UiT The Arctic University of Norway

BIO-AI LAB | ARCTIC LLM WORKSHOP 2023

Large Language Models

Day 1 - Session 1

Introduction to LLM Technology

Himanshu Buckchash

himanshu.buckchash@uit.no

27. Oct 2023

?

give the month name that has six sundays

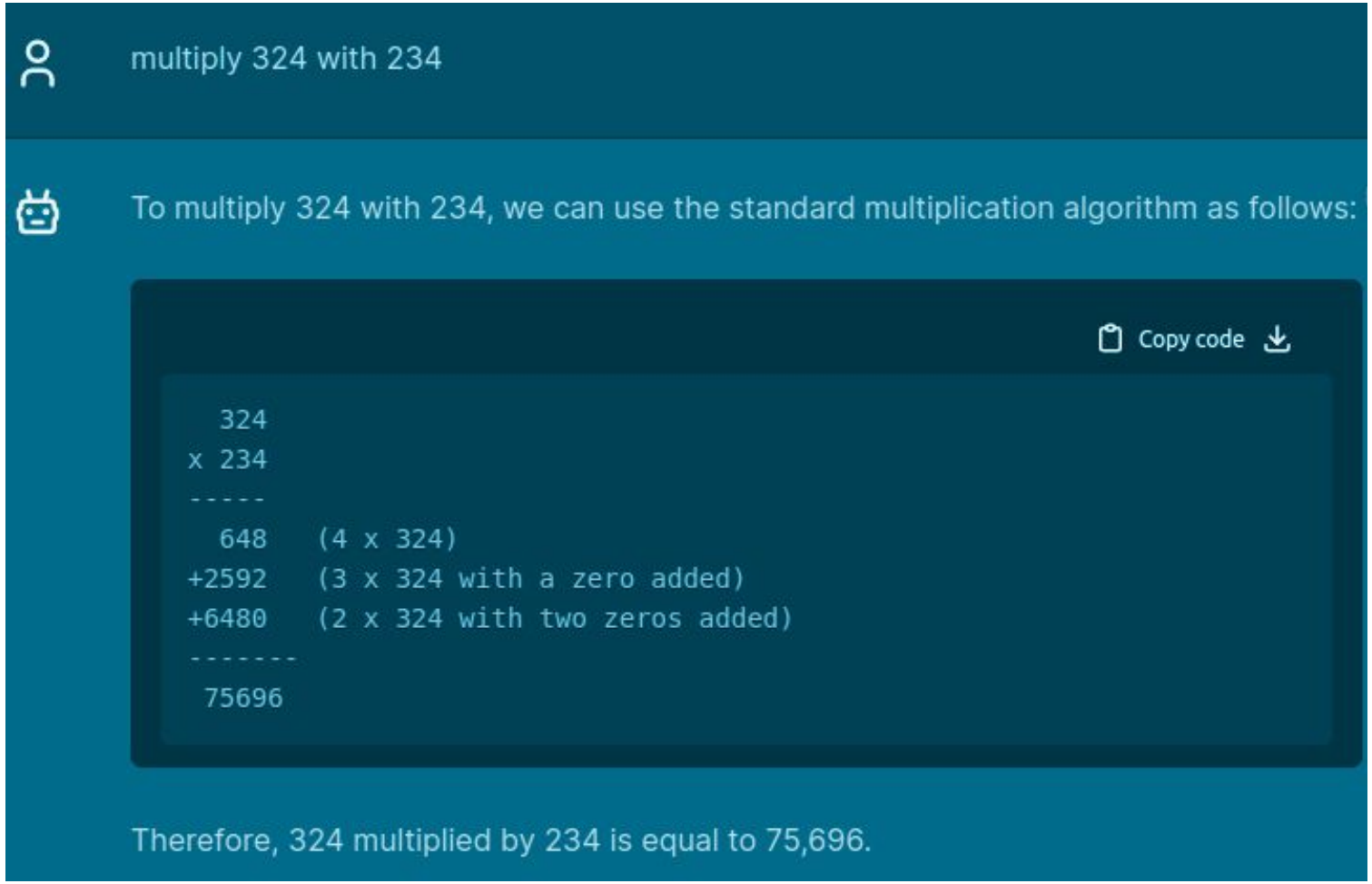
To determine the month that has six Sundays, we need to find a month that has 31 days and starts on a Sunday.

There are only seven possible starting days for a month, and each of them occurs 52 times in a non-leap year. This means that there are 365 days in a year, which is equal to 7 days per week times 52 weeks plus one day.

Therefore, the only month that can have six Sundays is a month that starts on a Sunday and has 31 days.

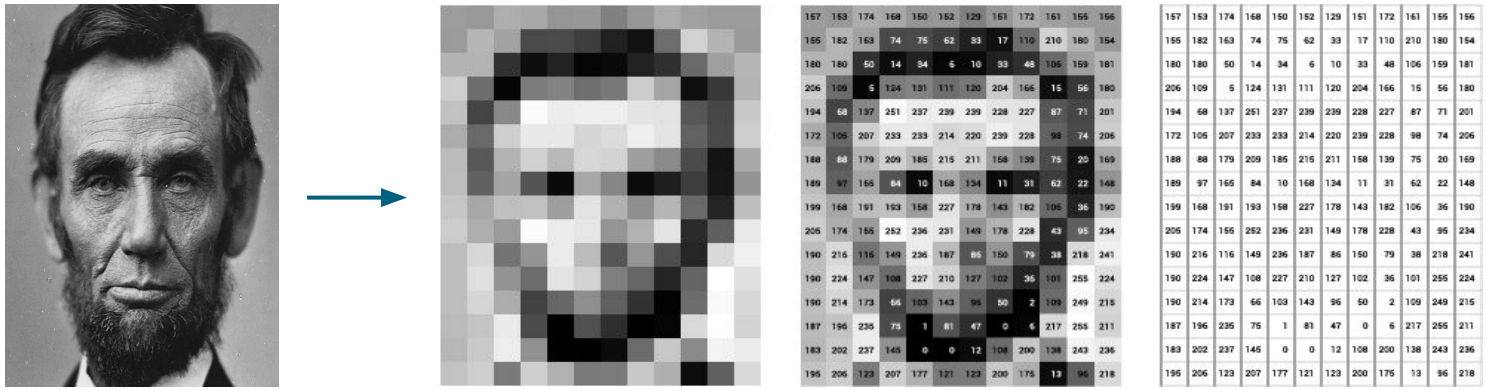
After checking the calendar, we can see that the only month that meets these criteria is **July**.

?



Answer: 75816

Preliminaries: Representation — Numbers



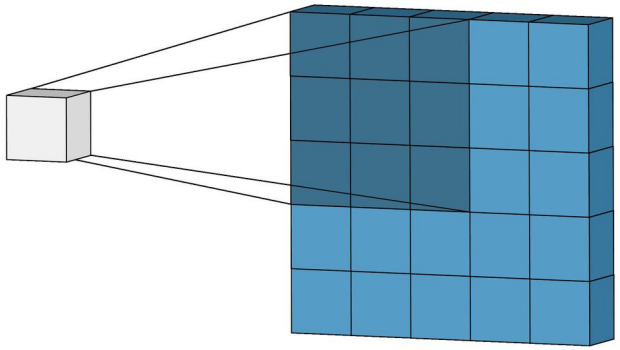
You like this talk → [You / like / this / talk] → [.031, .284, …, .701 / .972, .061, …, .548 / .019, .805, …, .732 / .979, .294, …, .101]

a vector

a matrix

[Image credit] https://vanshsethi.medium.com/convolutional-neural-networks-f40d64453904

Preliminaries: Convolution

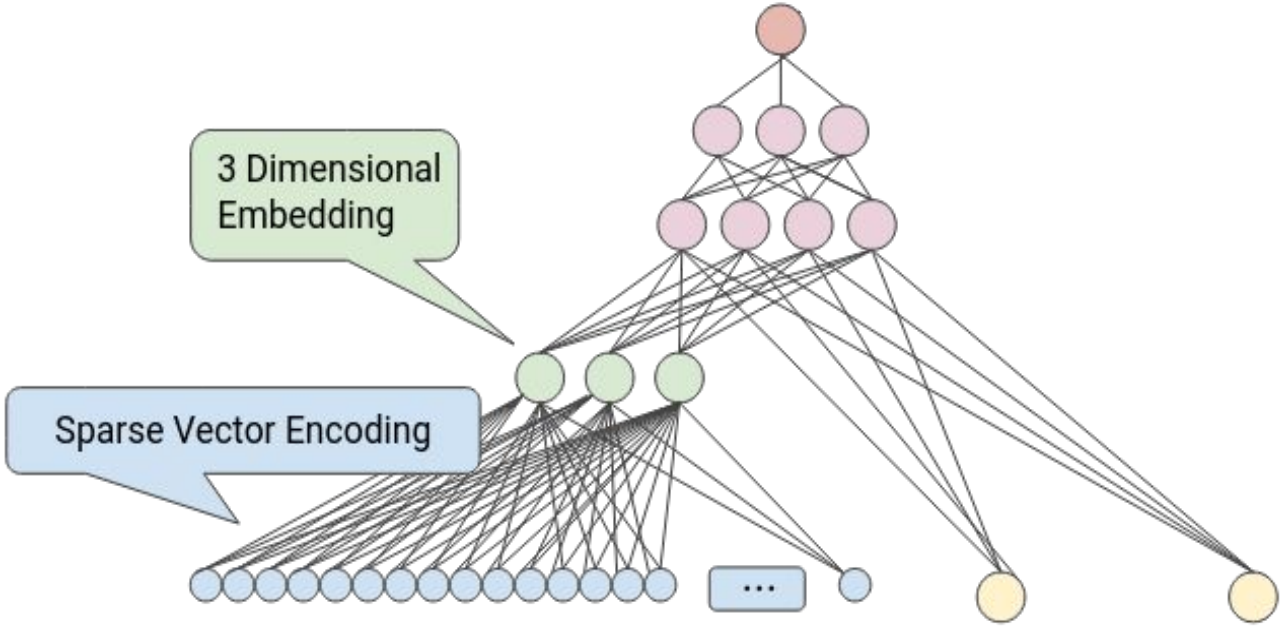


Convolution

Repeated matrix multiplication

[Image credit] https://towardsdatascience.com/convolutions-for-deep-learning-1f6f42faee1

Preliminaries: Embedding



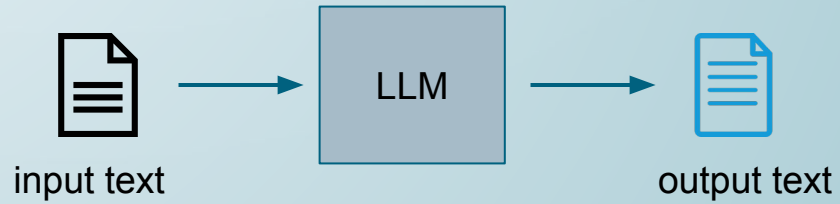
Multi stepped (chained) matrix multiplication

