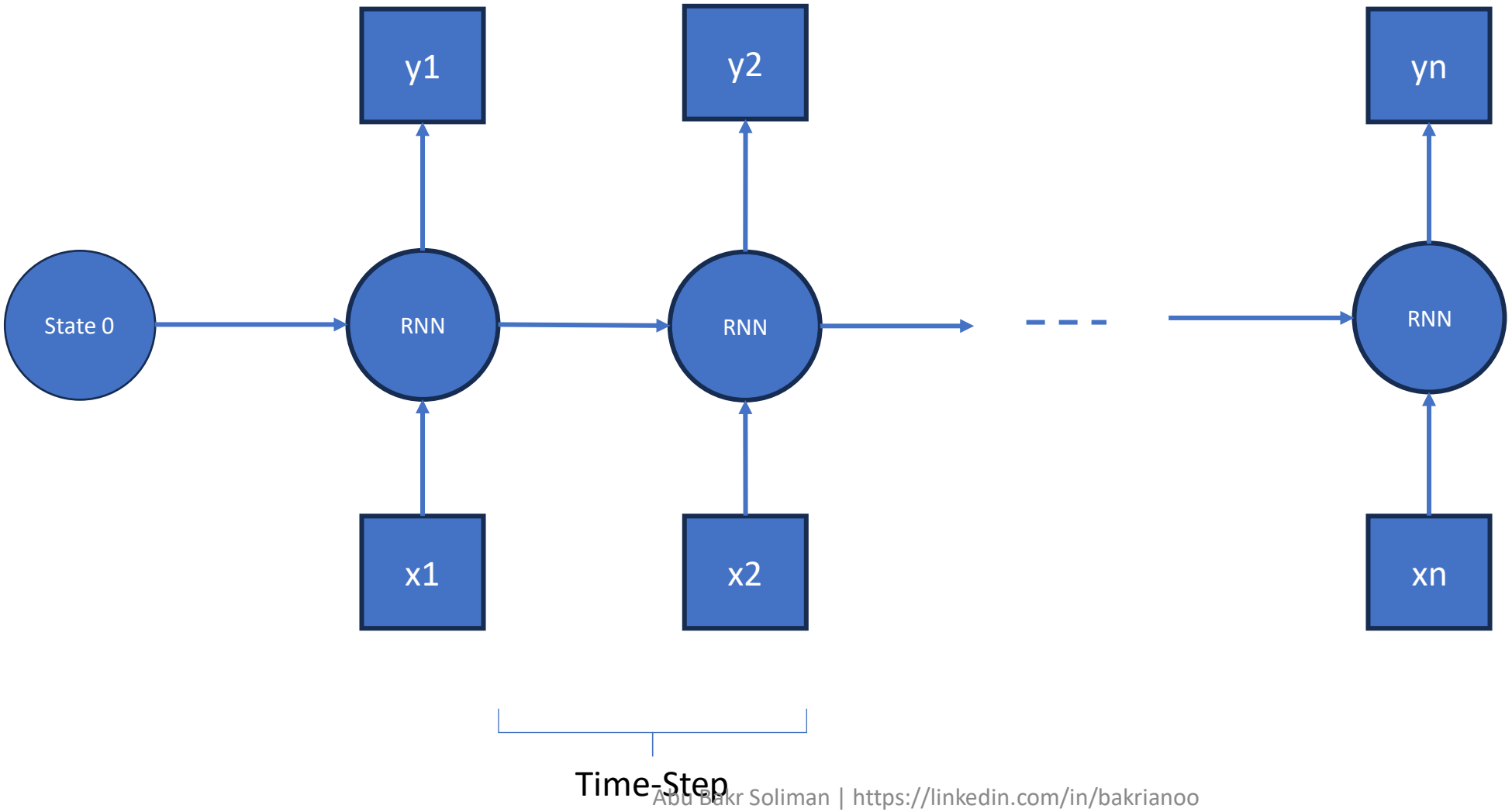
Transformers

Abu Bakr Soliman

Sequence to Sequence (Seq2Seq)

Input	x1	x2	..	xn
Output	y1	y2	..	yn

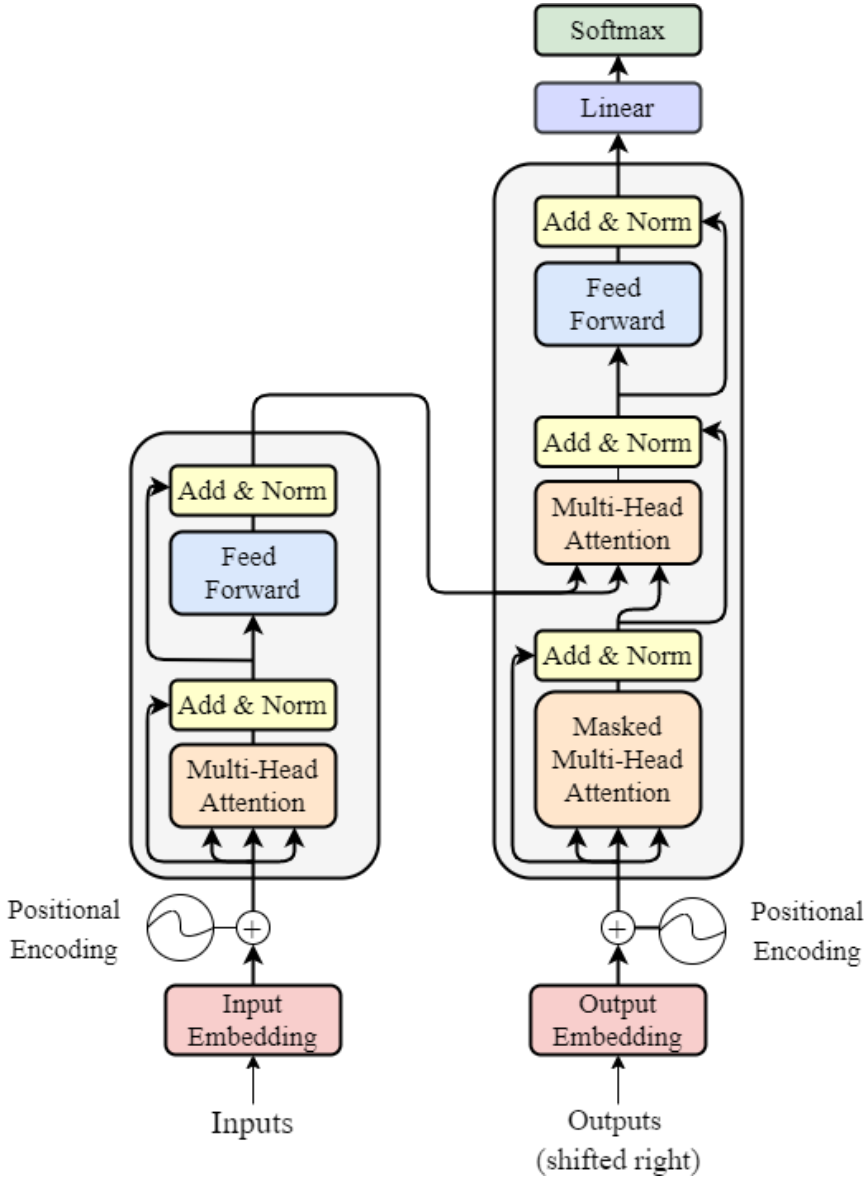
Recurrent Neural Network (RNN)



RNN Problems

- Long Sequences = Slow Computations
- Vanishing and Exploding Gradients
- Vanished Memory

