Dobler & de Melo, 2023)	**+0.4B**	31.9
tweety-7b-dutch-v24a (ours)	**+0.4B**	11.1
gpt-neo-1.3b-dutch	33B	21.2
mala-500-10b-v2	+30-60B	*18.9*
tweety-7b-dutch-v24a (ours)	+8.5B	**7.7**

Model	Tokenizer		SQuAD-NL ACC		
	Type	$\|\mathcal{V}\|$	0-shot	1-shot	2-shot
mistral-7b-v0.1	English BPE	32 000	**14.3**	21.3	24.2
towerbase-7b-v0.1	English BPE	32 000	13.0	20.9	22.6
gpt-neo-1.3b-dutch	Dutch BPE	50 257	0.0	0.0	0.0
tweety-7b-dutch-v24a (ours)	Dutch BPE	50 257	9.0	**25.8**	**27.6**

tweety-7b-dutch

tweety-7b-tatar

Community model

tweety-7b-italian

github.com/**RiTA-nlp**

Tatar: NLU← and summarization→

Model	Accuracy
Mistral	23.25
Mistral+FT	25.42
MistralRAND	0.00
MistralAVG	17.00
Tweety-7b-tatar-v24a (ours)	**49.34**
Mistral+GTrans	∼44.10

Model	ChrF
Mistral	13.30
Mistral+FT	23.15
MistralRAND	3.79
Tweety-7b-tatar-v24a (ours)	**30.03**
Mistral+GTrans	**30.43**

Hydra LLMs: Switching heads for zero-shot machine translation

Model	Short Text		Long Text		Social Media	
TowerInstruct	17.5	±0.4	13.5	±0.3	17.2	±0.5
TowerInstruct+ParFT	24.5	±0.4	16.5	±0.3	20.6	±0.6
HydraTower+ParFT	39.6	±0.5	18.4	±0.5	33.1	±1.4
HydraTower	47.3	±0.4	32.8	±0.4	39.2	±1.5
HydraTower+BackFT	53.7	±0.2	33.6	±0.3	46.1	±1.4
Google Translate	**55.5**	±0.2	35.3	±0.2	~~63.8~~	±1.8
HydraTower+BackFT+NFR	—	—	**39.2**	±0.6	—	—