Compression

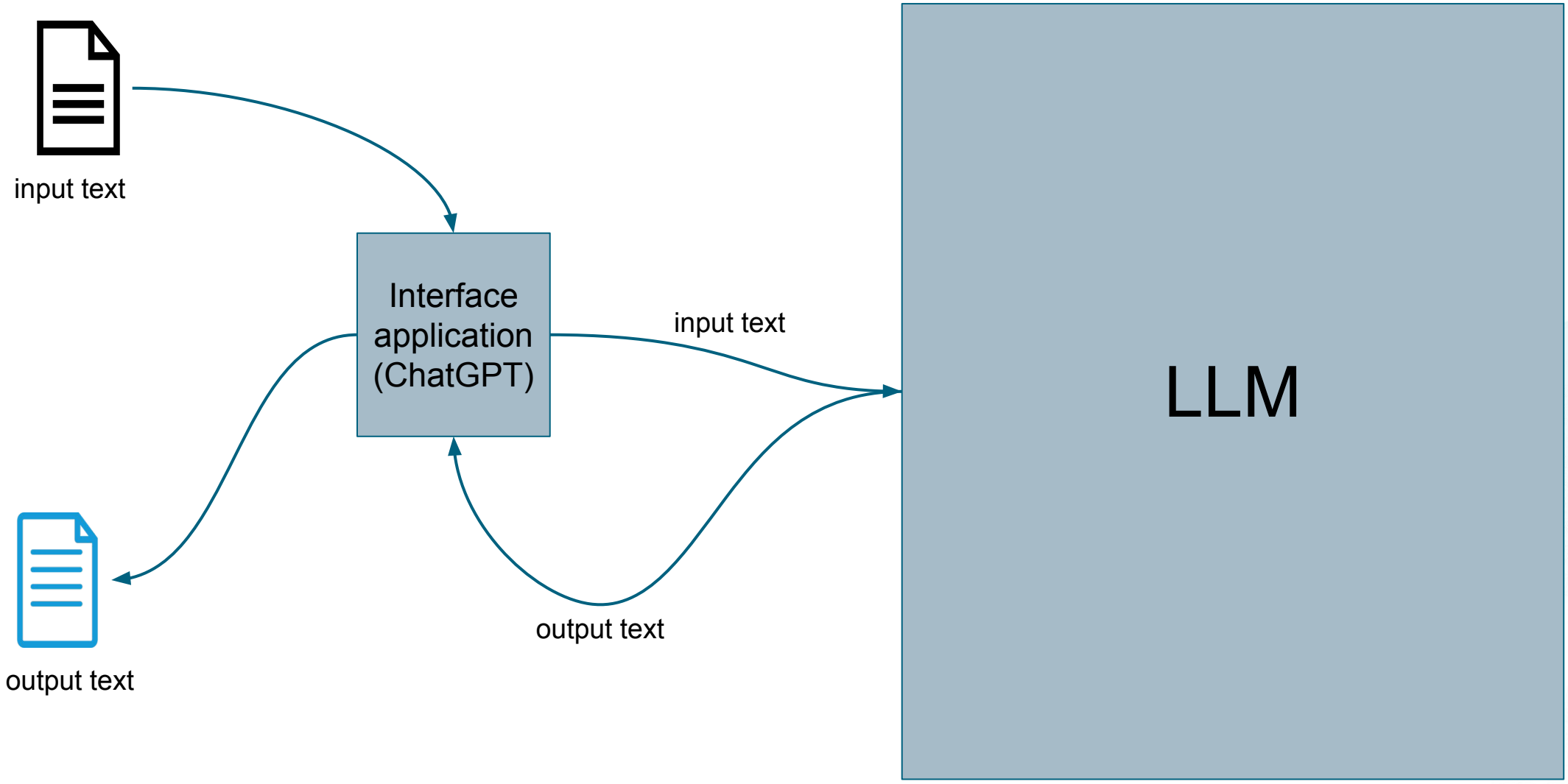
[Image credit] https://medium.com/@evertongomede/embeddings-in-machine-learning-2402bab526fe

SECTION 1

Journey of text through LLM



Journey of text through LLM



Task

Can U expand LLM?  Large Language Modeling

LLM Input preparation

Can U expand LLM?

can u expand llm?

Normalization

Remove accent, spaces, lowercasing

LLM Input preparation

Can U expand LLM?

Normalization

Remove accent, spaces, lowercasing

can u expand llm?

Pre-tokenization

Break at whitespace and punctuation

{can,u,expand,llm,?}

LLM Input preparation

Can U expand LLM?

Normalization

Remove accent, spaces, lowercasing

can u expand llm?