Attention is All You Need



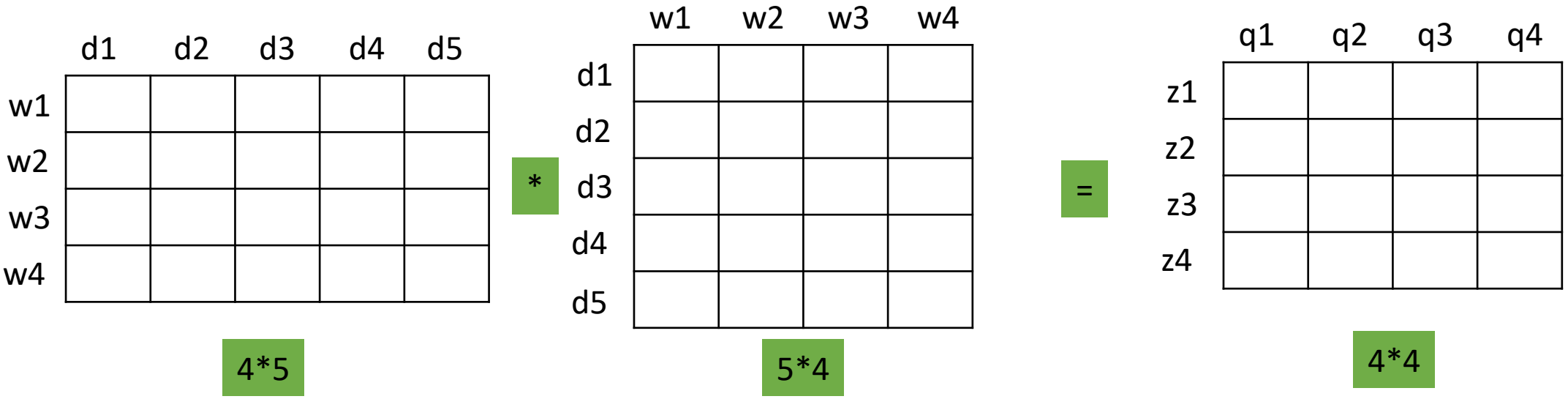
Vectors

Patient ID
10200
3600

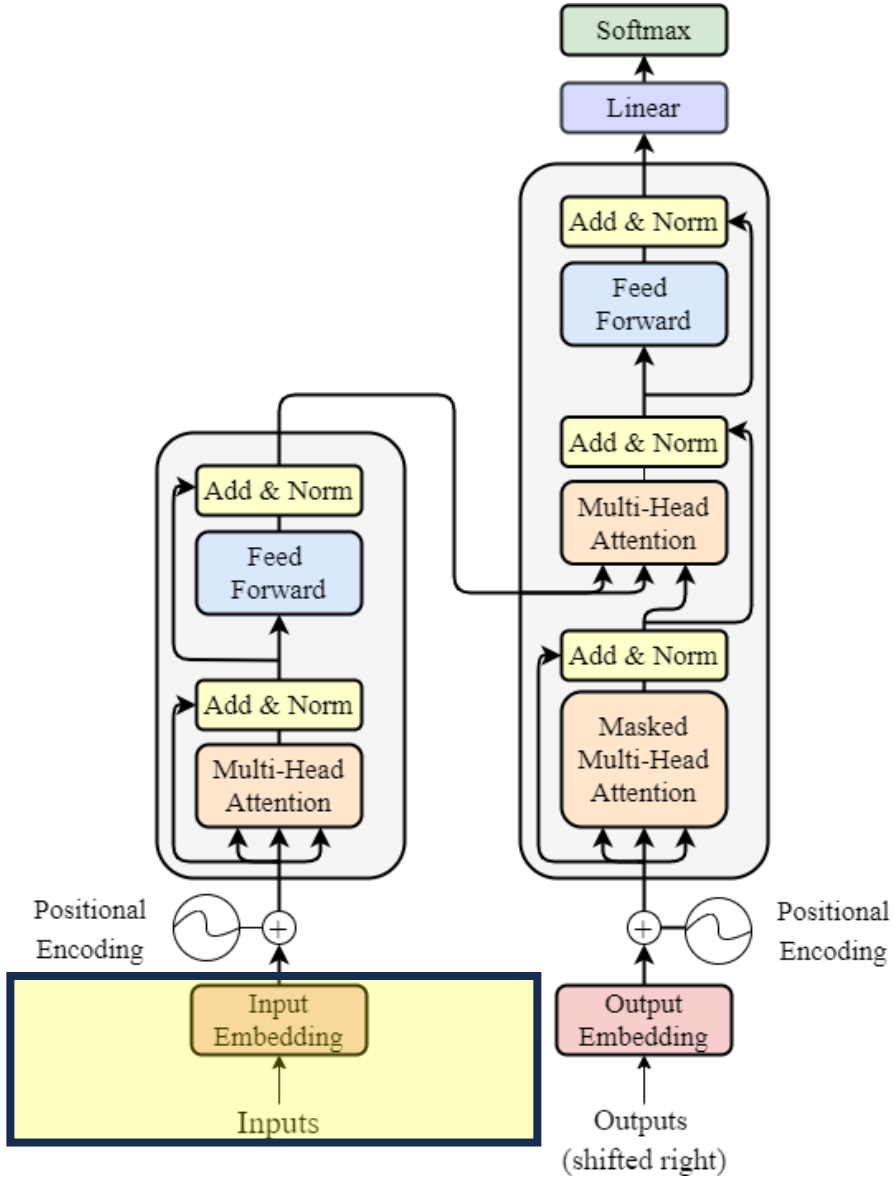
Vectors

Patient ID	Height (CM)	Weight (KG)	Systolic (mmHg)	Diastolic (mmHg)
10200	165	71	120	80
3600	180	92	110	85

Matrices Operations



Attention is All You Need

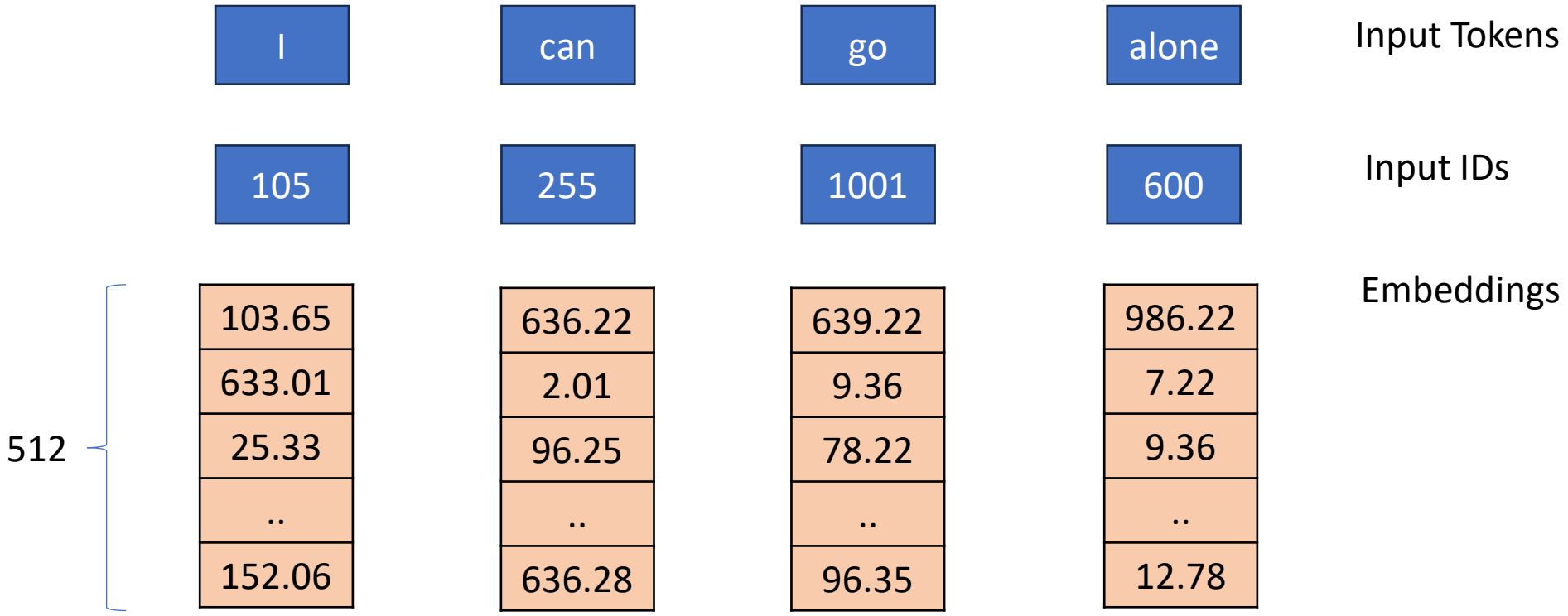


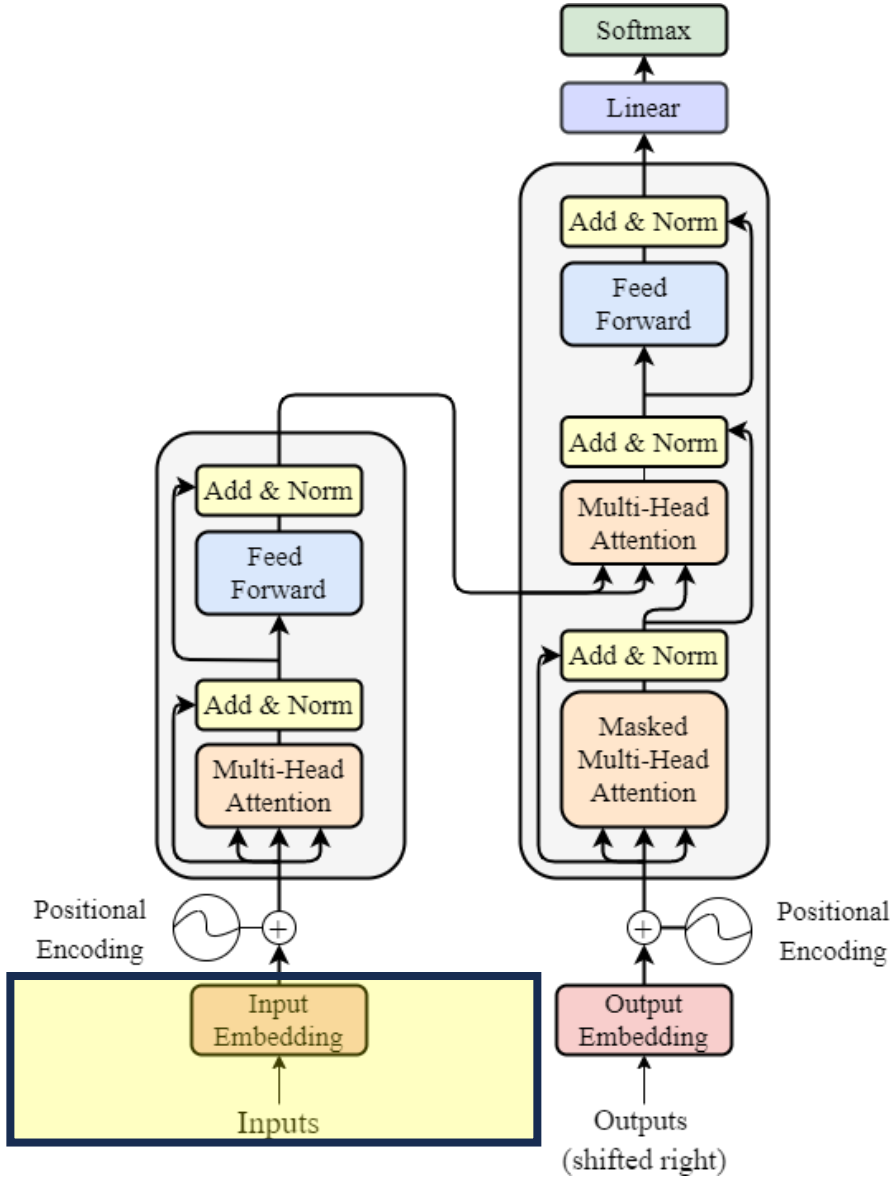
Tokenizer

Token	ID
Hello	1
Go	2
ed	3
..	..
red	32000

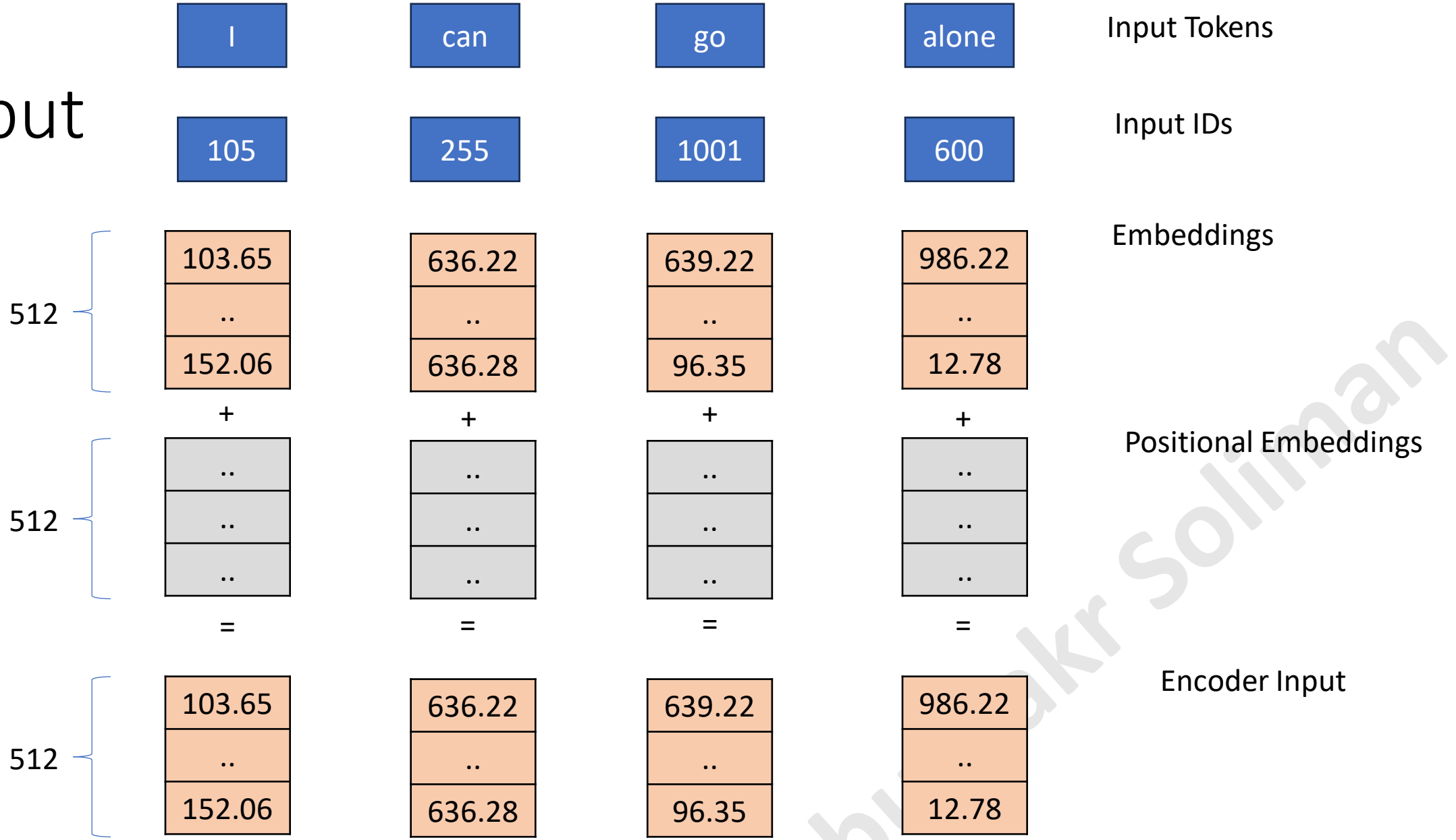
Input

I can go alone




Softmax
Linear
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
Add & Norm
Masked Multi-Head Attention
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
Positional Encoding
Positional Encoding
Input Embedding
Output Embedding
Inputs
Outputs (shifted right)