European Tweeties

Trans-tokenizing all EU languages

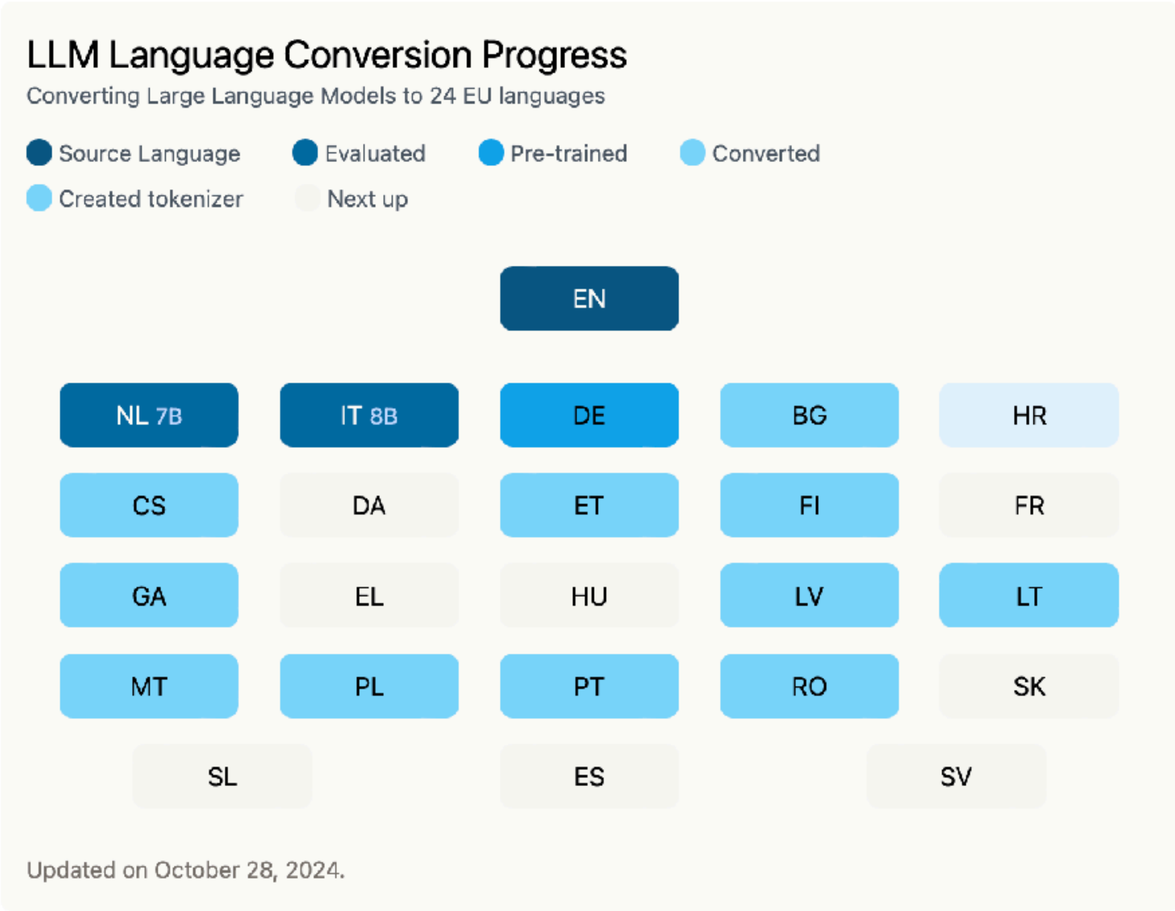
tweety-7b-dutch

tweety-7b-tatar

Community model

tweety-7b-italian

github.com/RiTA-nlp



All our models are publicly available

Model weights on Hugging Face

ChocoLlama/ChocoLlama-2-7B-base
Text Generation • Updated Dec 16, 2024 • 31 • 2

ChocoLlama/ChocoLlama-2-7B-instruct
Text Generation • Updated Dec 16, 2024 • 28 • 2

ChocoLlama/ChocoLlama-2-7B-tokentrans-instruct
Text Generation • Updated Dec 16, 2024 • 21 • 1

ChocoLlama/ChocoLlama-2-7B-tokentrans-base
Text Generation • Updated Dec 16, 2024 • 29

ChocoLlama/Llama-3-ChocoLlama-8B-base
Text Generation • Updated Dec 16, 2024 • 117 • 1

ChocoLlama/Llama-3-ChocoLlama-8B-instruct
Text Generation • Updated Dec 16, 2024 • 83 • 6

Tweeties/tweety-7b-dutch-v24a
Text Generation • Updated Aug 9, 2024 • 1.88k • 13

Tweeties/tweety-tatar-hydra-mt-7b-v24a
Text Generation • Updated Aug 9, 2024 • 13

Tweeties/tweety-tatar-hydra-base-7b-v24a
Text Generation • Updated Aug 9, 2024 • 14

Tweeties/tweety-7b-tatar-v24a
Text Generation • Updated Aug 9, 2024 • 40 • 11

Tweeties/tweety-7b-armenian-v24a
Text Generation • Updated May 27, 2024 • 4 • 1

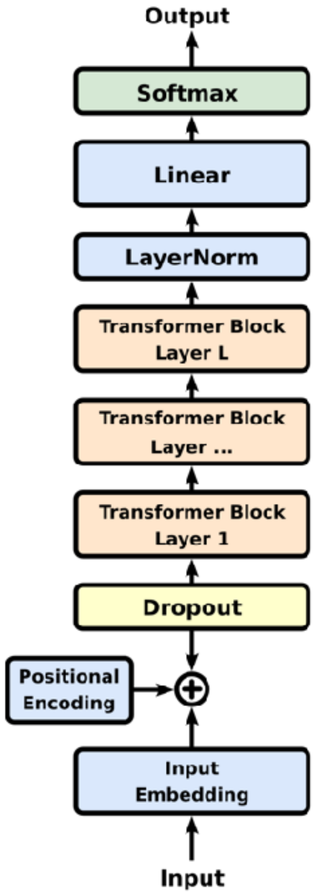
Tweeties/tweety-7b-italian-v24b-llama3 private
Text Generation • Updated May 13, 2024