Pre-tokenization

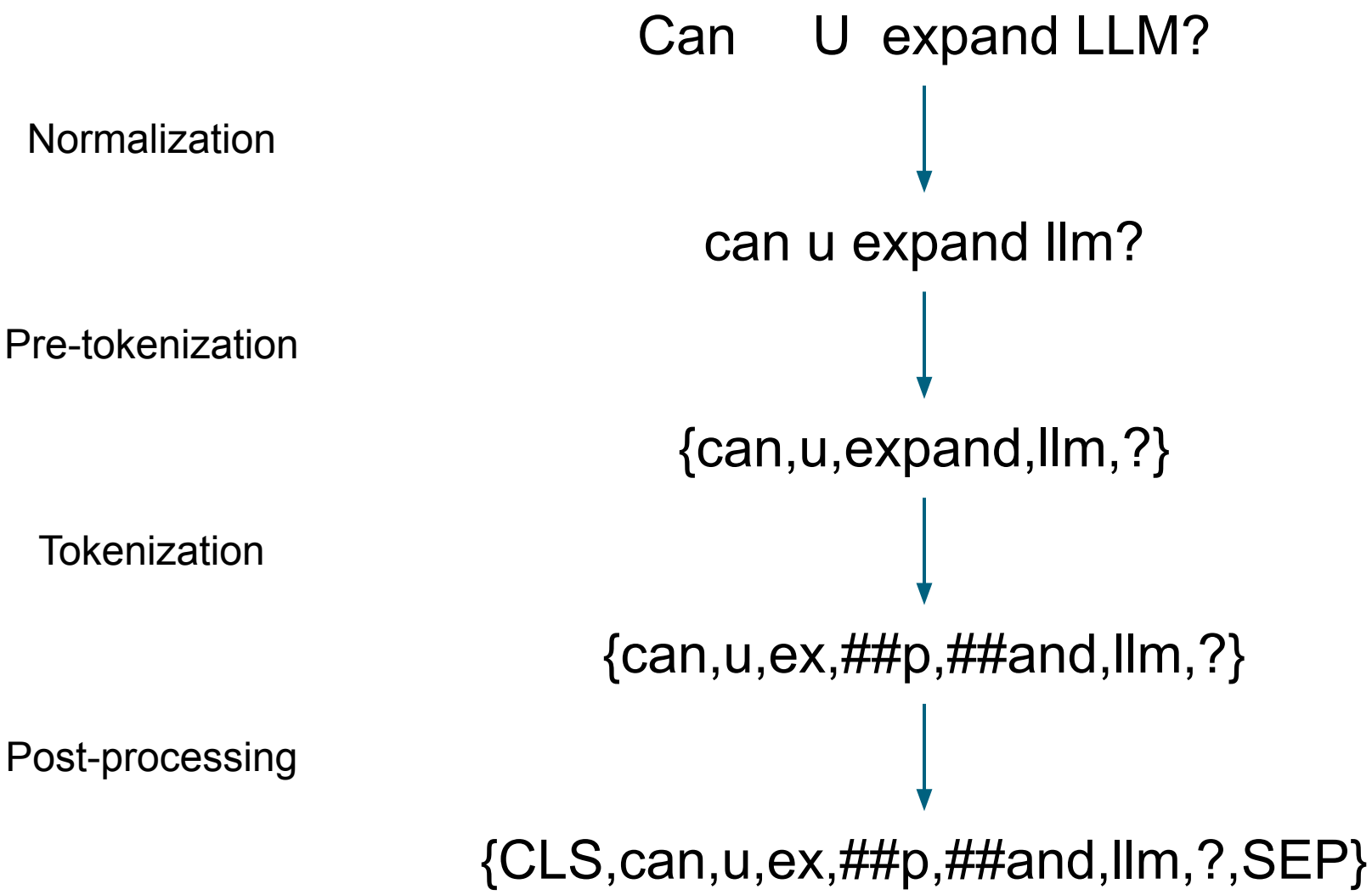
Break at whitespace and punctuation

{can,u,expand,llm,?}

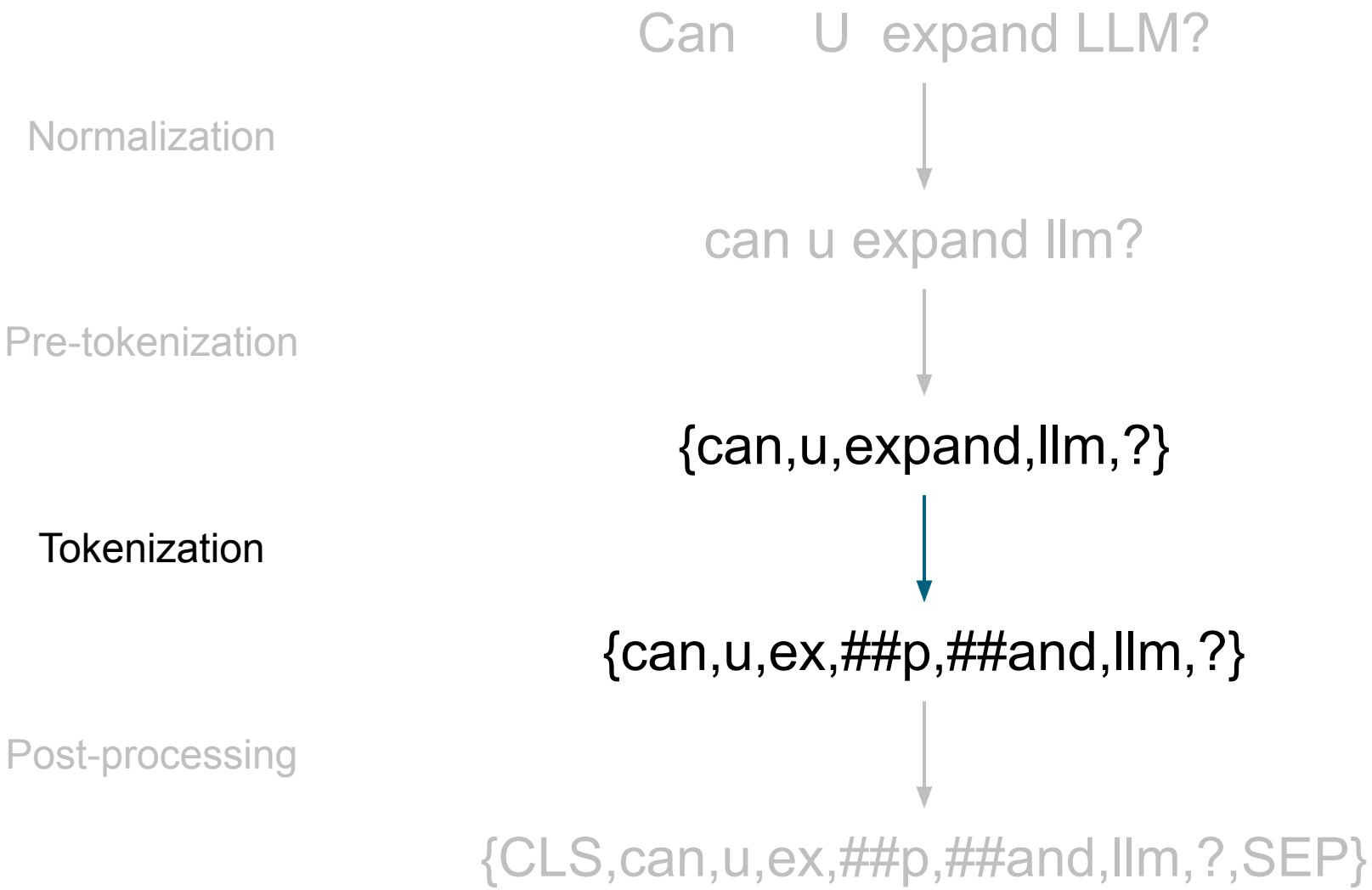
Tokenization

{can,u,ex,##p,##and,llm,?}

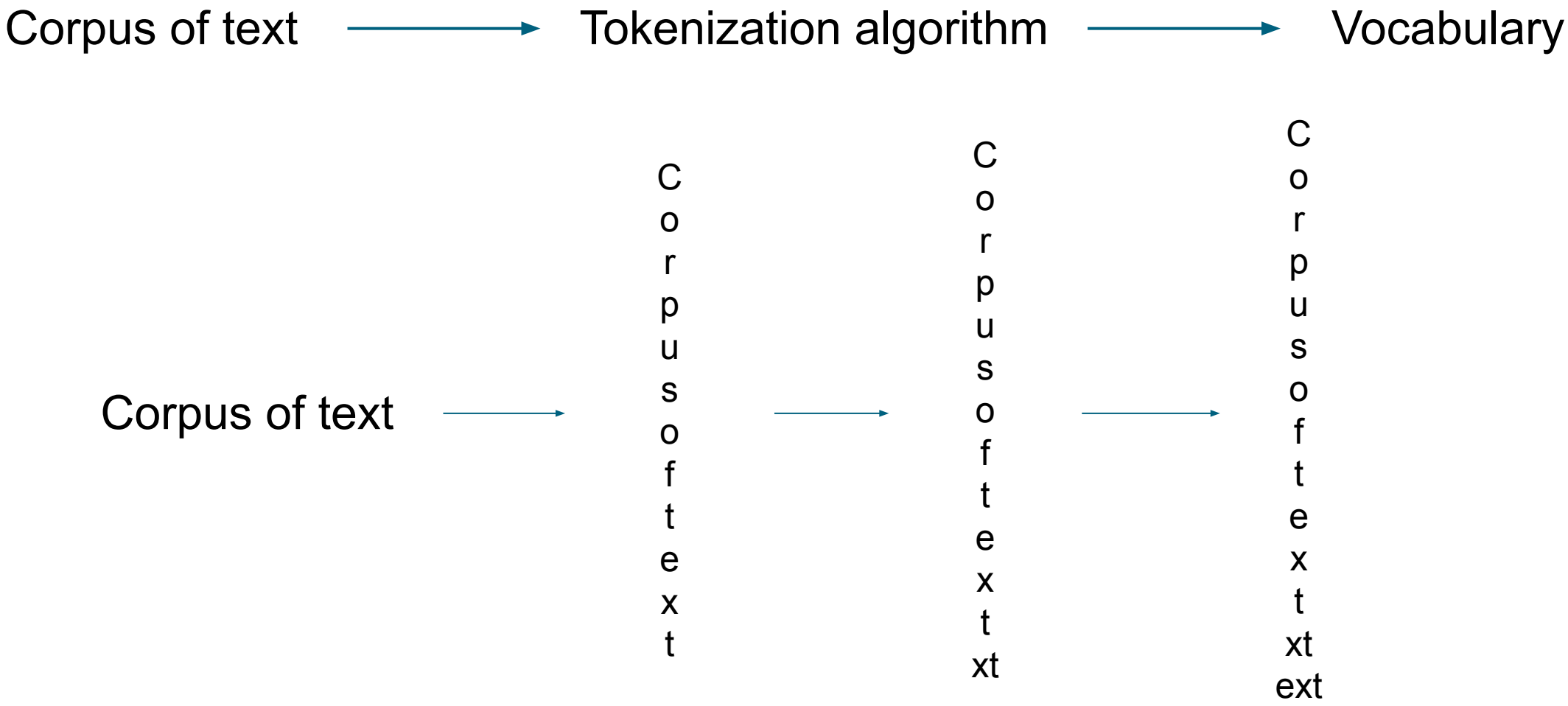
LLM Input preparation



Tokenization: Vocabulary creation



Tokenization: Vocabulary creation



Tokenization algorithms

Byte pair encoding

WordPiece

SentencePiece

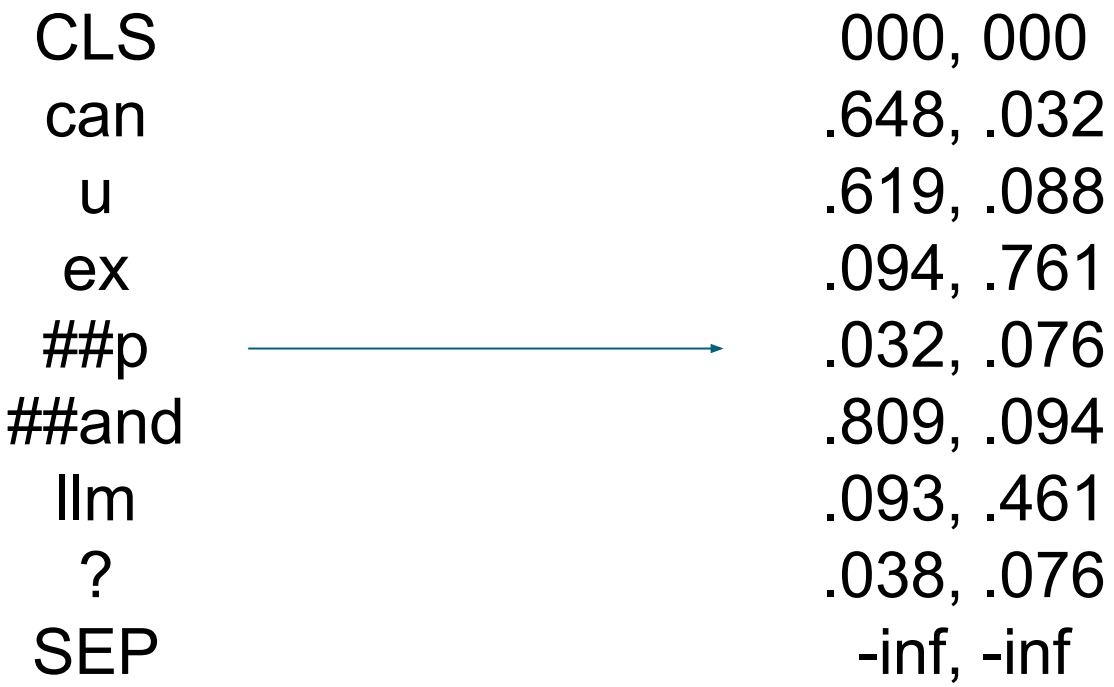
Unigram

Embedding

{CLS,can,u,ex,##p,##and,llm,?,SEP}

{000,.648,.61,.088,.094,.761,.032,.076,.809}

Embedding: fixed length floats

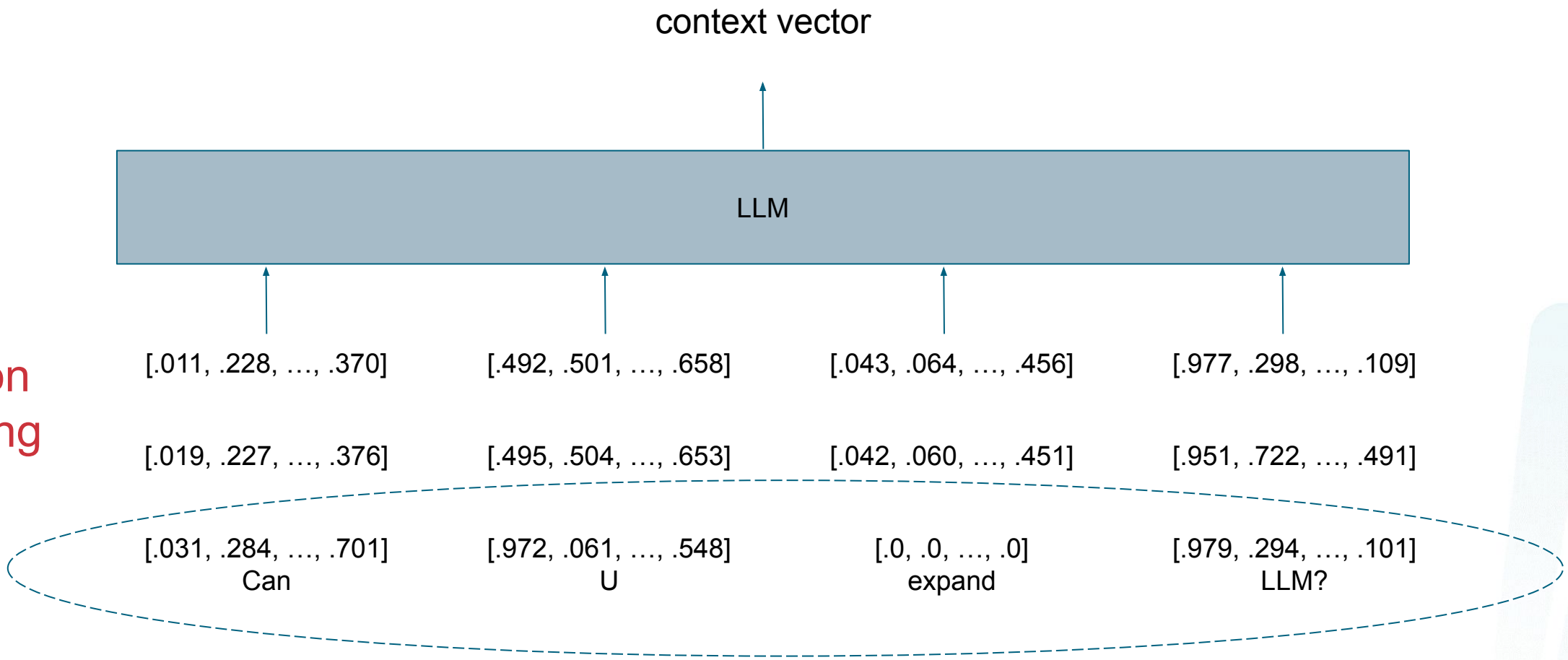


Static vs. Dynamic

Causal Language Modeling (CLM)