Input

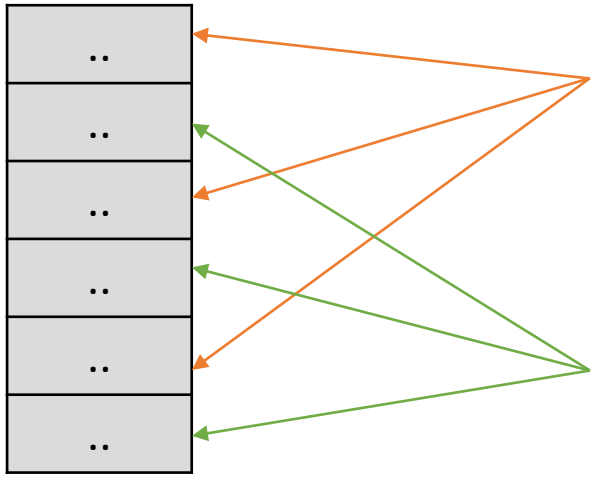


Abu Bakr Soliman

I	can	go	alone
105	255	1001	600

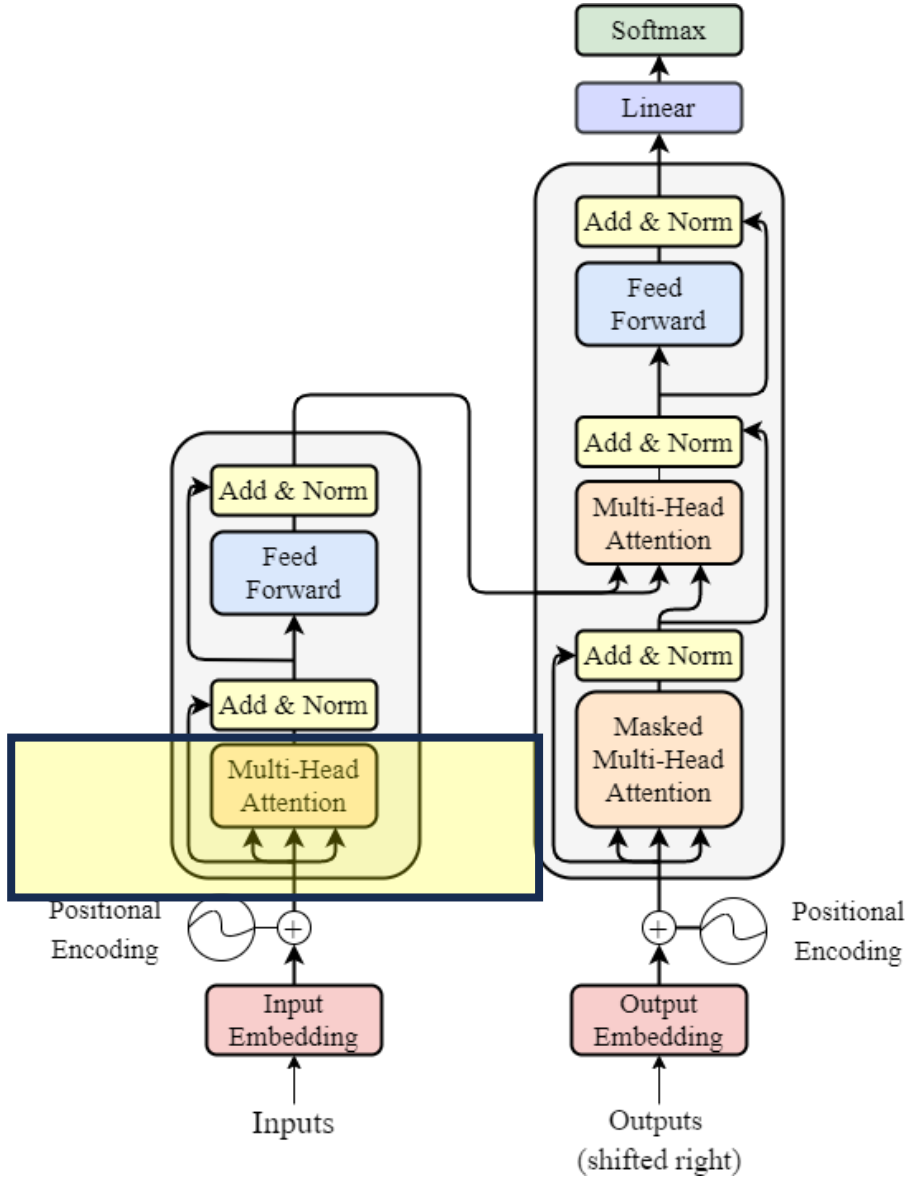
Input Tokens

Input IDs



$$PE(pos, 2i) = \sin \frac{pos}{10000^{\frac{2i}{d_{model}}}}$$

$$PE(pos, 2i+1) = \cos \frac{pos}{10000^{\frac{2i}{d_{model}}}}$$


Softmax
Linear
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
Add & Norm
Masked Multi-Head Attention
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
Positional Encoding
Positional Encoding
Input Embedding
Output Embedding
Inputs
Outputs (shifted right)

Self-Attention

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

$d_k = d_{model}$	512
seq	4

$$\text{softmax}\left\{\frac{Q_{4*512} * K^T_{512*4}}{\sqrt{512}}\right\} =$$

	I	can	go	alone
I	0.7	0.2	0.1	0.1
can	0.3	0.5	0.1	0.1
go	0.1	0.3	0.4	0.2
alone	0.05	0.05	0.1	0.8

4*4

Abu Bakr Soliman

Self-Attention

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

	I	can	go	alone
I	0.7	0.2	0.1	0.1
can	0.3	0.5	0.1	0.1
go	0.1	0.3	0.4	0.2
alone	0.05	0.05	0.1	0.8