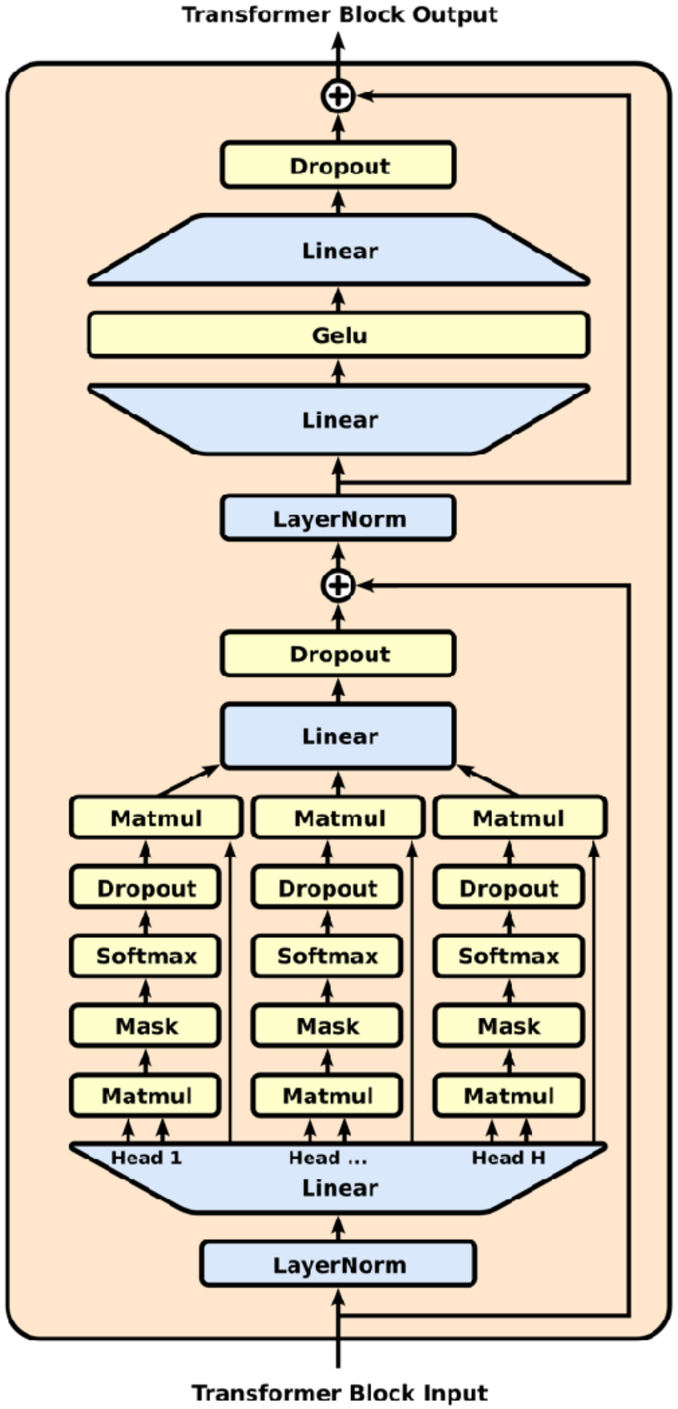
Piet

Inference

An inference pass

through GPT



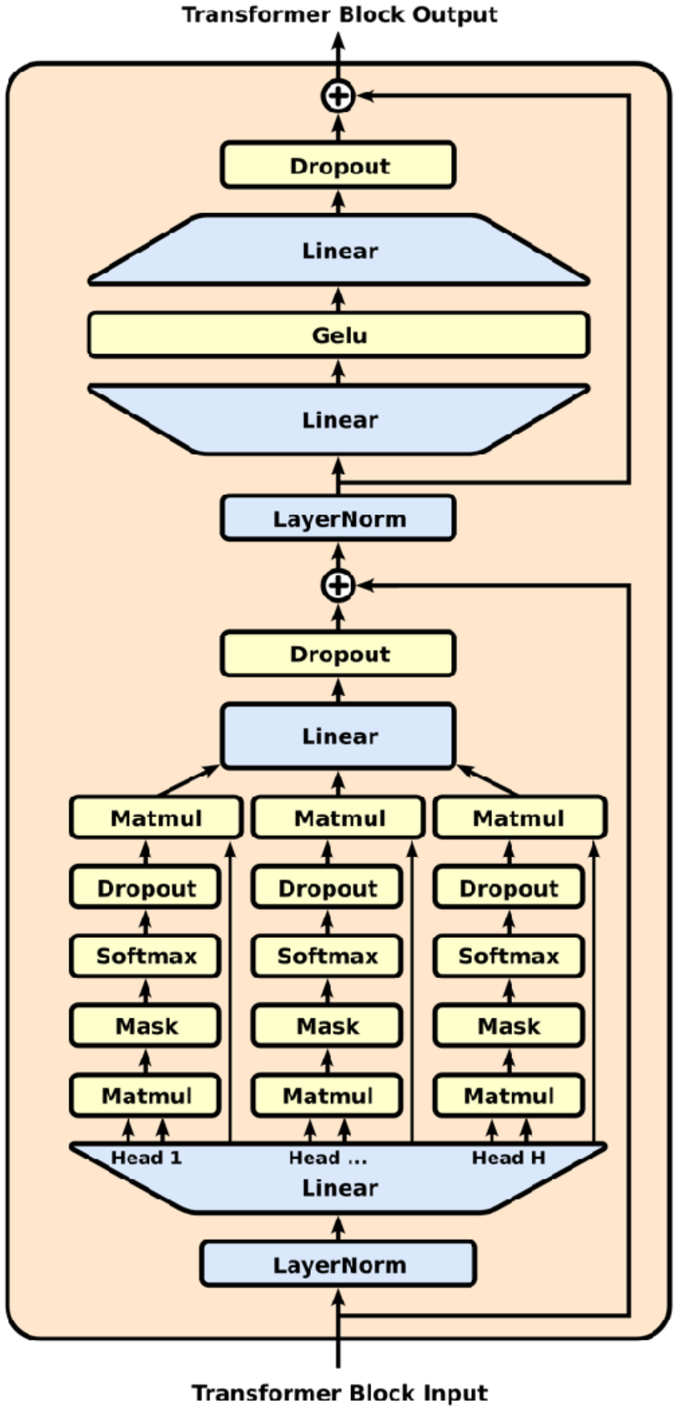


An inference pass

through GPT

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, ..., \text{head}_\text{h})W^O$$