Autoregressive Language Inference

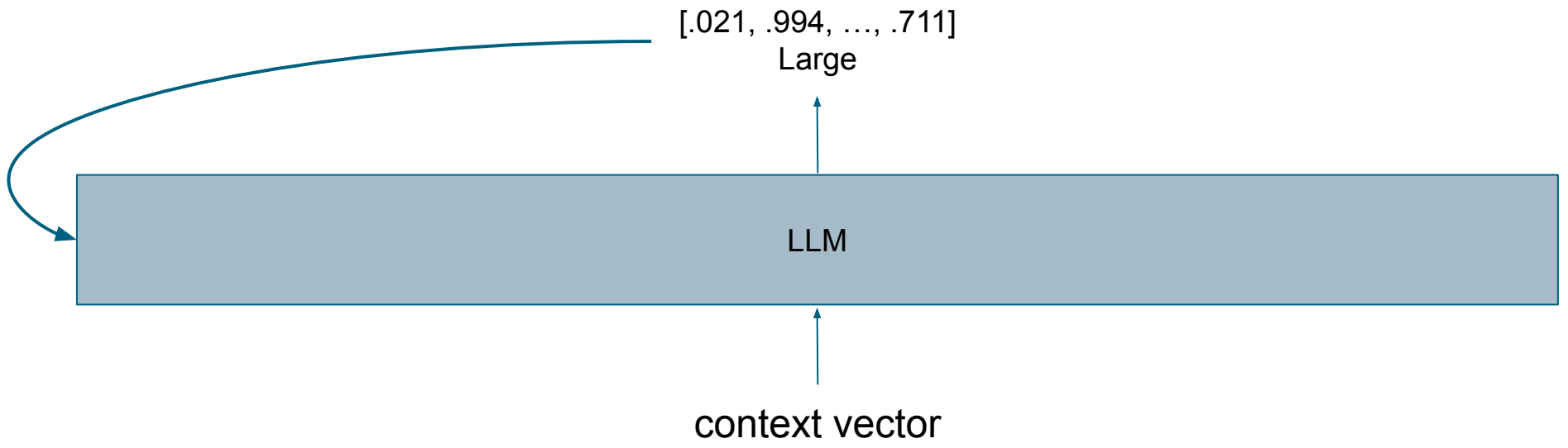
Autoregressive Generation: Causal Language Modeling (CLM)



Context length, Block length, Chunk length

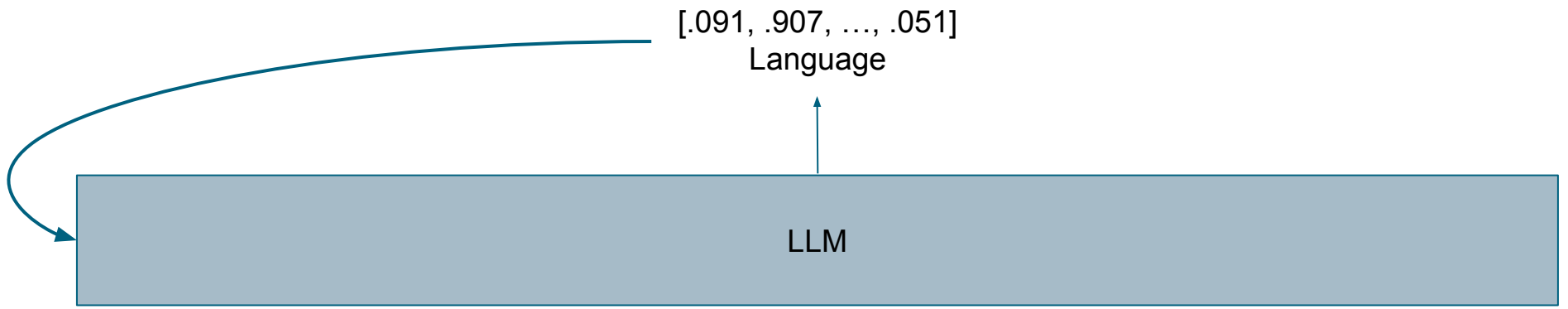
Autoregressive Generation: Causal Language Modeling (CLM)

Output:



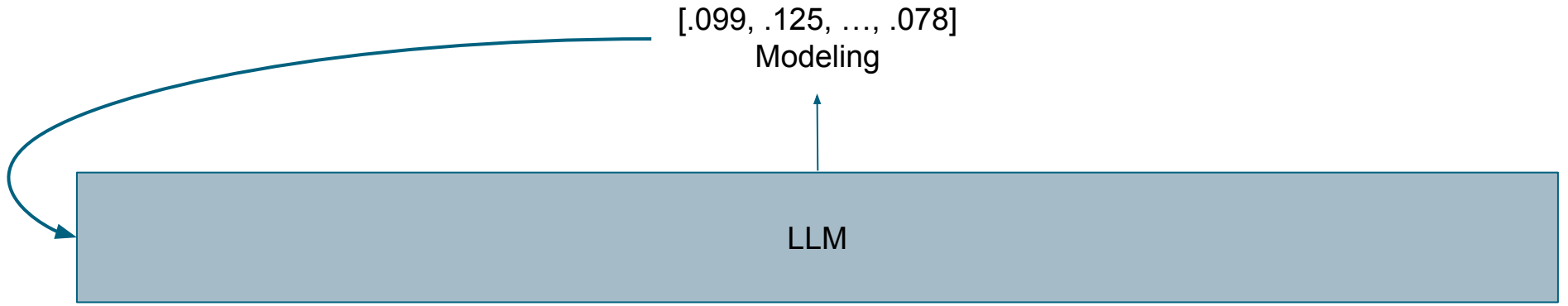
Autoregressive Generation: Causal Language Modeling (CLM)

Output: Large



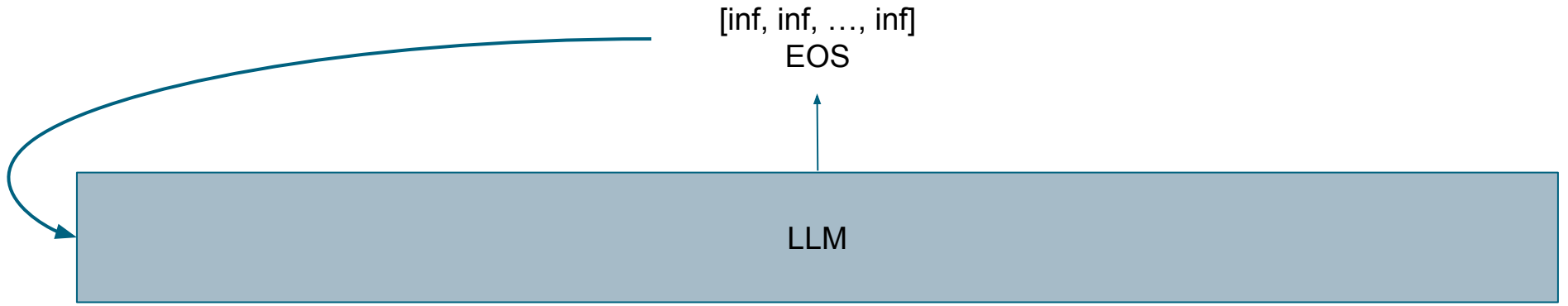
Autoregressive Generation: Causal Language Modeling (CLM)

Output: Large Language

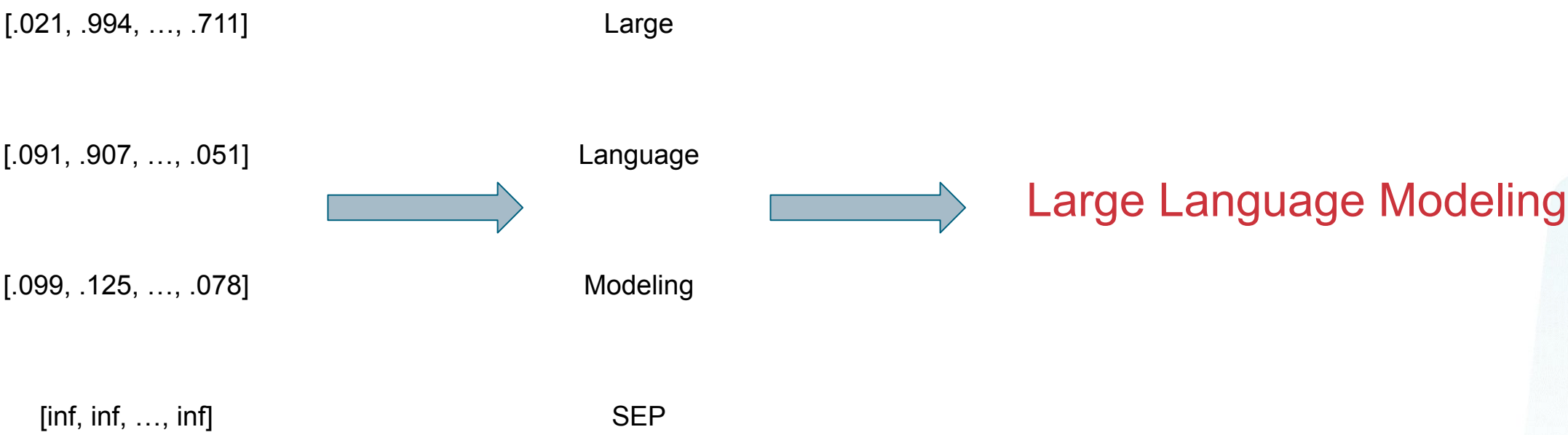


Autoregressive Generation: Causal Language Modeling (CLM)