4*4

*

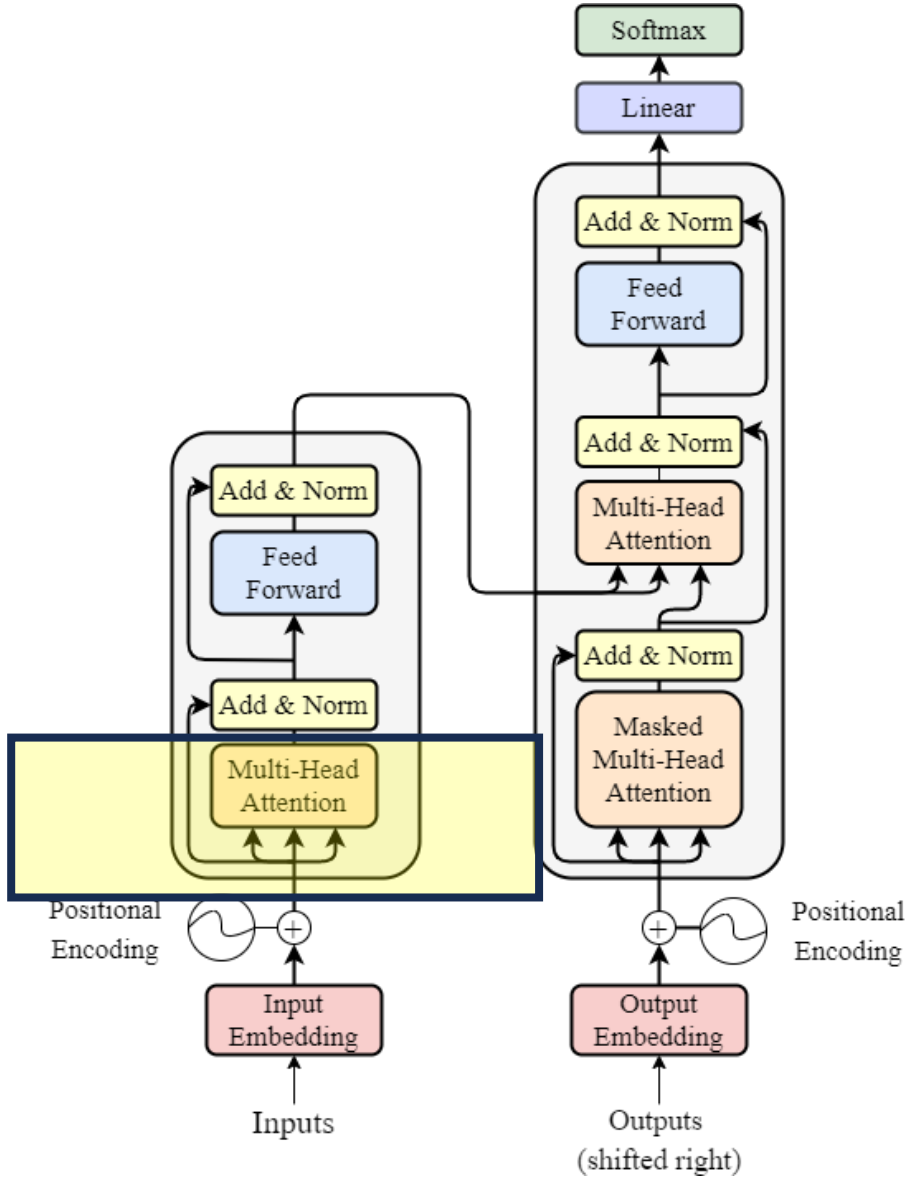
V

4*512

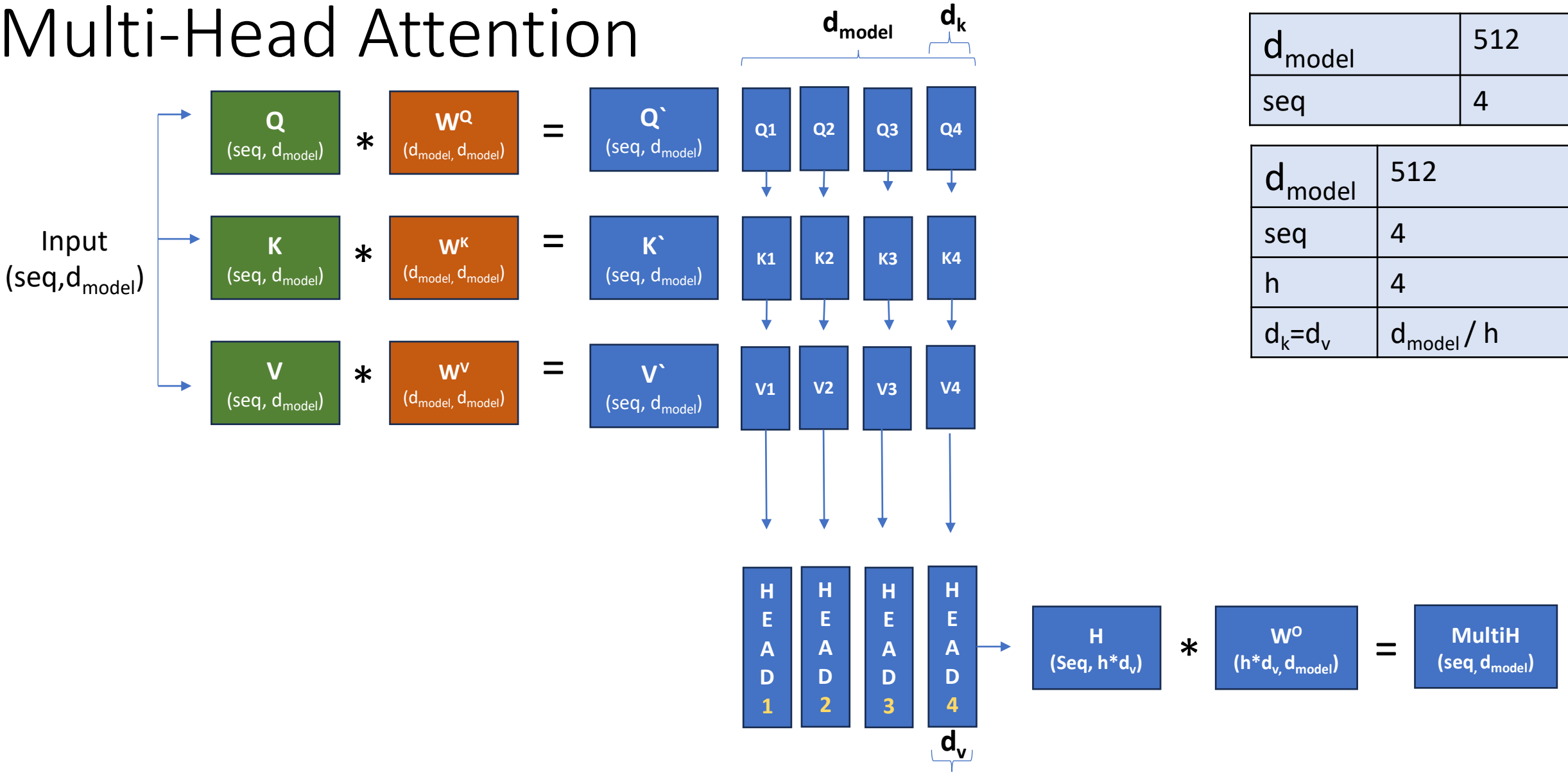
=

$$Attention(Q, K, V)$$

4*512


Softmax
Linear
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
Add & Norm
Masked Multi-Head Attention
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
Positional Encoding
Positional Encoding
Input Embedding
Output Embedding
Inputs
Outputs (shifted right)

Multi-Head Attention

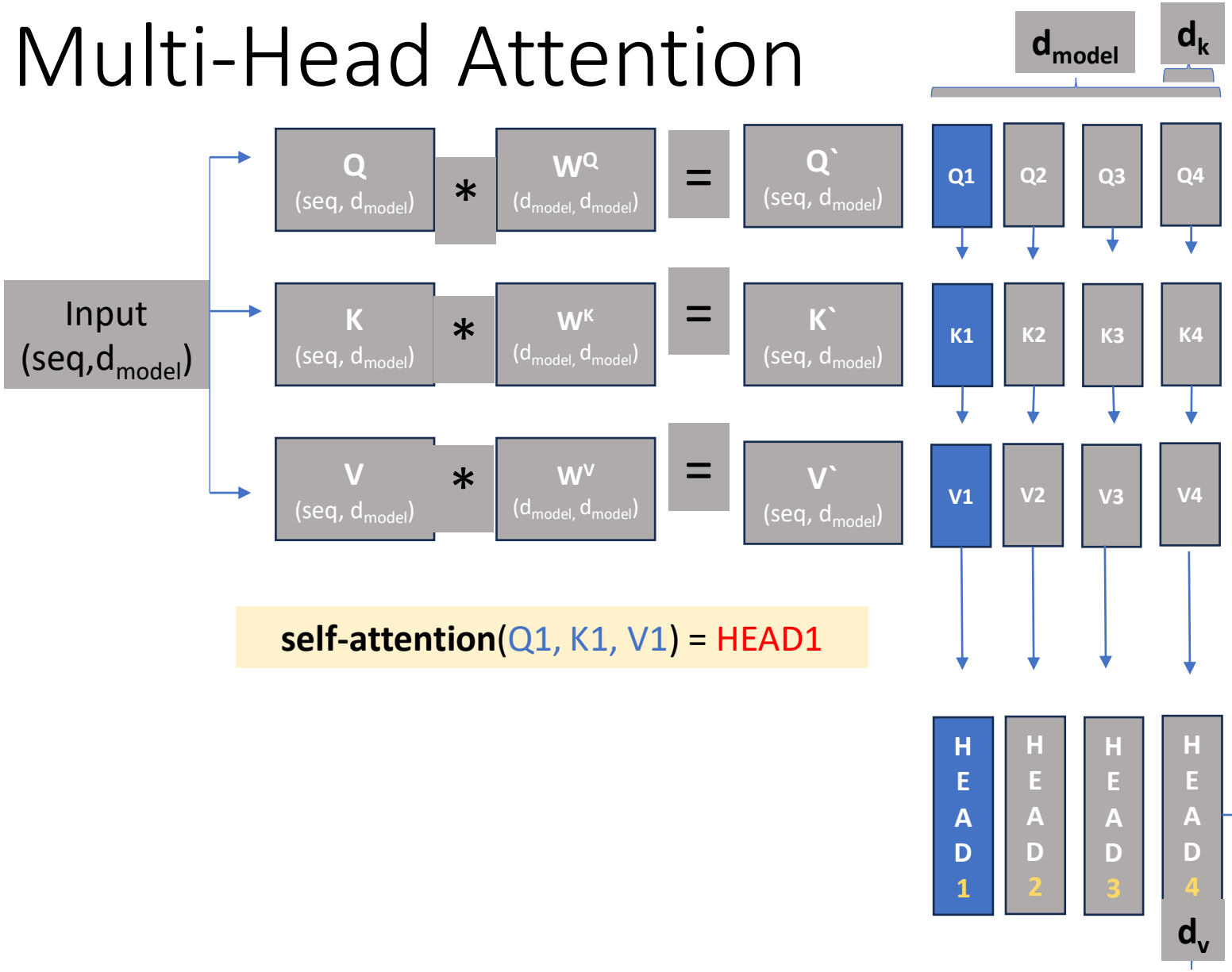


d_{model}	512
seq	4

d_{model}	512
seq	4
h	4
d_k=d_v	d_{model} / h

$$MultiHead(Q,K,V) = Concat(head_1 \ldots head_h)W^O$$

Multi-Head Attention



d_{model}	512
seq	4

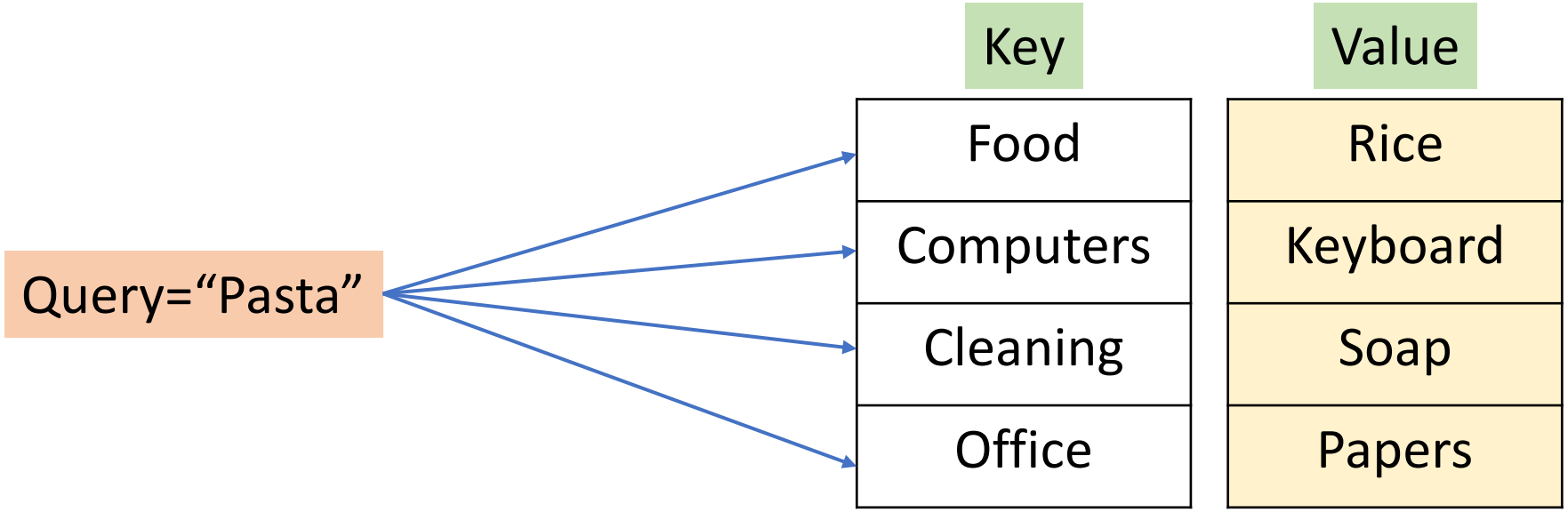
d_{model}	512
seq	4
h	4
d_k=d_v	d_{model} / h

self-attention(Q1, K1, V1) = HEAD1

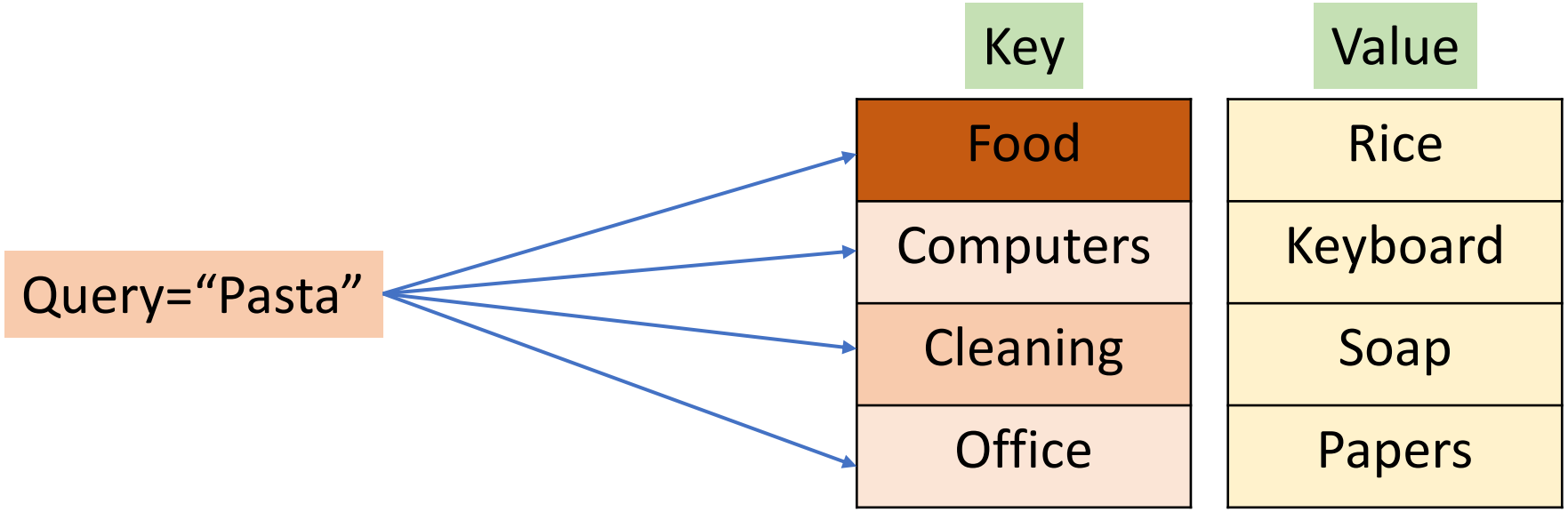
H (Seq, h*d_v) * W^O (h*d_v, d_{model}) = MultiH (seq, d_{model})