$$\text{where head}_\text{i} = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$$

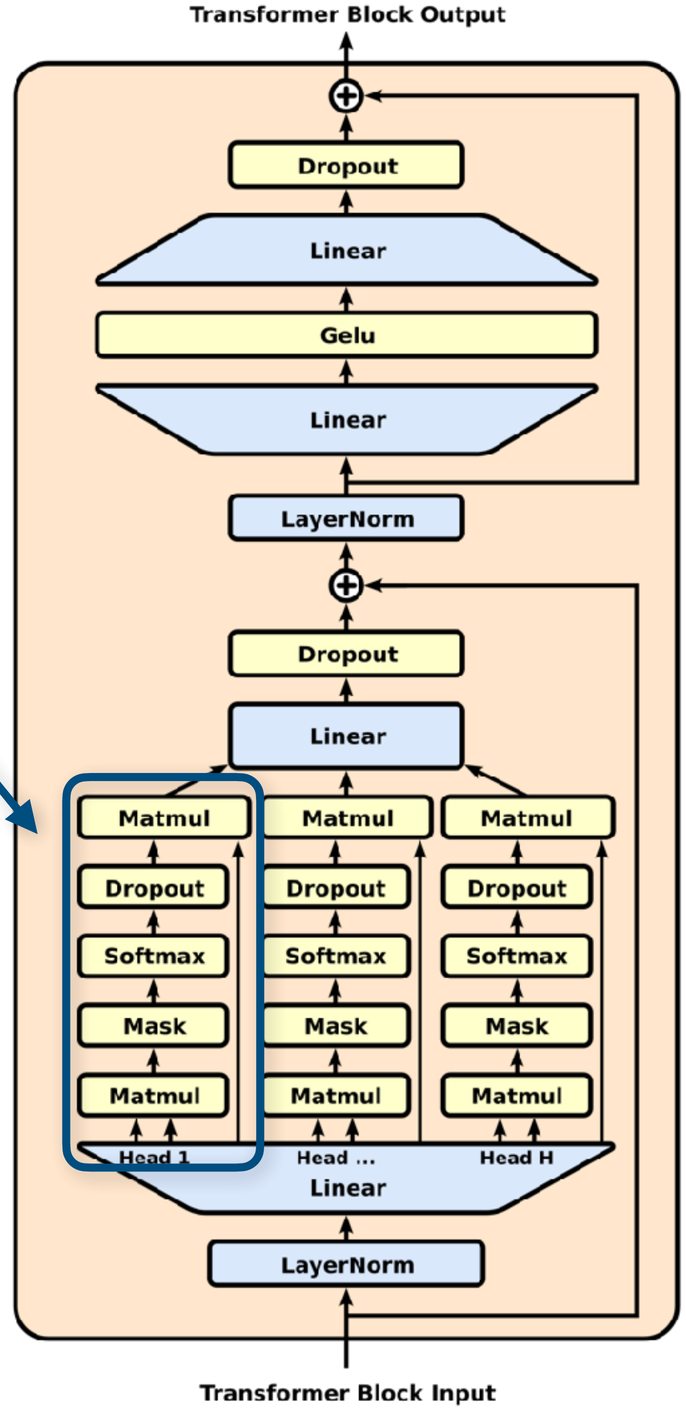


An inference pass

through GPT

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, ..., \text{head}_\text{h})W^O$$
$$\text{where head}_\text{i} = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$$