Output: Large Language Modeling

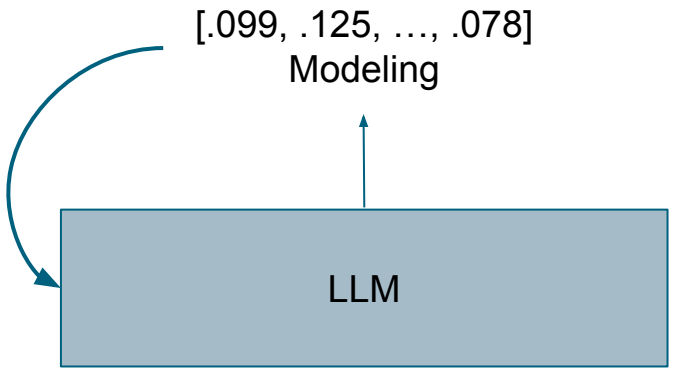


Autoregressive Generation: Decoding Stack



Complex Probability Distribution: Greedy Decoding

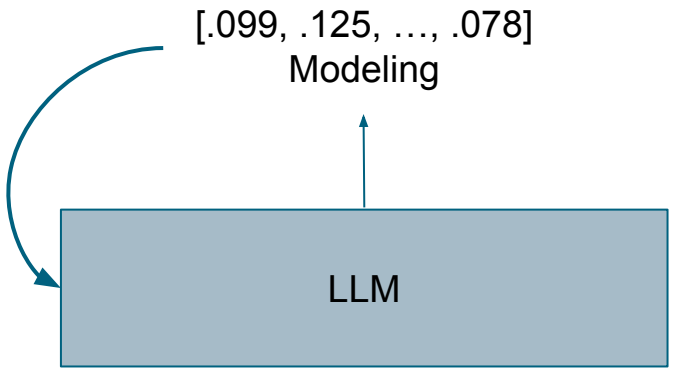
Output: Large Language



Large	.0002
Language	.0003
Modeling	.0059

Beam Search: Non greedy approach

Output: Large Language



Search Frontier: {Language, Modeling}

Controlling creativity: Temperature

High temperature: more creative generation

Cat sits on: moon, mars, sea

Low temperature: more realistic generation

Cat sits on: sofa, floor

Beam Search: Top-P

.0859
.0602
.0548
.0248
.0088
.0018
.0012
.0008
.0001

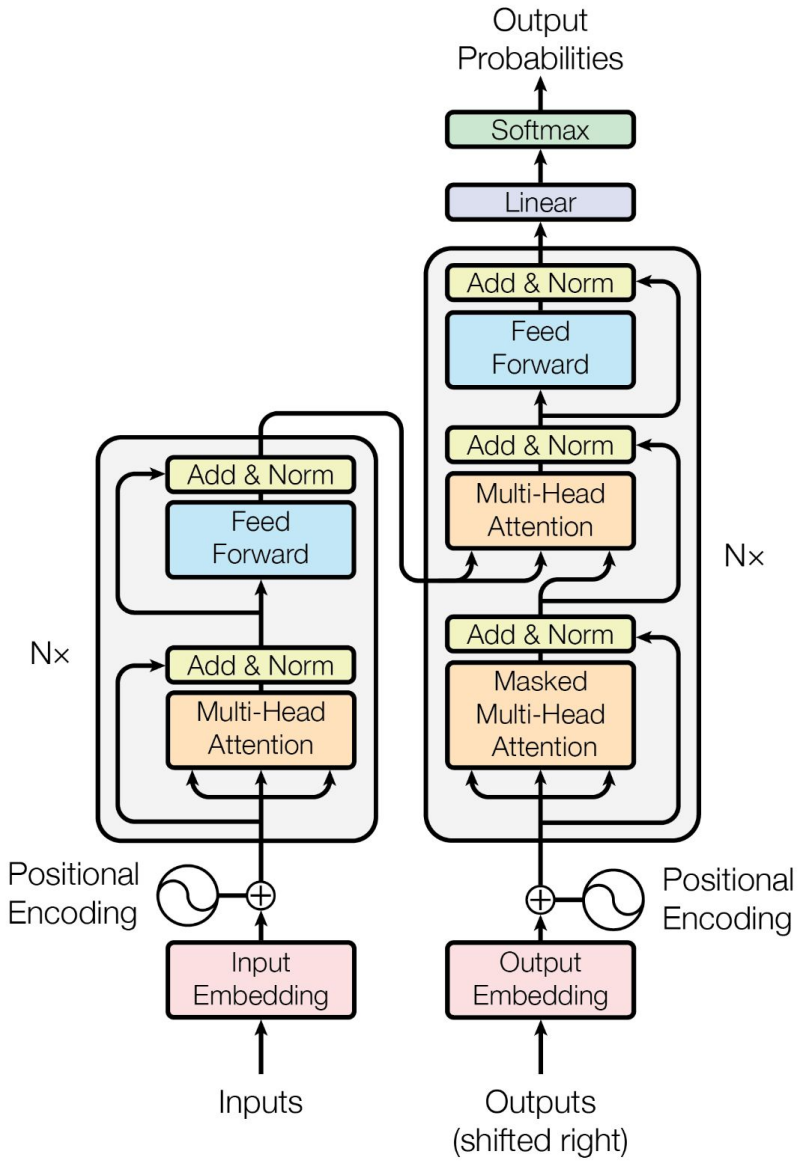
Top **P** % tokens are considered for generation

LLM architectures

LLM architectures

Encoder only	Encoder decoder	Decoder only
BERT	GPT - 2	GPT - 3,4

encoder - decoder



Vaswani *et al.*, 2017, Attention is all you need

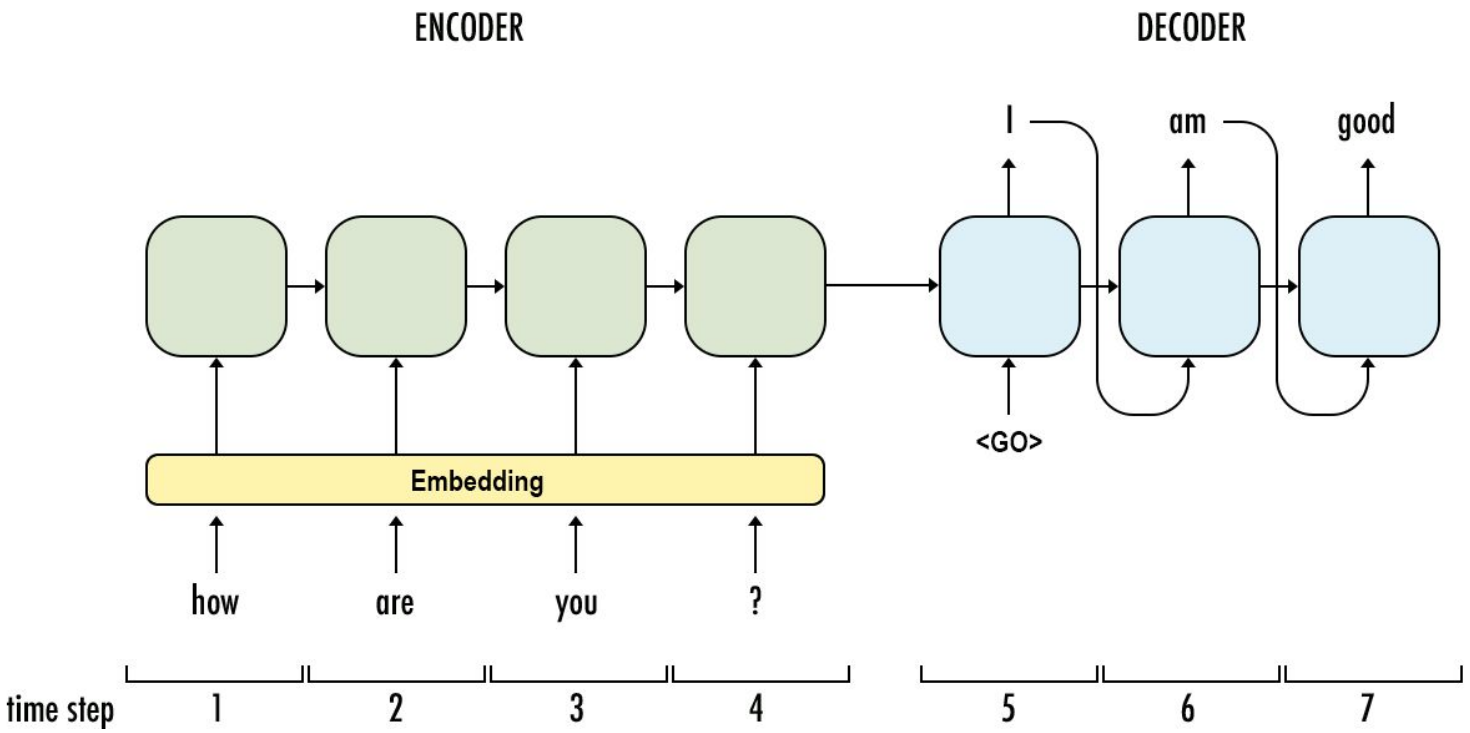
LLM modelling (tasks) types

Masked Language Modeling (MLM)	Causal Language Modeling (CLM)
BERT	GPT - 2,3,4

SECTION 2

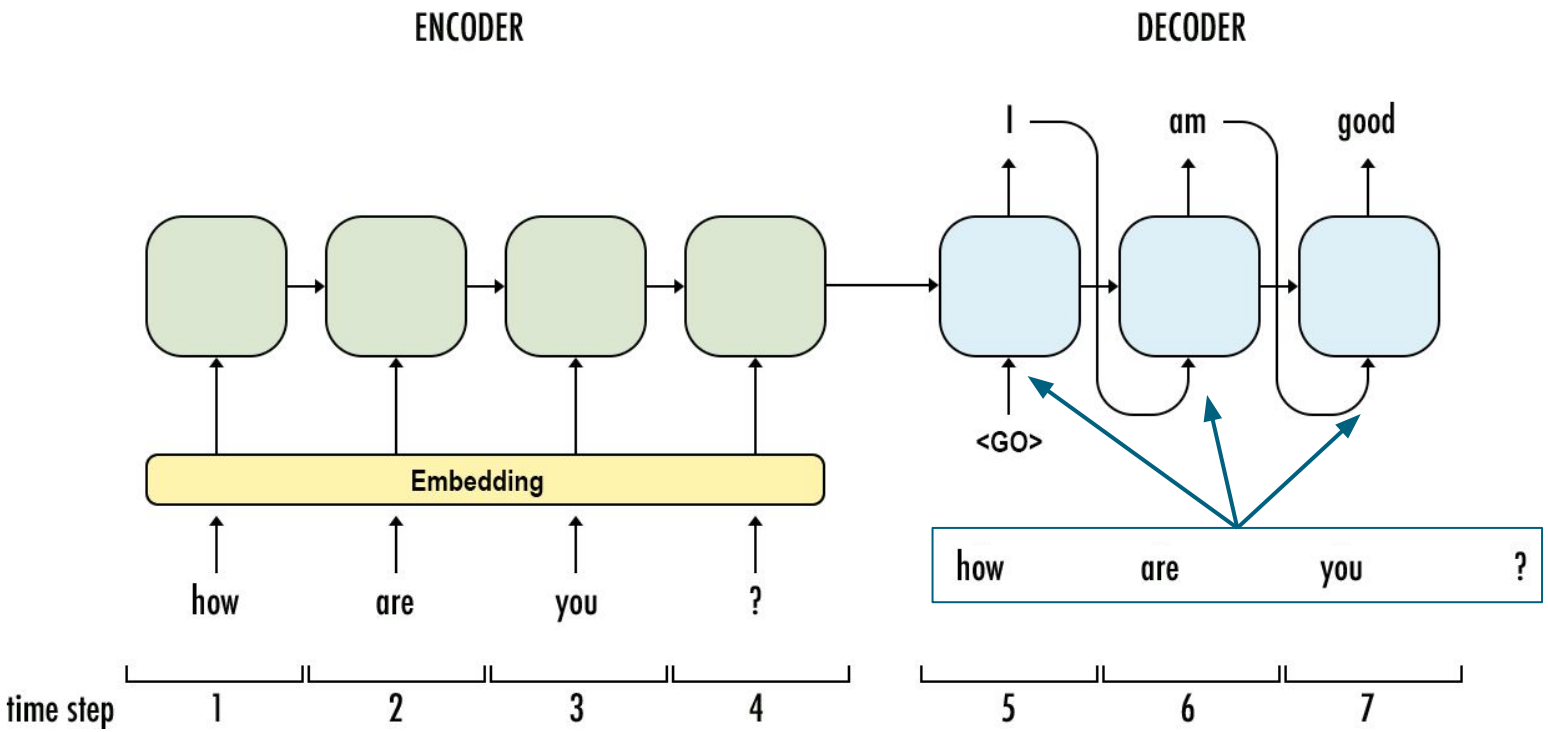
Evolution of LLM

Sequence to Sequence Models



Recurrent neural networks (without attention)
Problem: short context

Sequence to Sequence Models (with attention)