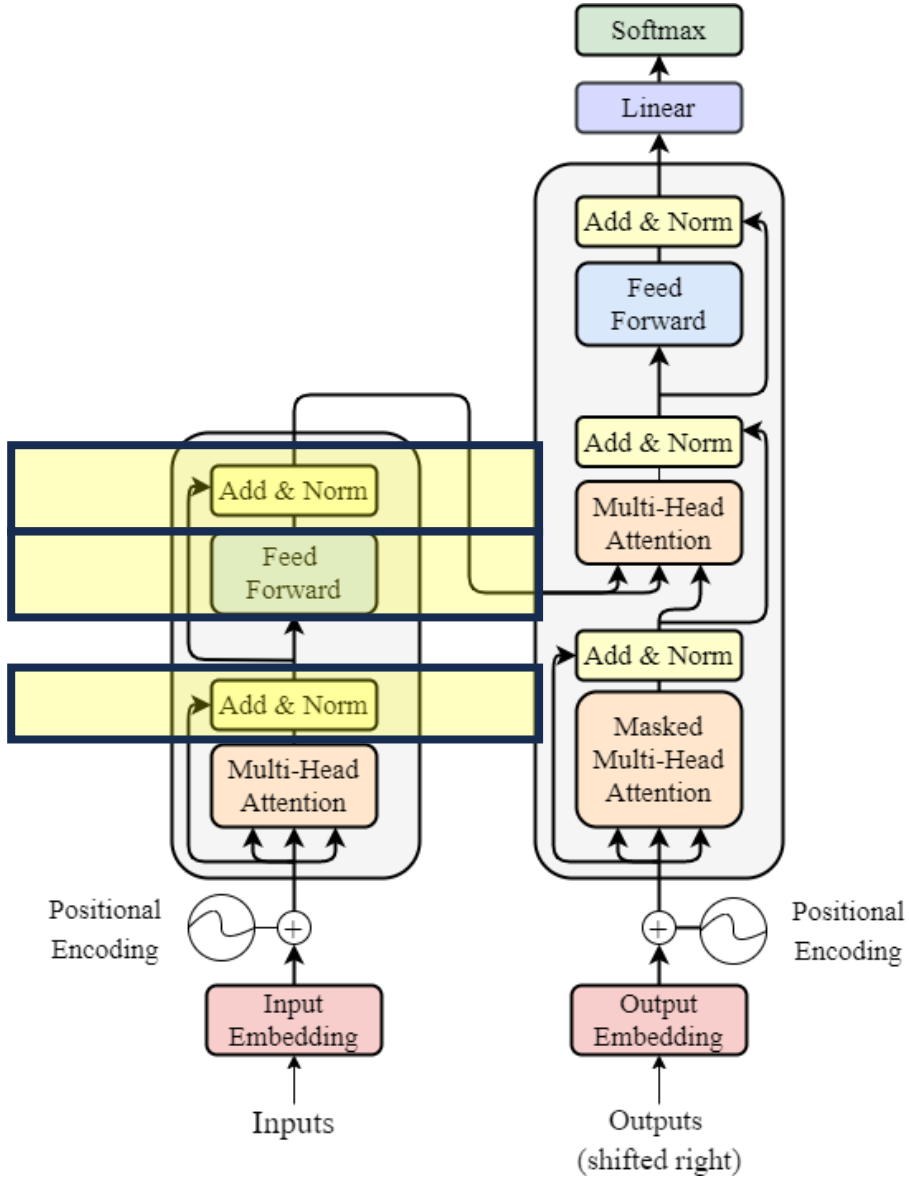
$$MultiHead(Q, K, V) = Concat(head_1 \dots head_h)W^O$$

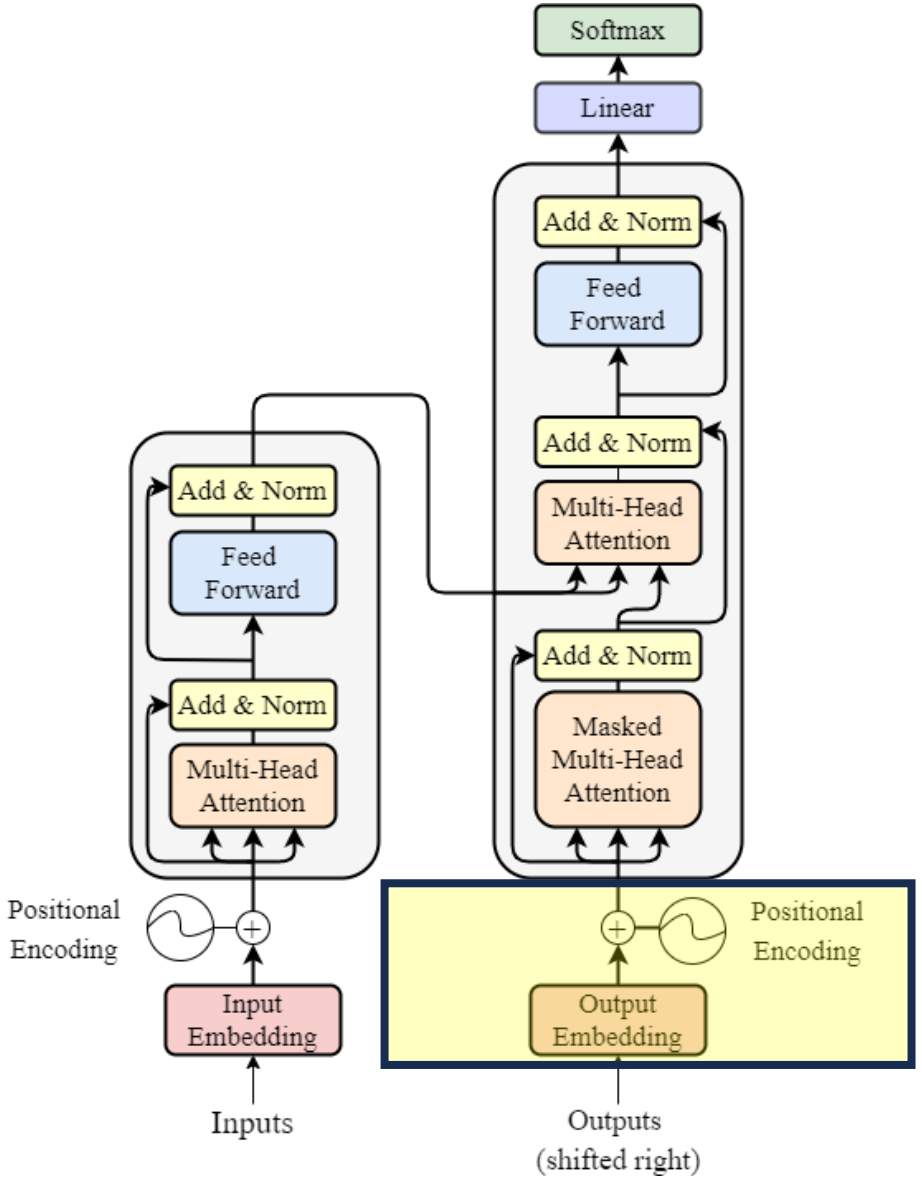
Query, Key & Value

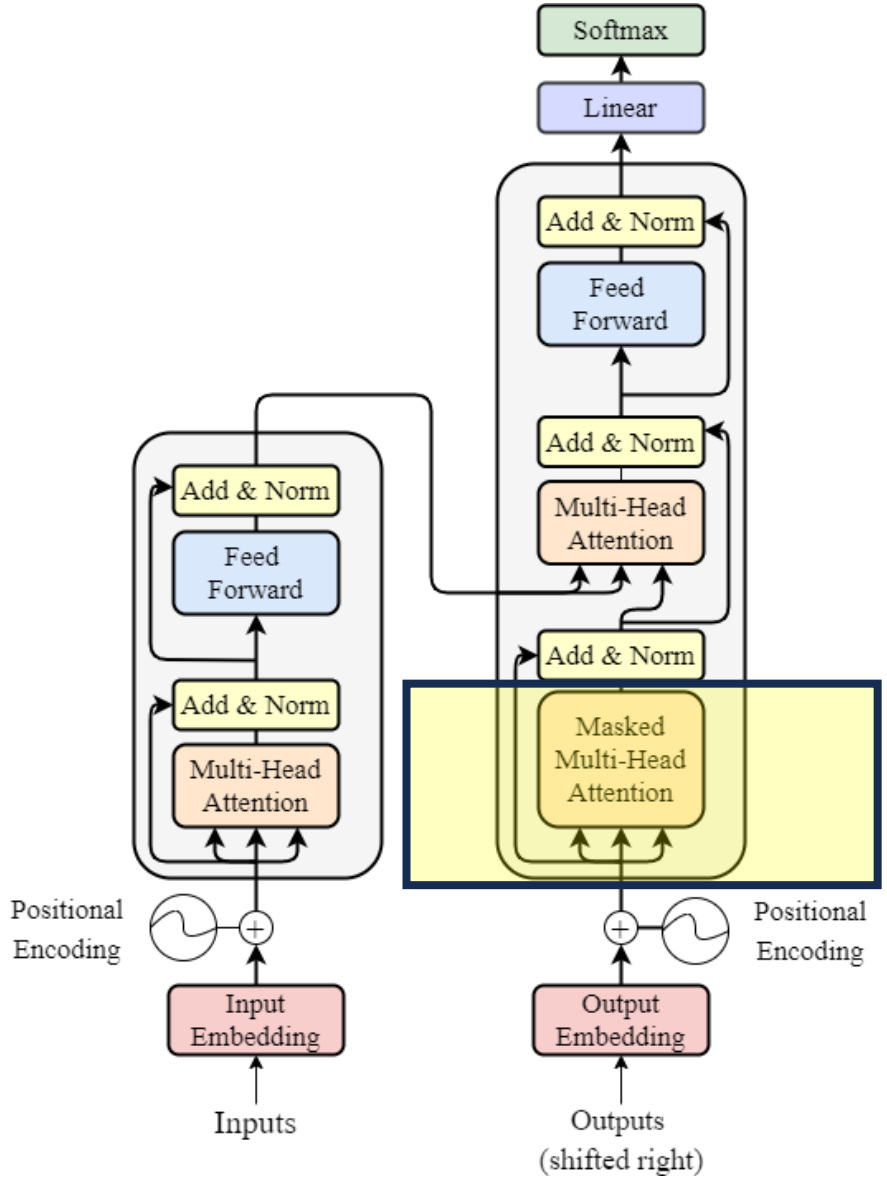


Query, Key & Value




Softmax
Linear
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
Add & Norm
Masked Multi-Head Attention
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
Positional Encoding
Positional Encoding
Input Embedding
Output Embedding
Inputs
Outputs (shifted right)