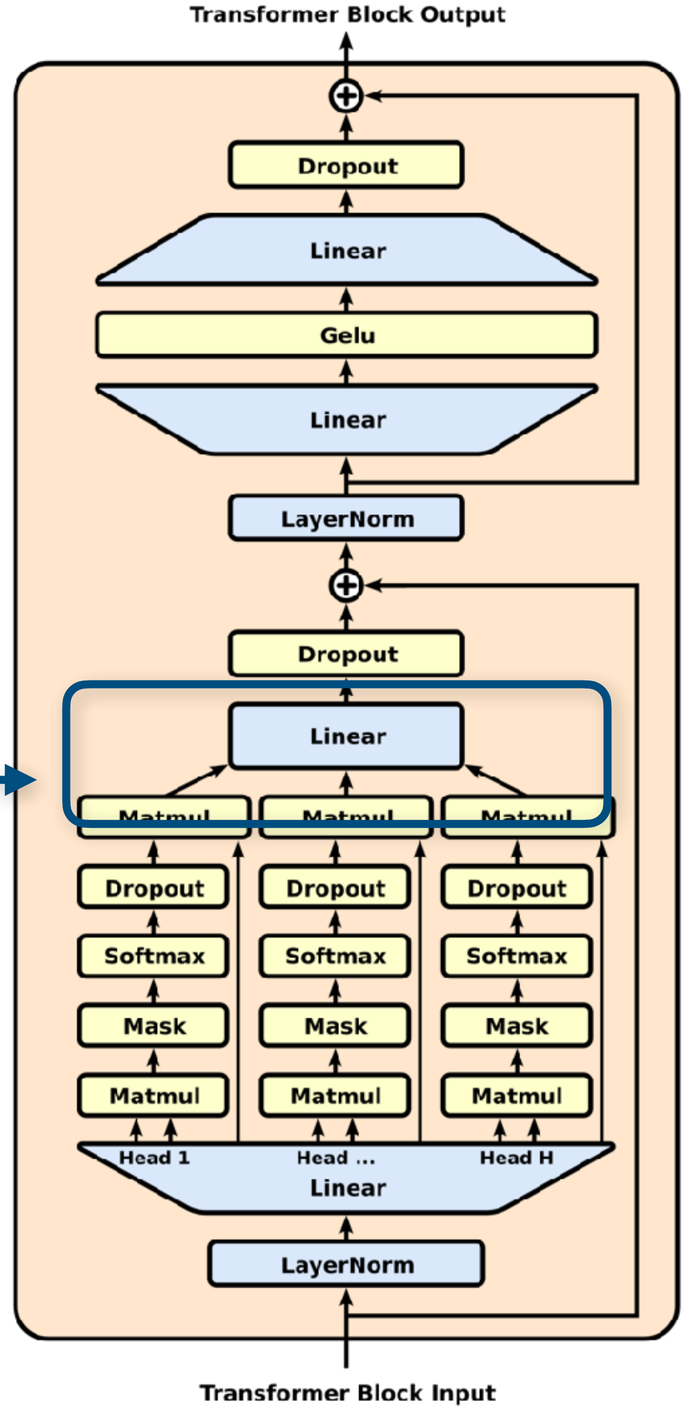


An inference pass

through GPT

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

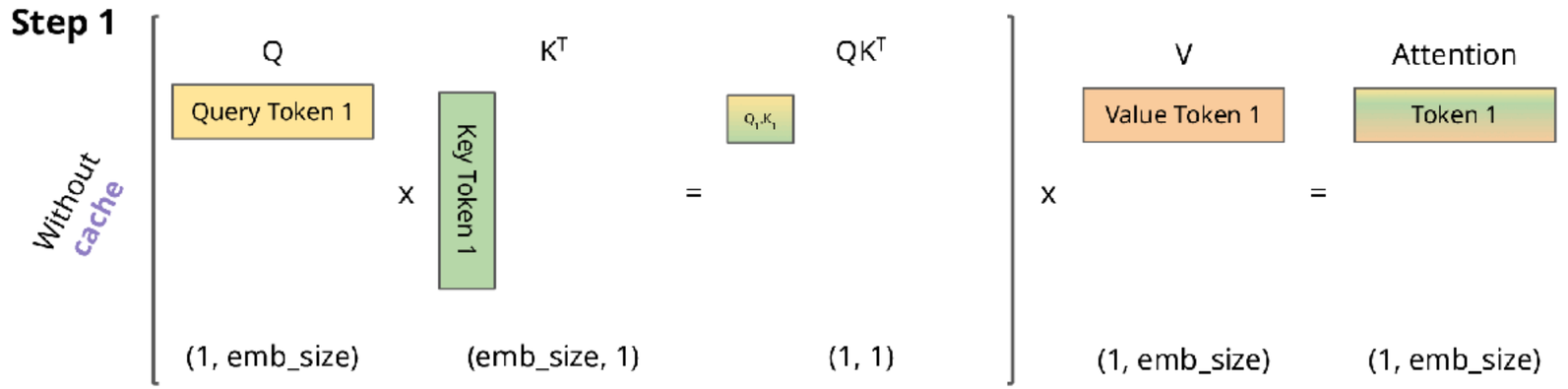
$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, ..., \text{head}_\text{h})W^O$$
$$\text{where head}_\text{i} = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$$