Recurrent neural networks (with attention)
Problem: vanishing gradients

Path Breaking Idea (Transformer)

RNN (attention) → Attention(RNN)

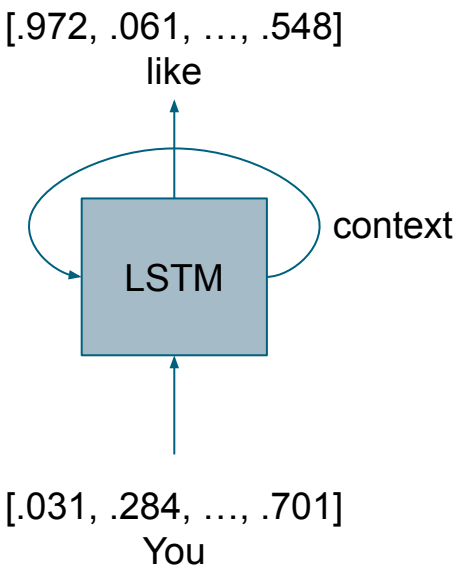
↓

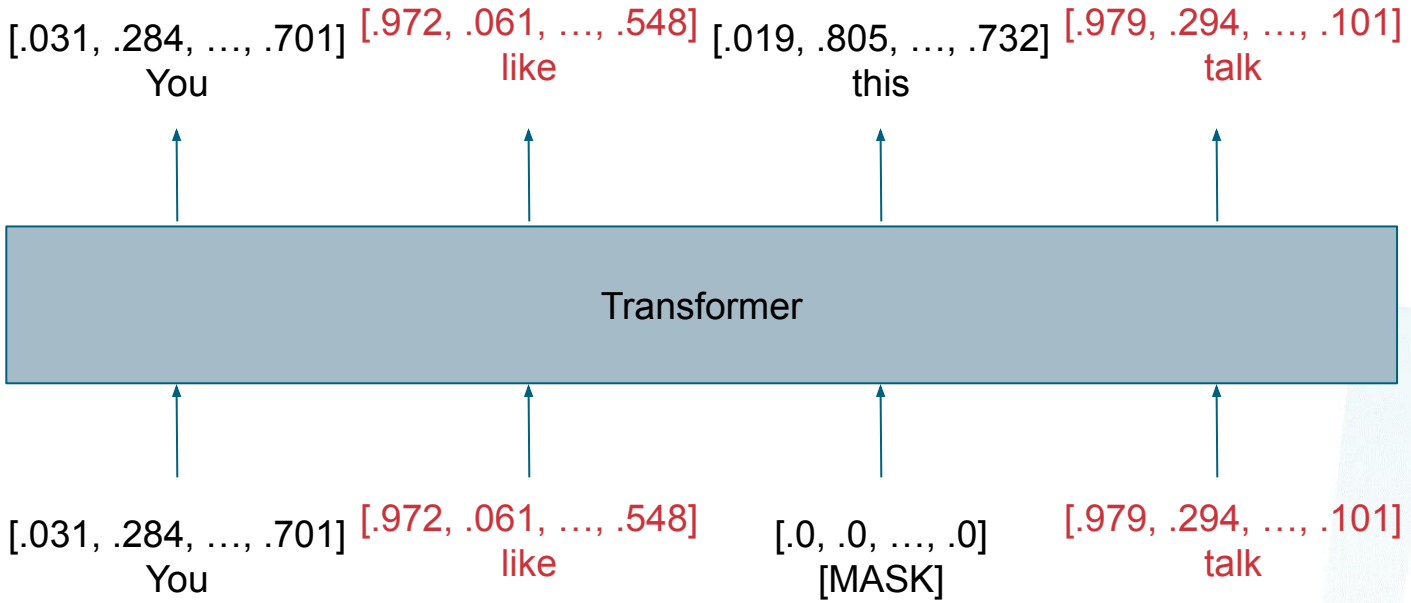
Sequential learning → Parallel Learning

↓

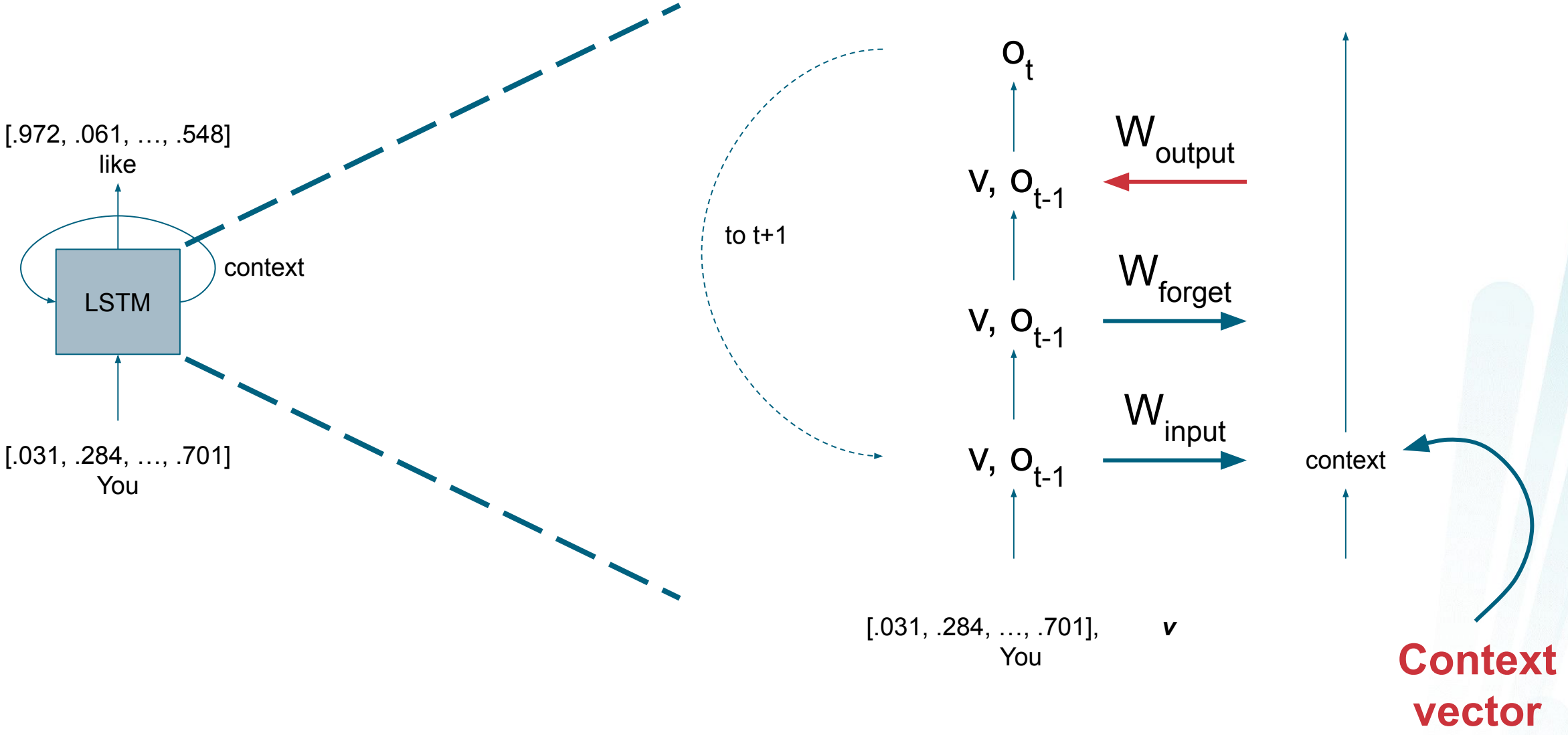
RNN → Transformer

RNN vs. Transformer

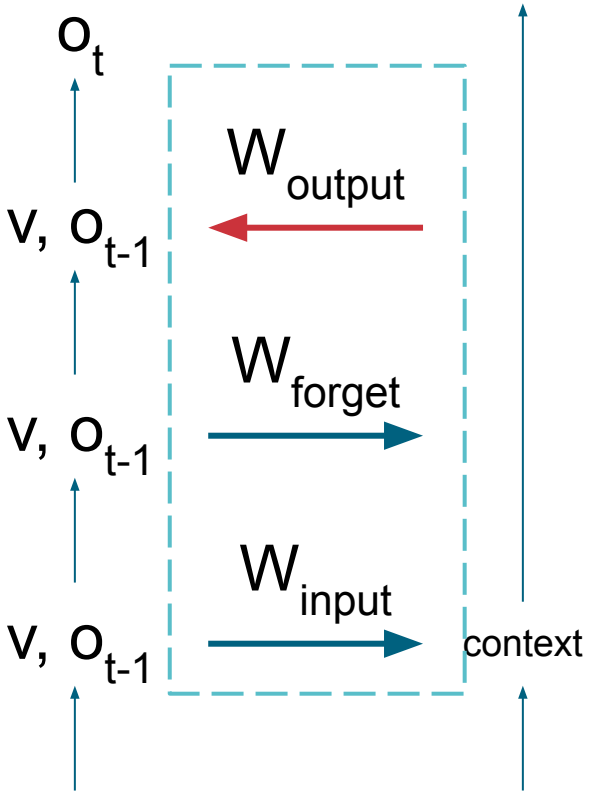




LLM evolution: Recurrent Neural Networks (RNNs)