Softmax
Linear
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
Add & Norm
Masked Multi-Head Attention
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
Positional Encoding
Positional Encoding
Input Embedding
Output Embedding
Inputs
Outputs (shifted right)


Softmax
Linear
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
Add & Norm
Masked Multi-Head Attention
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
Positional Encoding
Positional Encoding
Input Embedding
Output Embedding
Inputs
Outputs (shifted right)

Masked Attention

$$\text{softmax}\left(\frac{Q * K^T}{\sqrt{512}}\right) =$$

Q: 4*512 ; K^T: 512*4

	I	can	go	alone
I	0.7	0.2	0.1	0.1
can	0.3	0.5	0.1	0.1
go	0.1	0.3	0.4	0.2
alone	0.05	0.05	0.1	0.8

4*4

Masked Attention

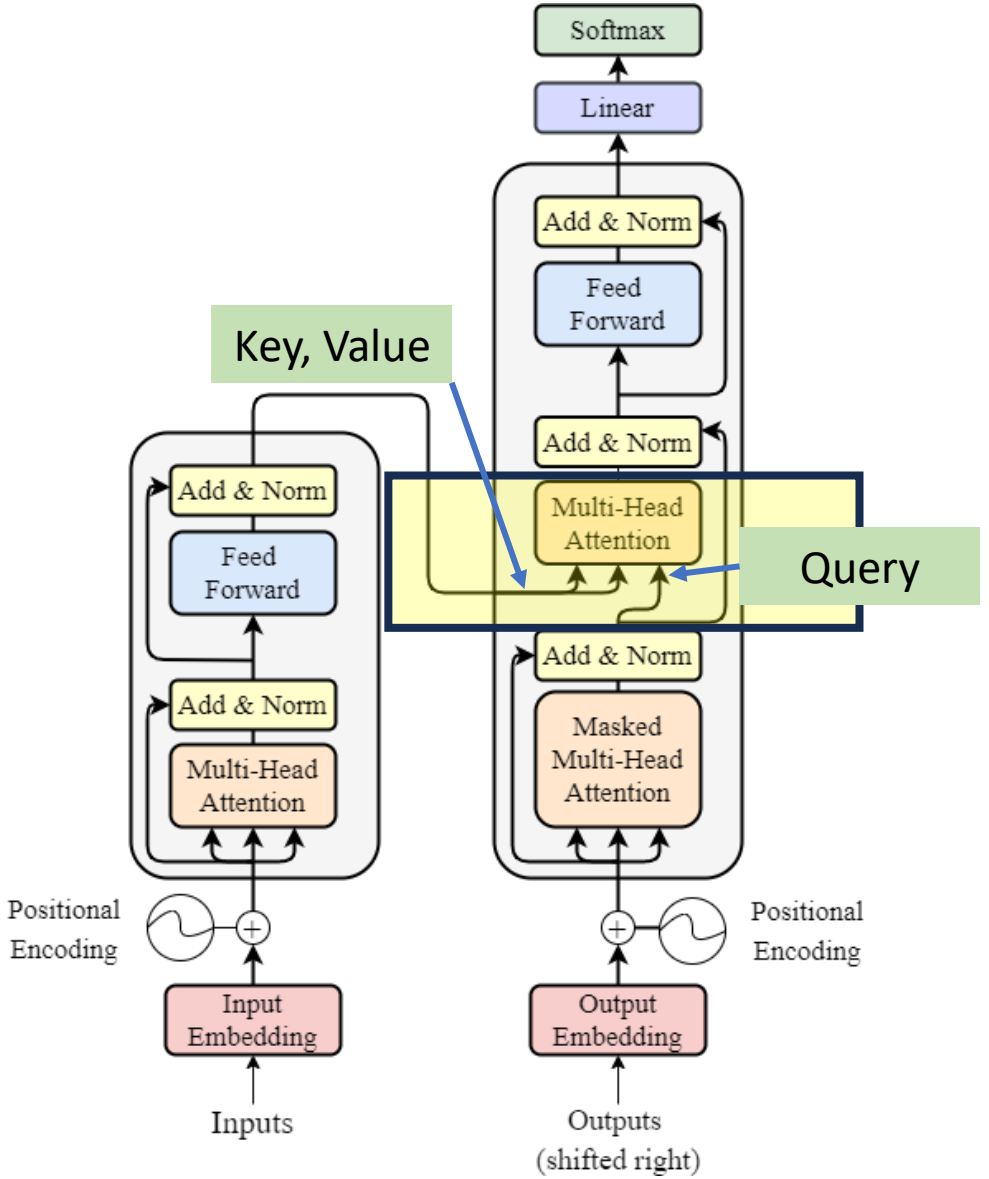
Causal Model: The model must not be able to see the **future** words

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

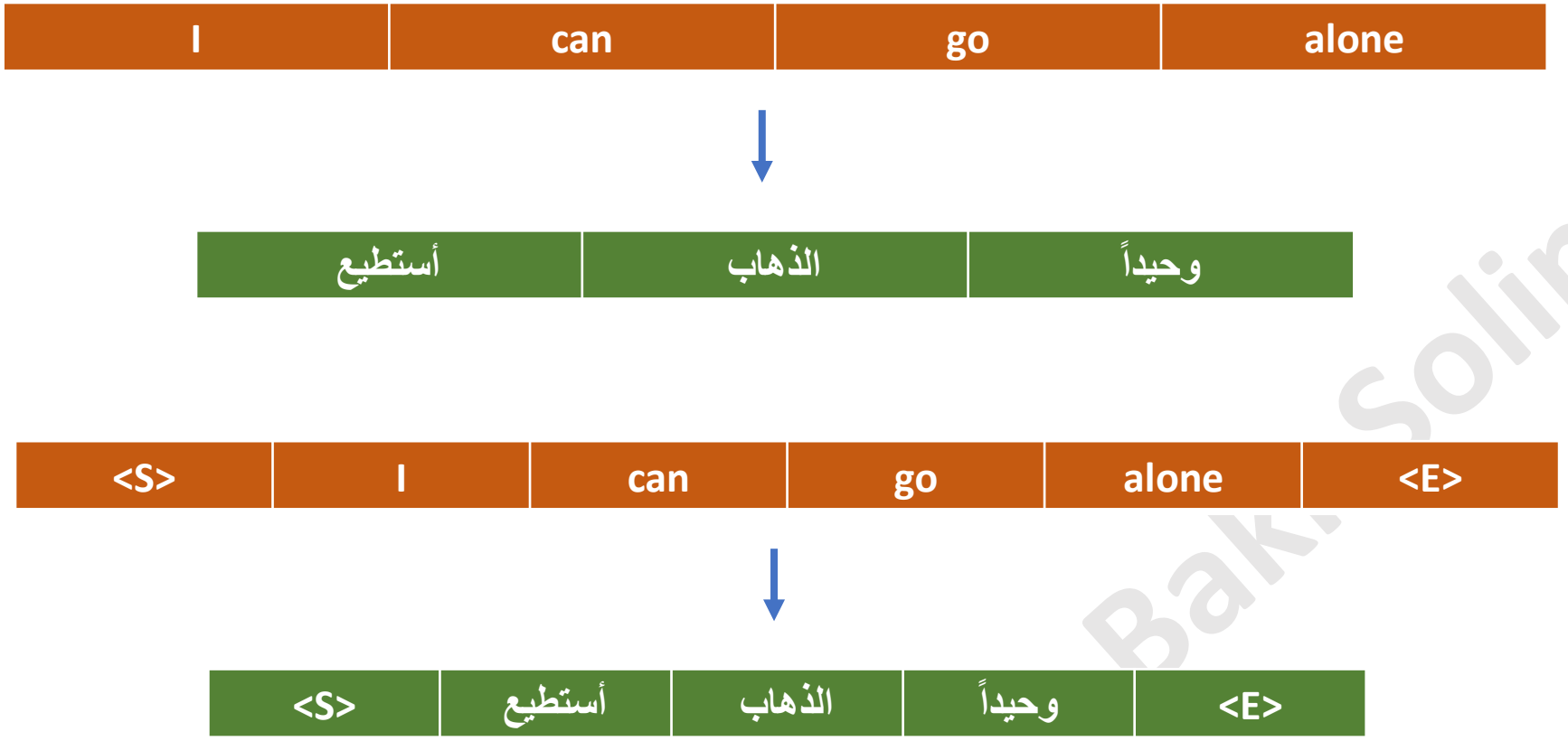
$$\left(\frac{QK^T}{\sqrt{d_k}}\right)$$

	I	can	go	alone
I	63.3	1.2	2.6	7.2
can	3.25	0.3	1.2	2.1
go	12.0	11.9	52.9	2.9
alone	1.6	63.1	14.2	101.3

	I	can	go	alone
I	63.3	-∞	-∞	-∞
can	3.25	96.1	-∞	-∞
go	12.0	11.9	52.9	-∞
alone	1.6	63.1	14.2	101.3