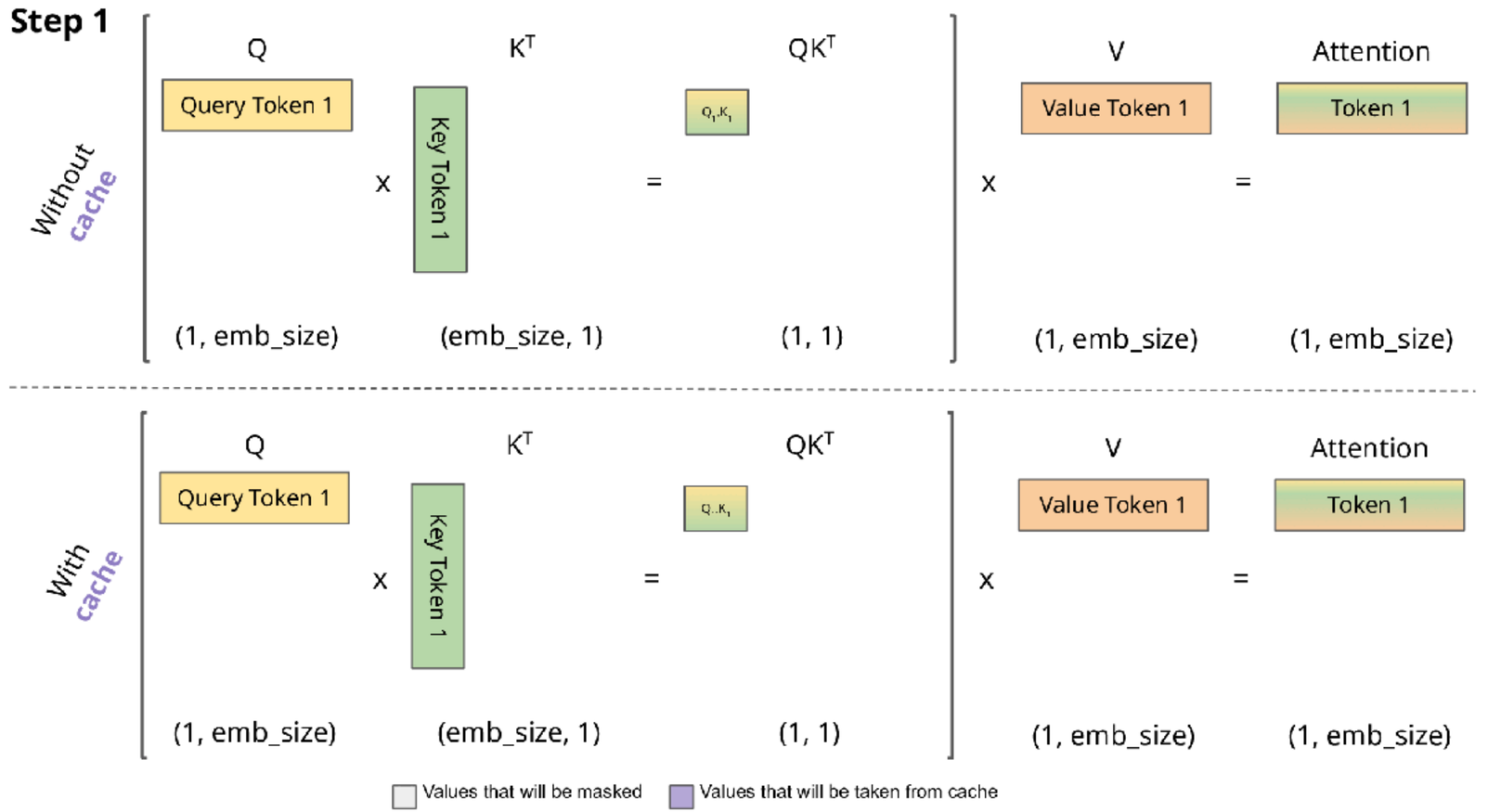


KV cache

- LLM inference is split into 2 steps
  - Prefill
  - Generation
- LLMs are “causal”, conditioned on the previous tokens