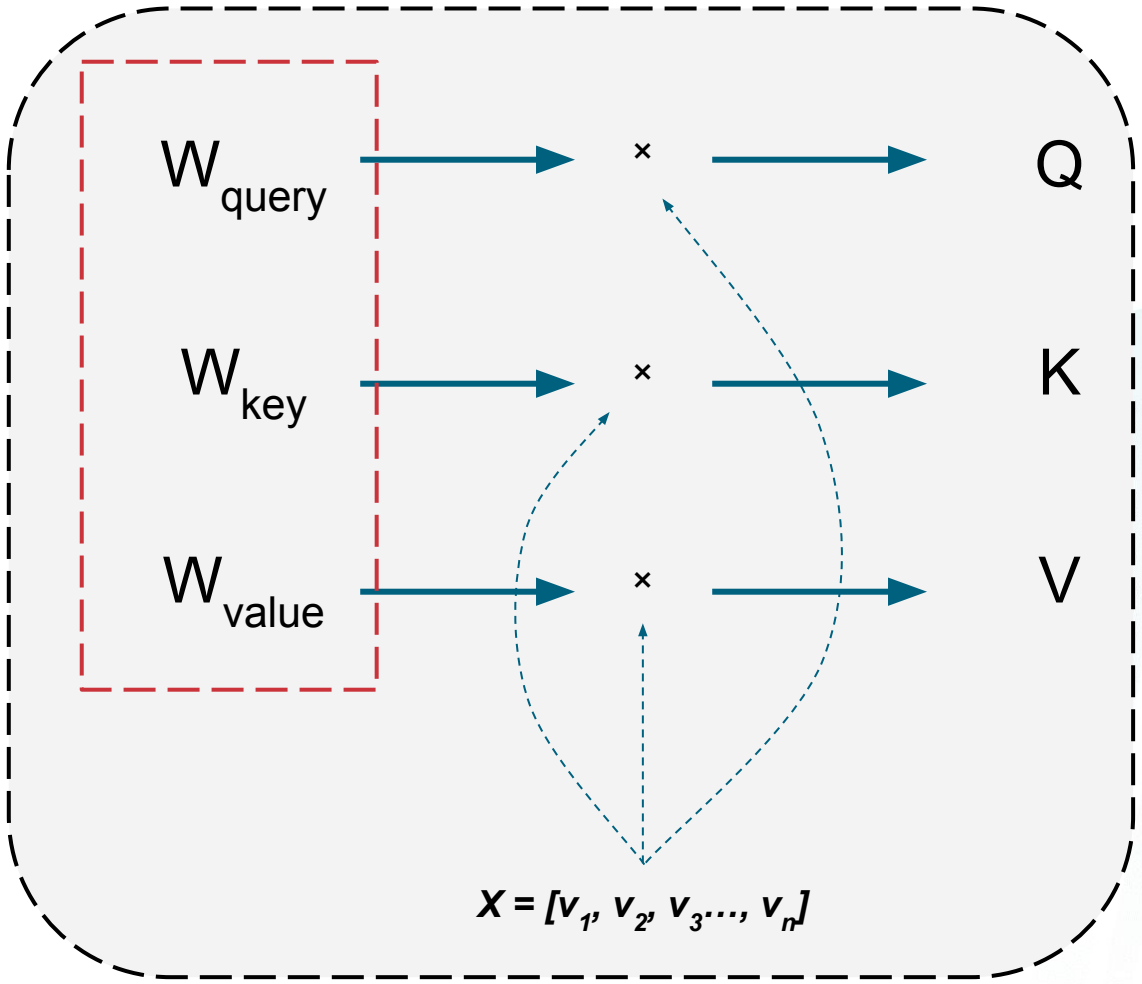


LLM evolution: Context/Ops Symmetry (RNN – Transformer)

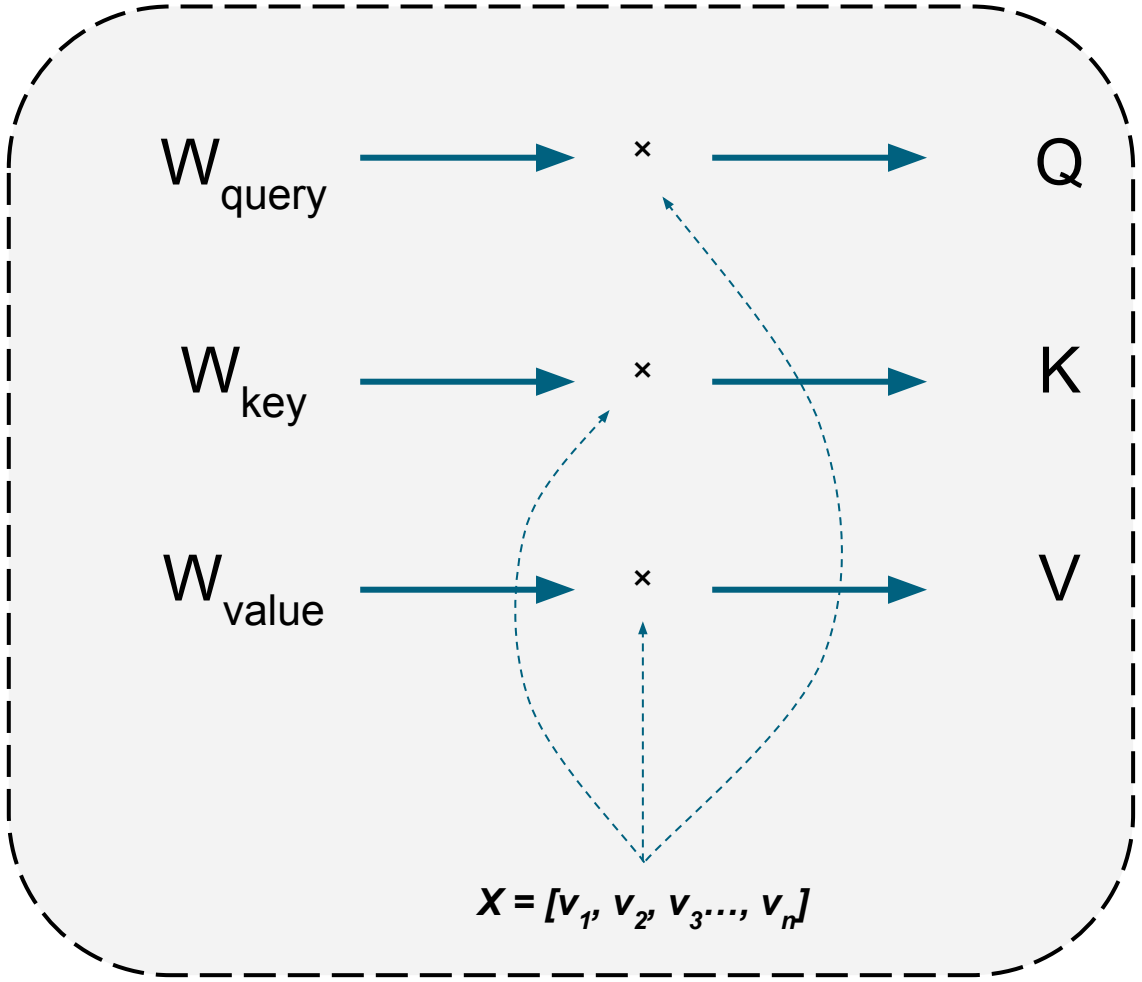


[.031, .284, …, .701], ***v***
You

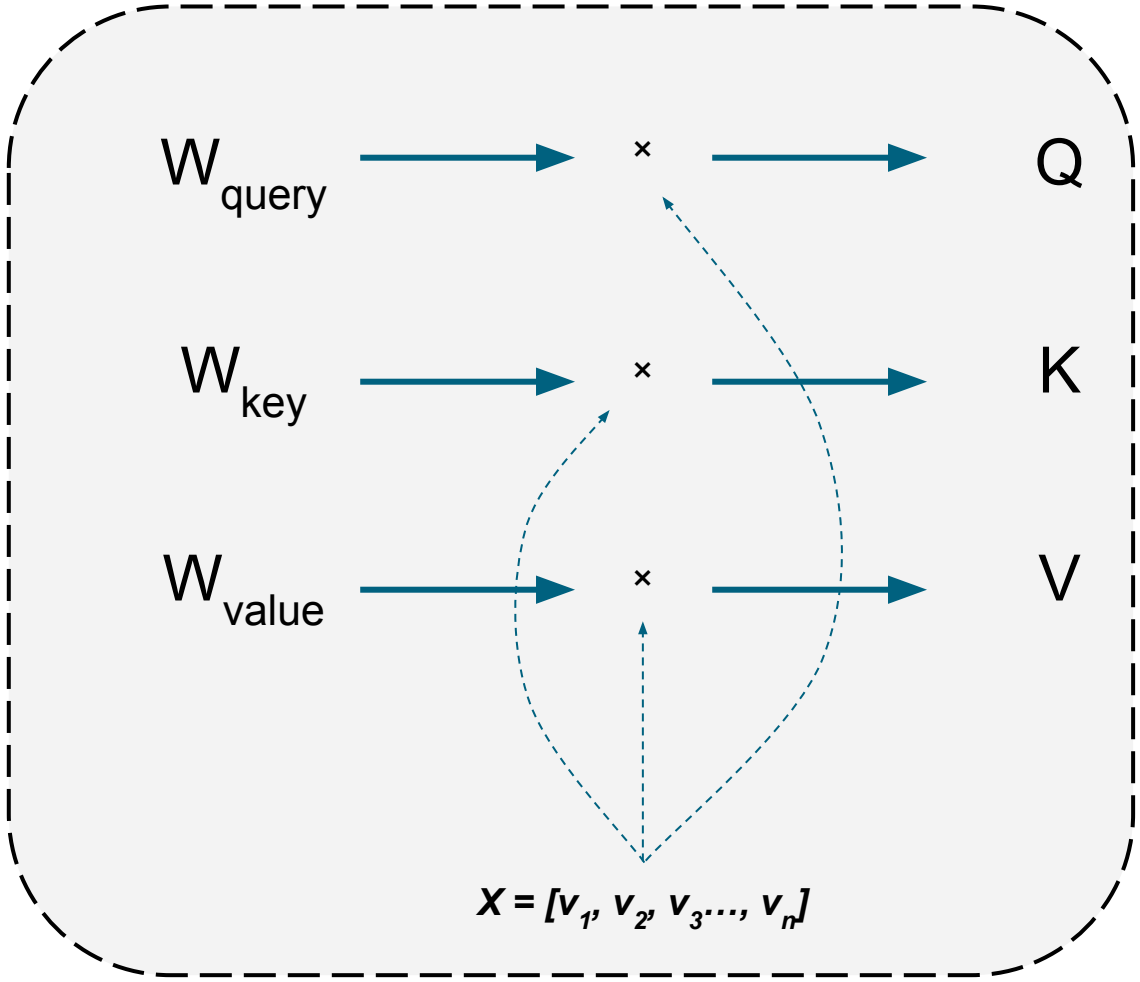
Self attention (heart of transformer)



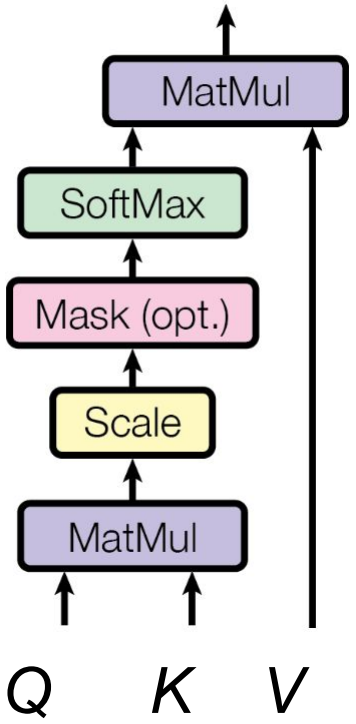
LLM evolution: Transformer (self-attention)