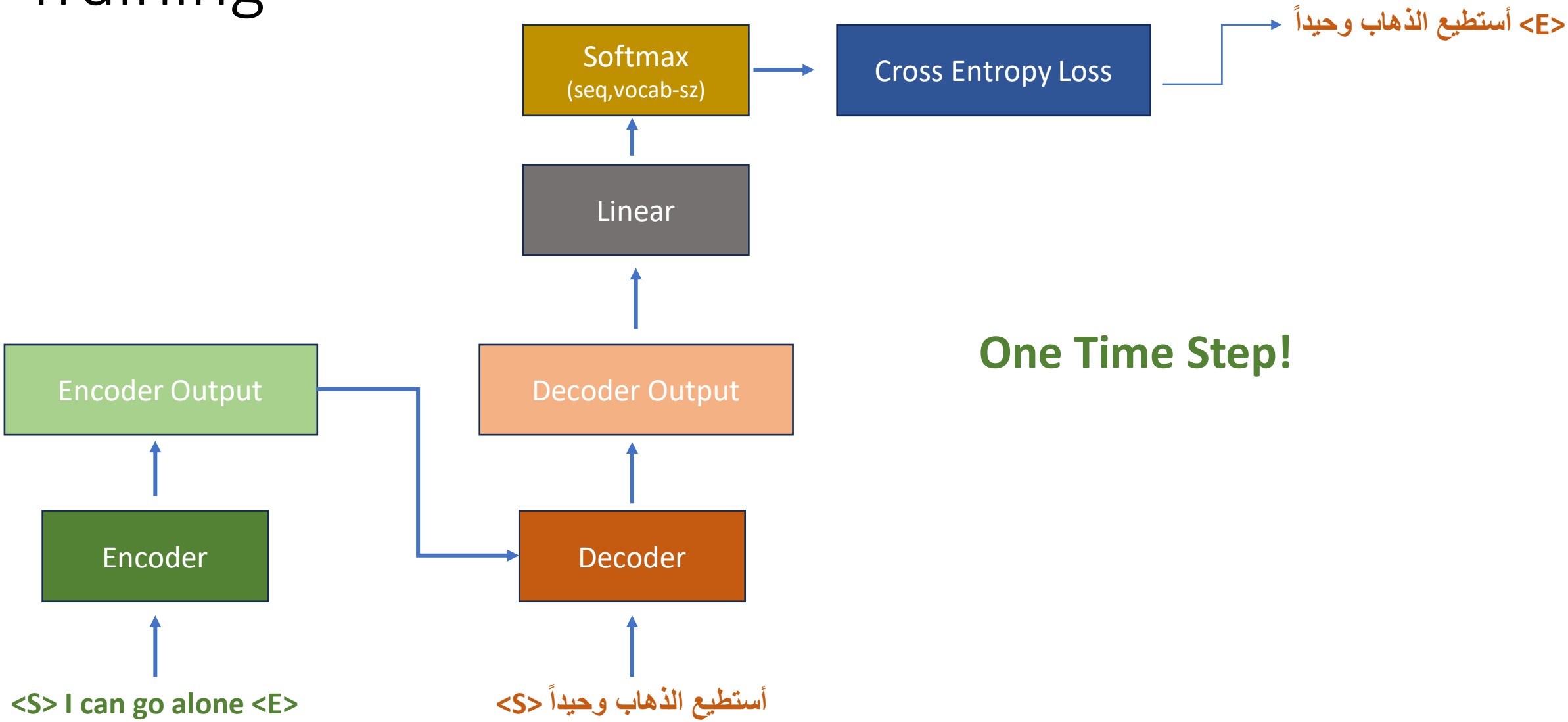

Softmax
Linear
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
Add & Norm
Masked Multi-Head Attention
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
Key, Value
Query
Positional Encoding
Positional Encoding
Input Embedding
Output Embedding
Inputs
Outputs (shifted right)

Training



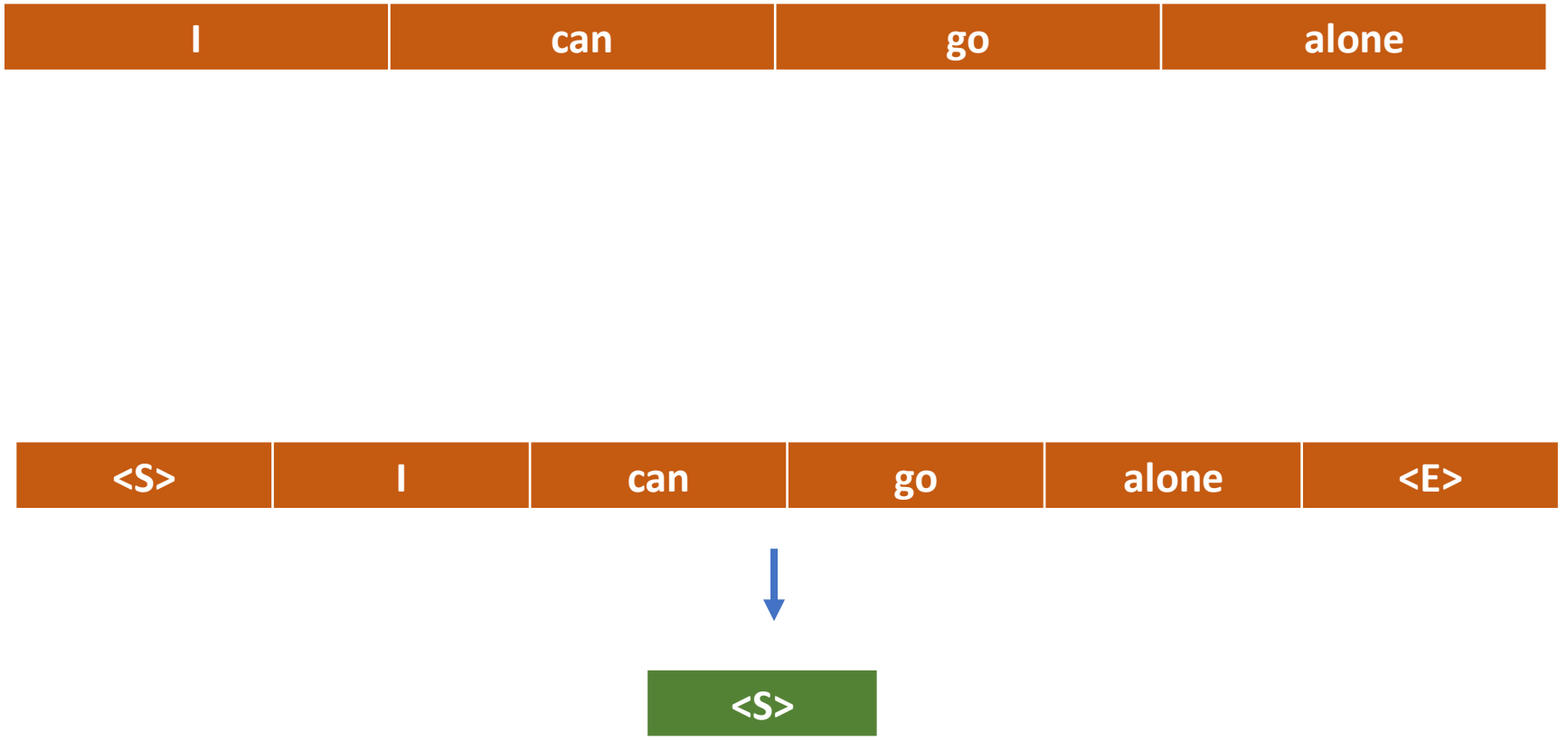
Abu Bakr Soliman

Training

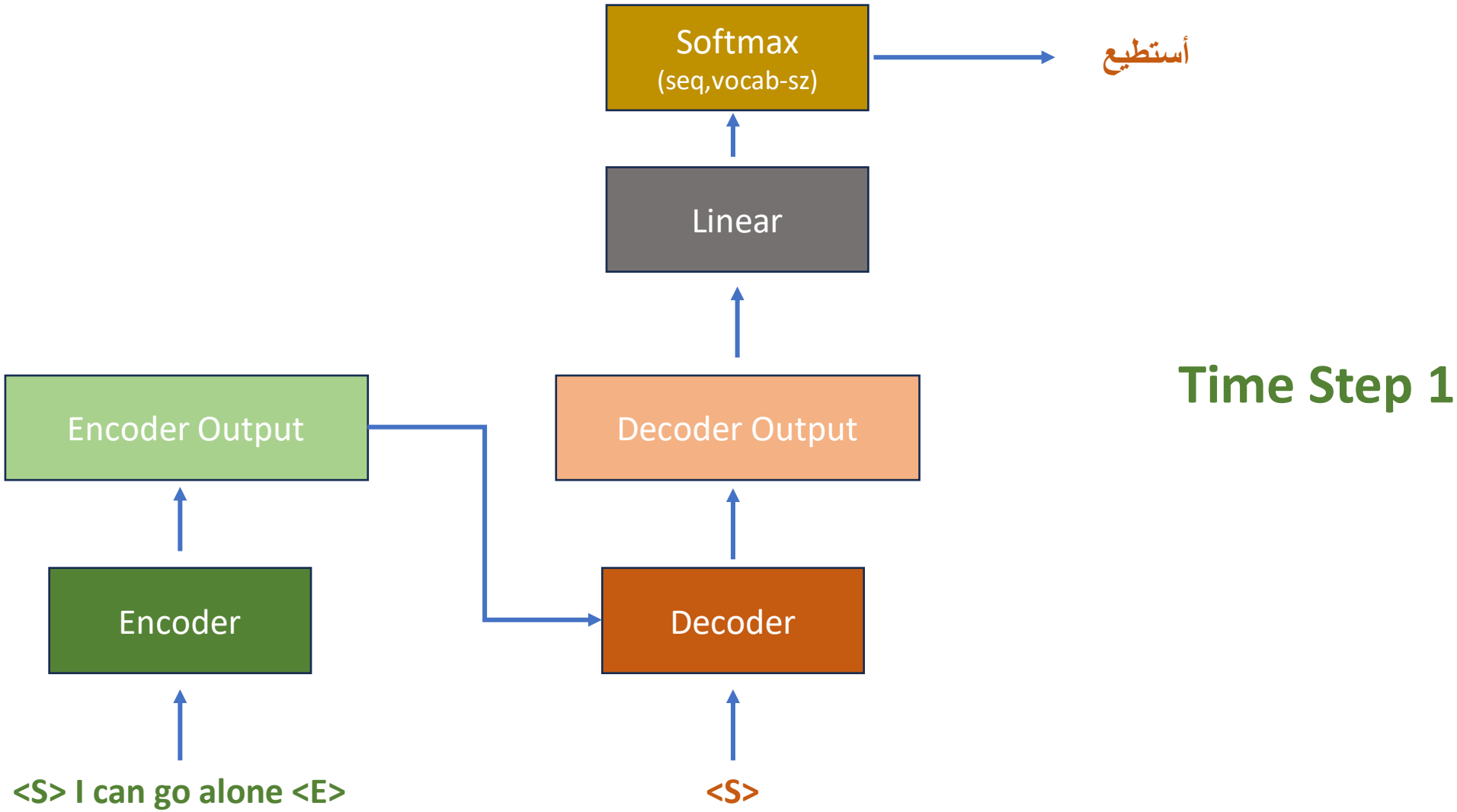


One Time Step!