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V$$


Step 1
Without cache
Q
K^T
QK^T
Query Token 1
Key Token 1
x
=
$Q_1 \cdot K_1$
V
Attention
Value Token 1
Token 1
x
=
(1, emb_size)
(emb_size, 1)
(1, 1)
(1, emb_size)
(1, emb_size)


Step 1
Without cache
Q
K^T
QK^T
Query Token 1
Key Token 1
x
=
V
Attention
Value Token 1
Token 1
x
=
(1, emb_size)
(emb_size, 1)
(1, 1)
(1, emb_size)
(1, emb_size)
With cache
Q
K^T
QK^T
Query Token 1
Key Token 1
x
=
V
Attention
Value Token 1
Token 1
x
=
(1, emb_size)
(emb_size, 1)
(1, 1)
(1, emb_size)
(1, emb_size)
Values that will be masked
Values that will be taken from cache

Batching: the key to good GPU utilization

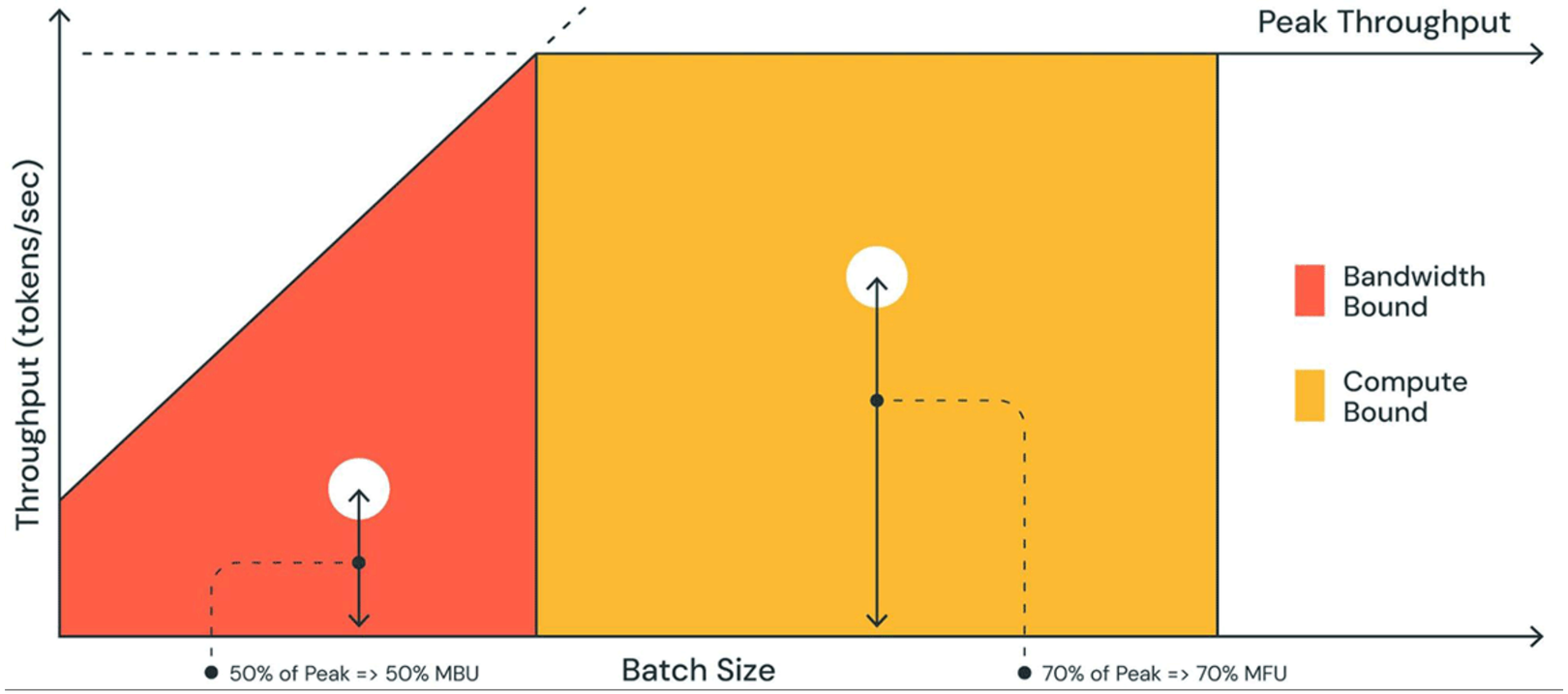
Bandwidth is a limiting factor

For A100

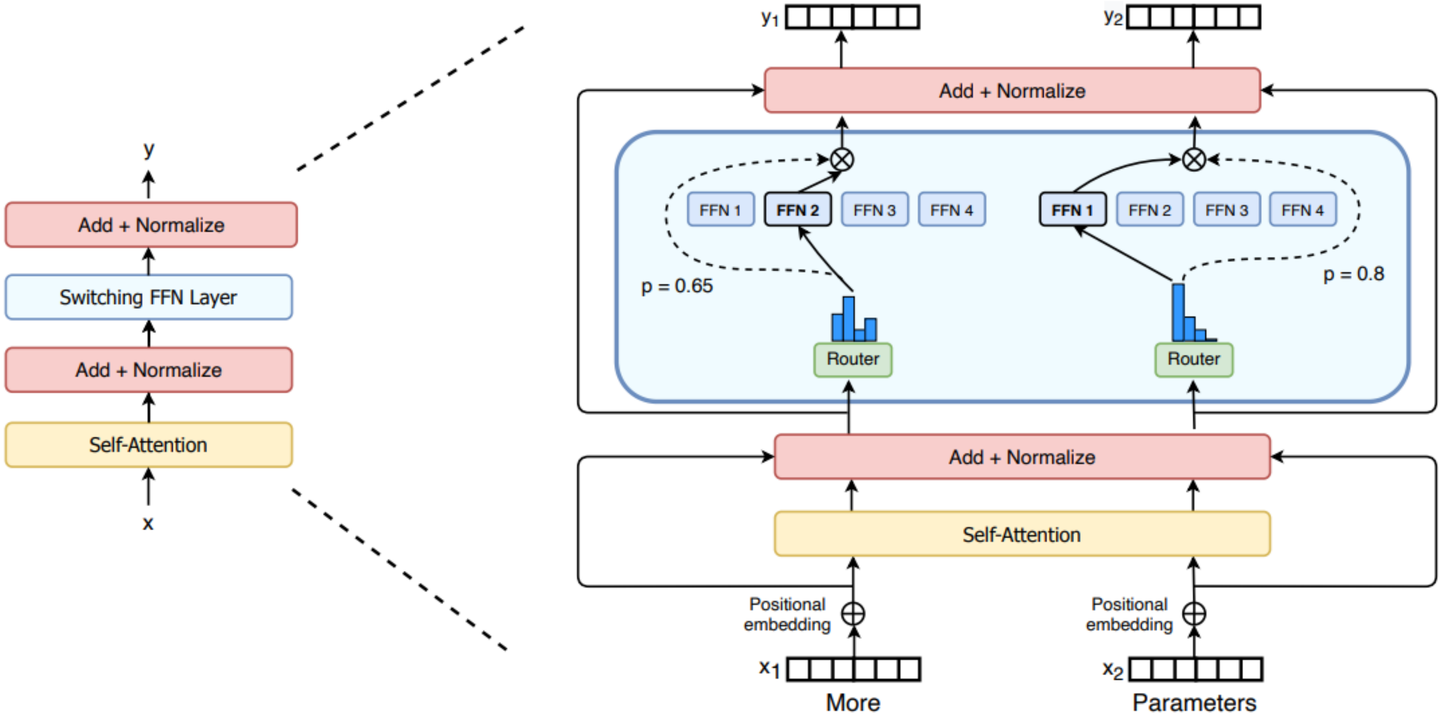
- FLOPS: 3.12 * 10^14 ops
- Memory: 2.03 * 10^12 B/s

Data type	H100-SXM5 (TFLOPS)	A100-SXM4 (TFLOPS)	Difference
TF32	494	156	3.2x
BF16	989	312	3.2x
FP16	989	312	3.2x
FP8	1979	-	6.3x (vs BF16)
Bandwidth (GB/s)	3350	2039	1.6x

The roofline model



Expert parallism



https://arxiv.org/abs/2101.03961

Expert parallism enables deployments at scale

E.g. Deepseek R1 is deployed on 22 nodes

Conclusion

Fast inference at home?

Small batches, local inference

ollama, lmstudio

Multiple users, openAPI-compatible

vLLM, SGLang

Slides available: pieter.ai/appearances.html