LLM evolution: Transformer (self-attention)



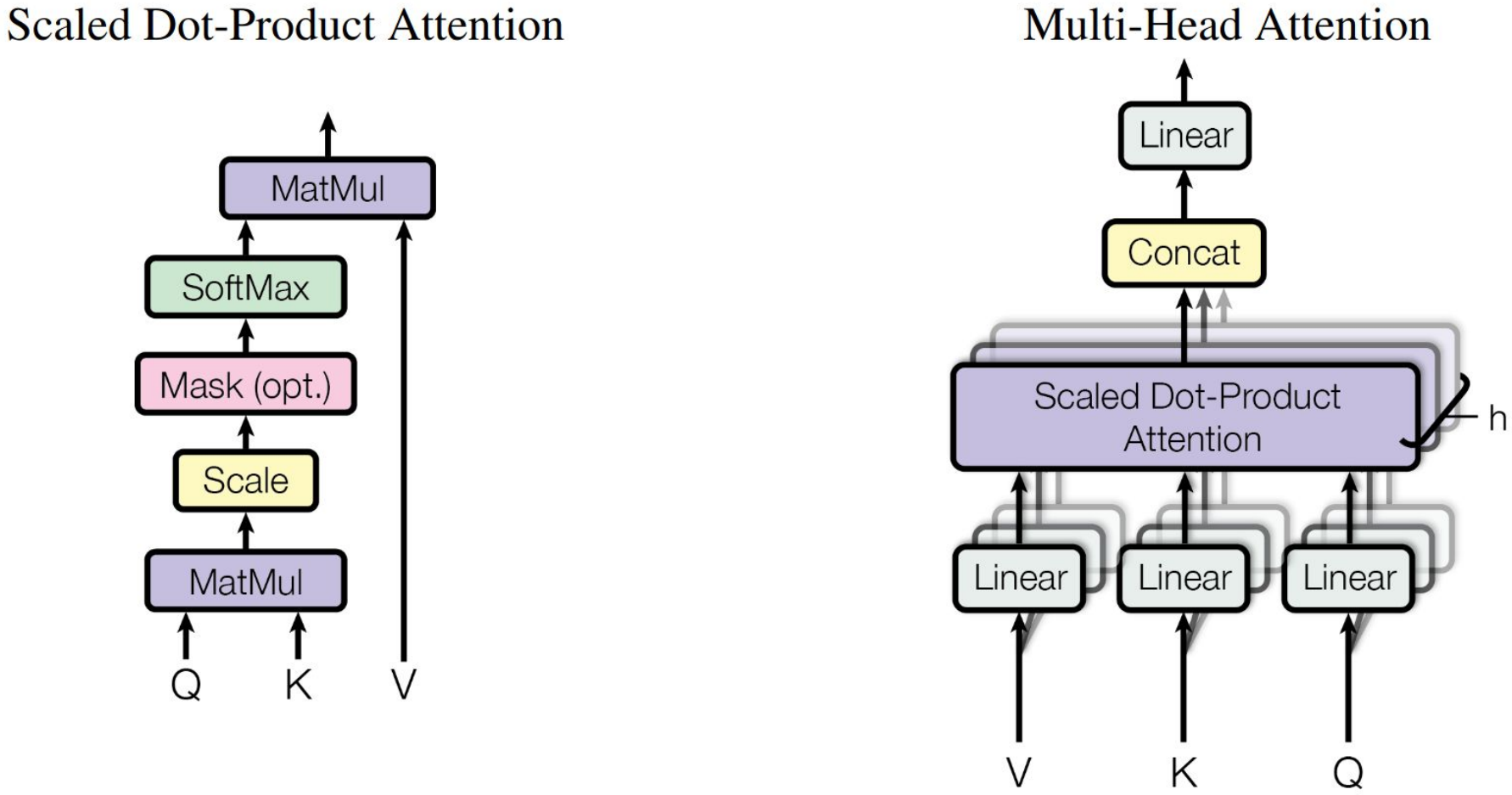
$$\text{Attention}(Q, K, V)$$



$$\text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

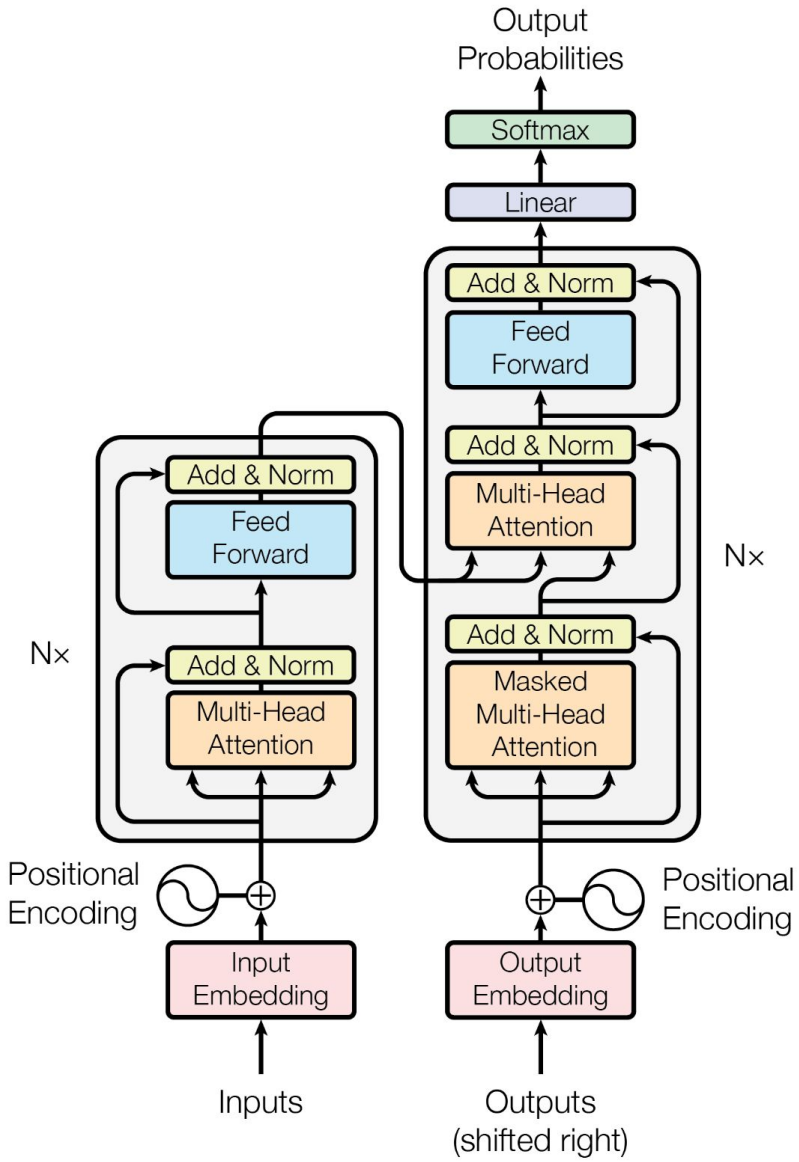
Vaswani *et al.*, 2017, Attention is all you need

LLM evolution: Transformer (multi-head attention)



Vaswani *et al.*, 2017, Attention is all you need

LLM evolution: Transformer (encoder - decoder)

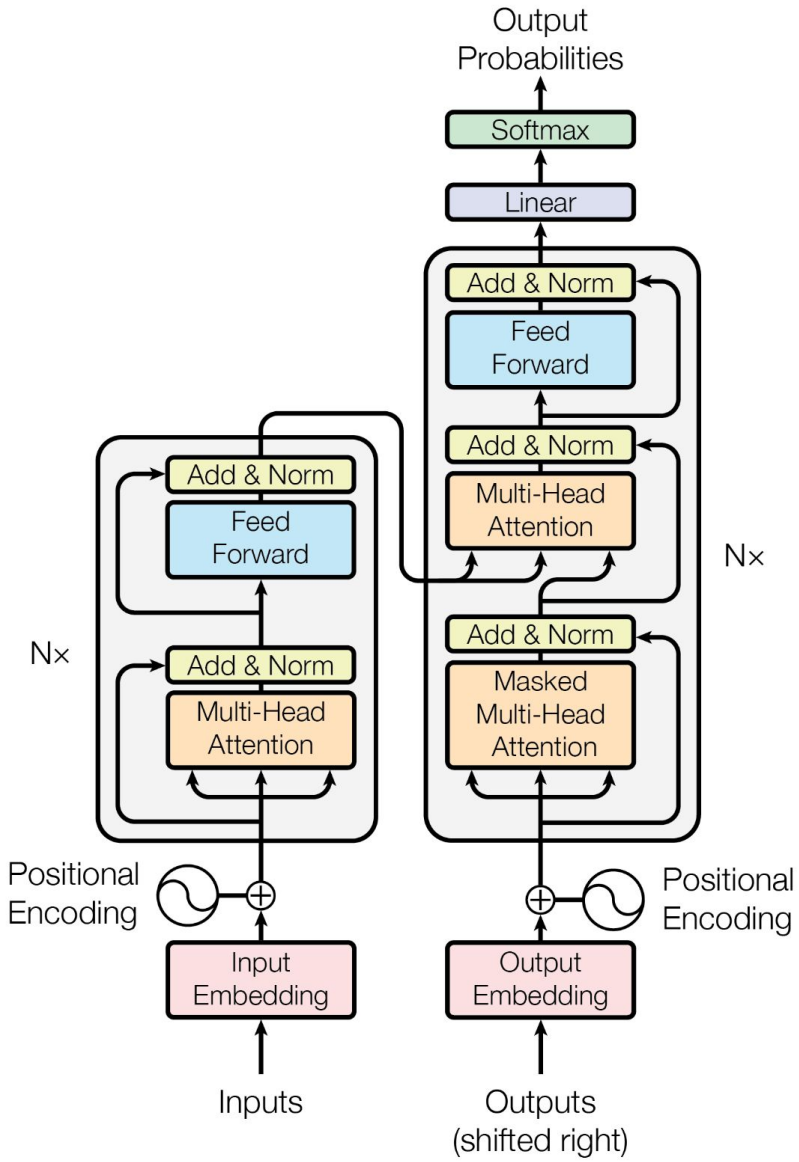


Vaswani *et al.*, 2017, Attention is all you need

LLM evolution: Transformer (encoder - decoder)

LLM

Massive Stacking



Vaswani *et al.*, 2017, Attention is all you need

?

When does LLM stop generation?

Conclusions

Tokenization: context encoding

Transformer: parallelization, efficient learning

Fatty encoders: scaling (compute, data)

Thank You

Arctic LLM Workshop 2023
Dept. of Computer Science

https://en.uit.no/enhet/ifi

www.bioailab.org