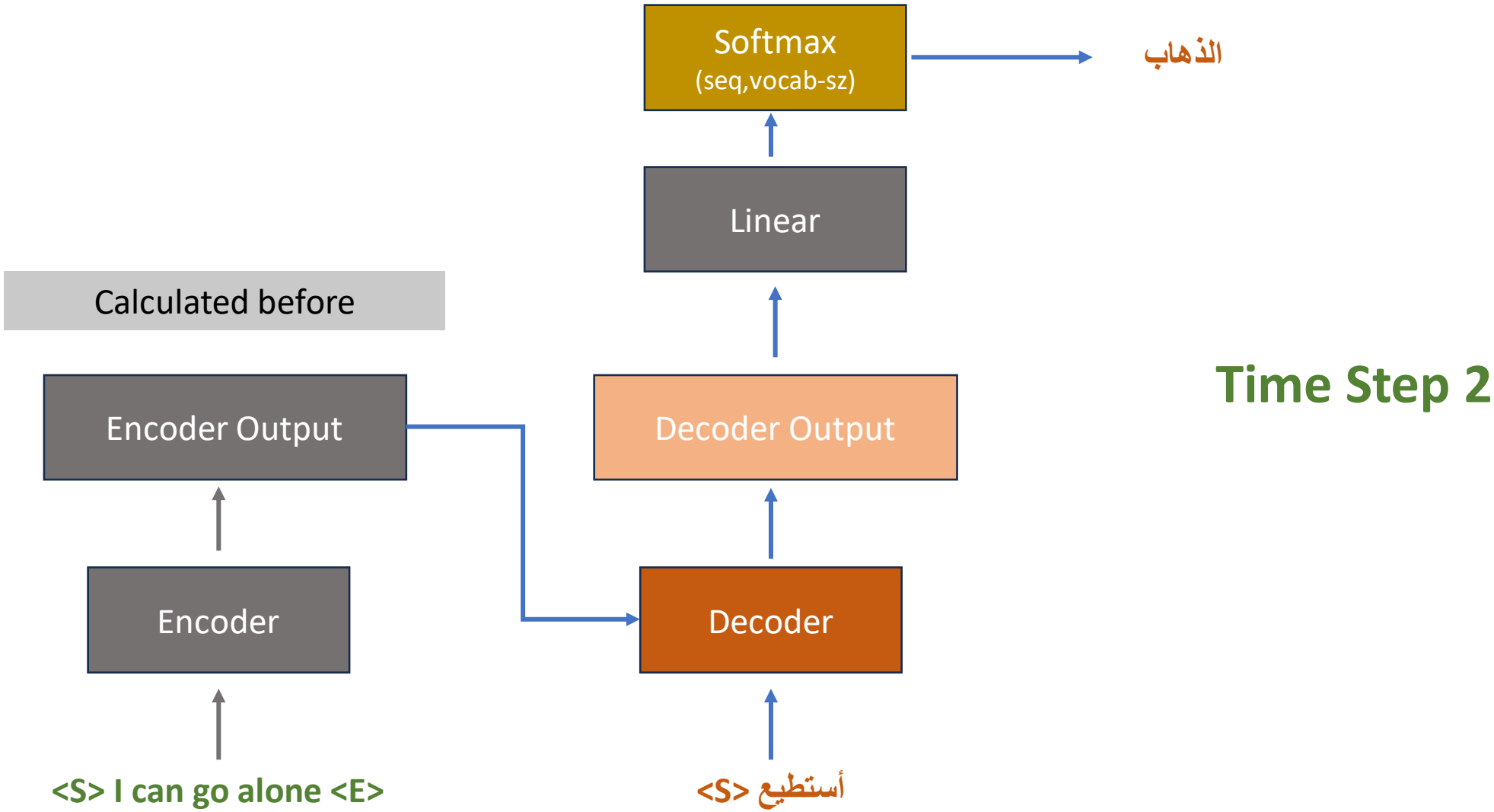
Inference



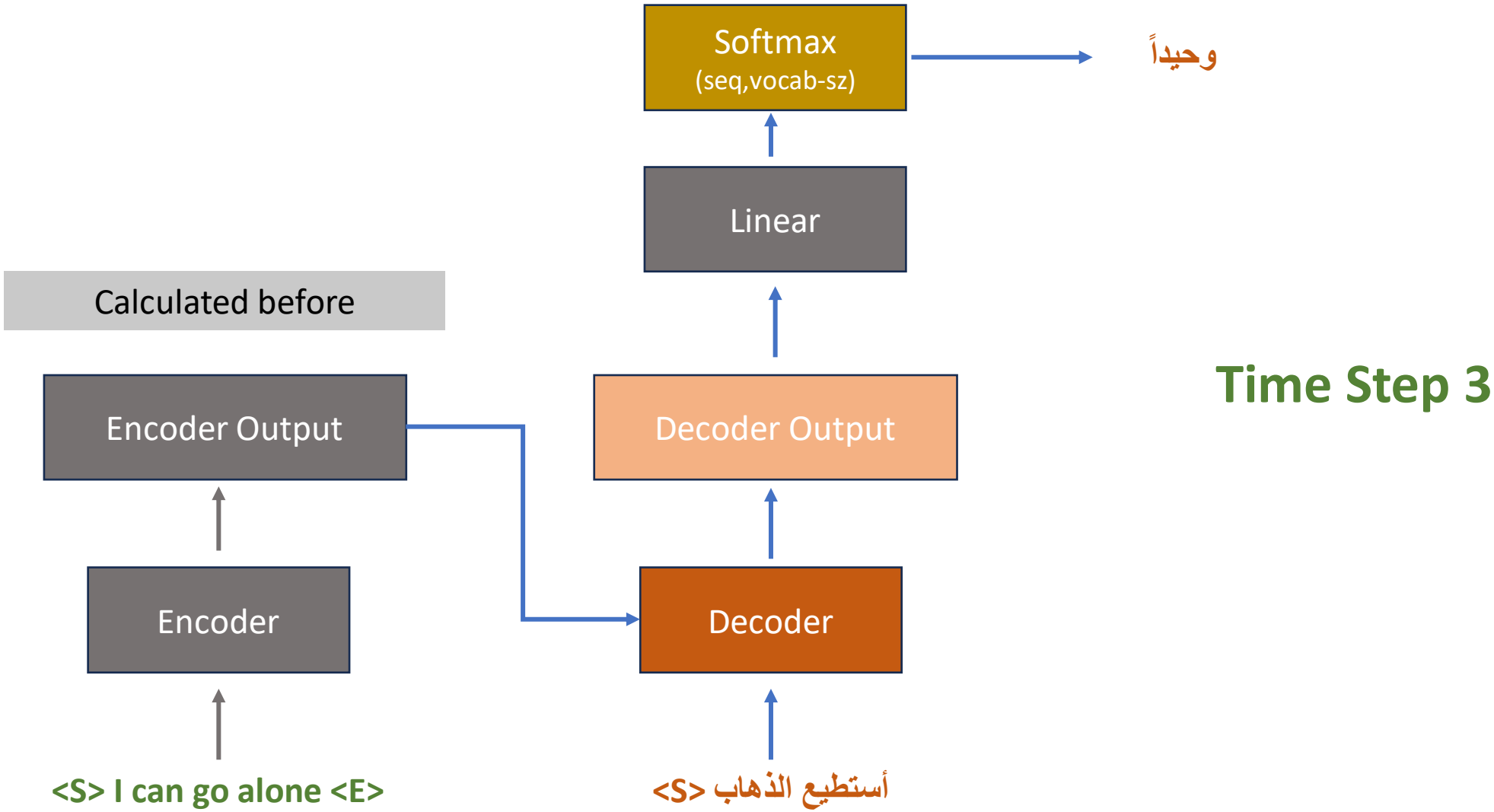
Inference


Softmax
(seq,vocab-sz)
أستطيع
Linear
Encoder Output
Decoder Output
Time Step 1
Encoder
Decoder
<S> I can go alone <E>
<S>

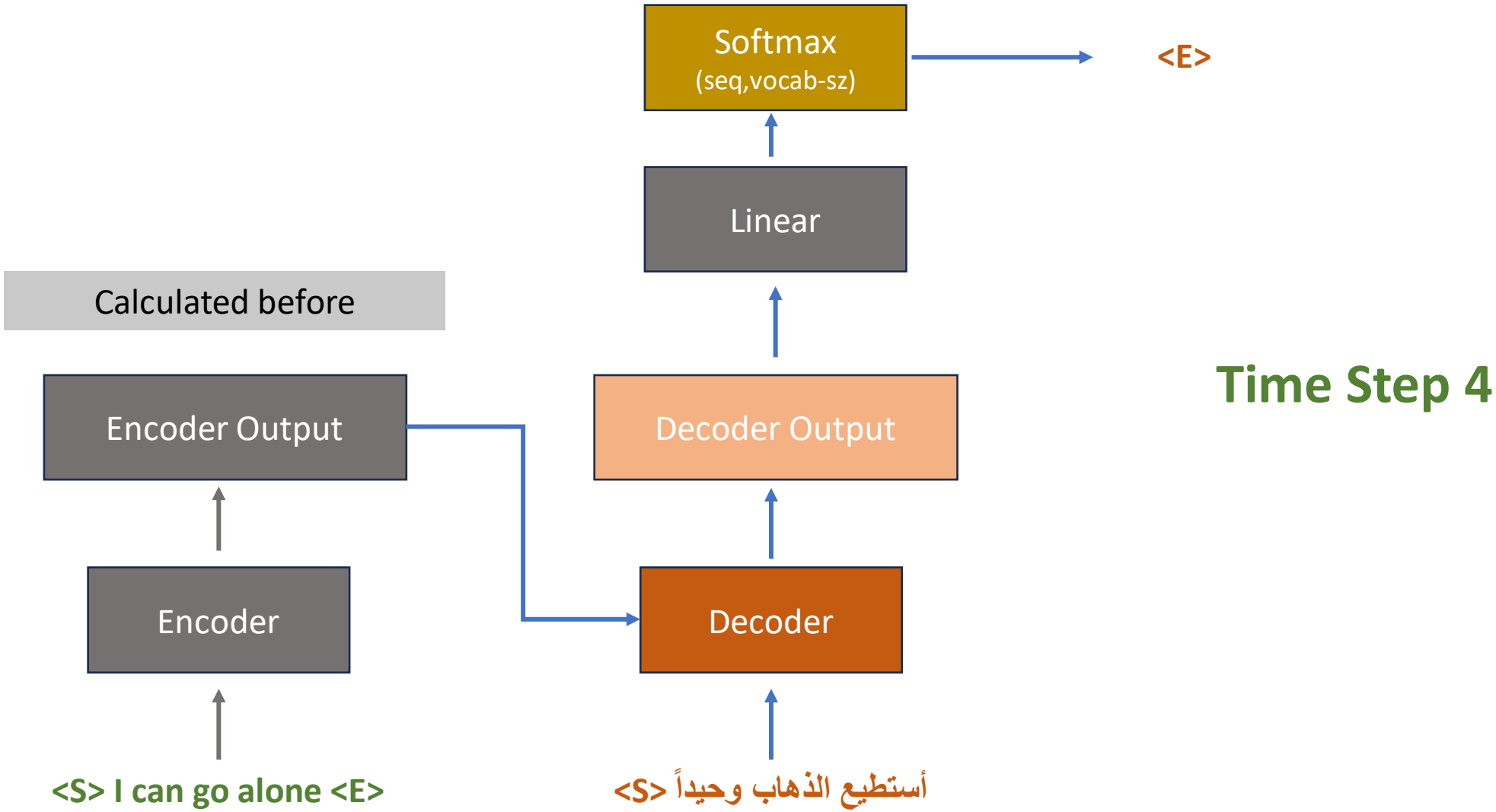
Inference


Softmax
(seq,vocab-sz)
الذهاب
Linear
Calculated before
Encoder Output
Decoder Output
Time Step 2
Encoder
Decoder
<S> I can go alone <E>
أستطيع <S>

Inference


Softmax
(seq,vocab-sz)
وحيداً
Linear
Calculated before
Encoder Output
Decoder Output
Time Step 3
Encoder
Decoder
<S> I can go alone <E>
أستطيع الذهاب <S>

Inference


Softmax
(seq,vocab-sz)
<E>
Linear
Calculated before
Encoder Output
Decoder Output
Time Step 4
Encoder
Decoder
<S> I can go alone <E>
<S> أستطيع الذهاب وحيداً

How to generate text: using different decoding methods for language generation with Transformers

Published March 1, 2020

Update on GitHub

patrickvonplaten
Patrick von Platen

Open in Colab

Follow Me

https://www.linkedin.com/in/bakrianoo/