Attention

is all you need

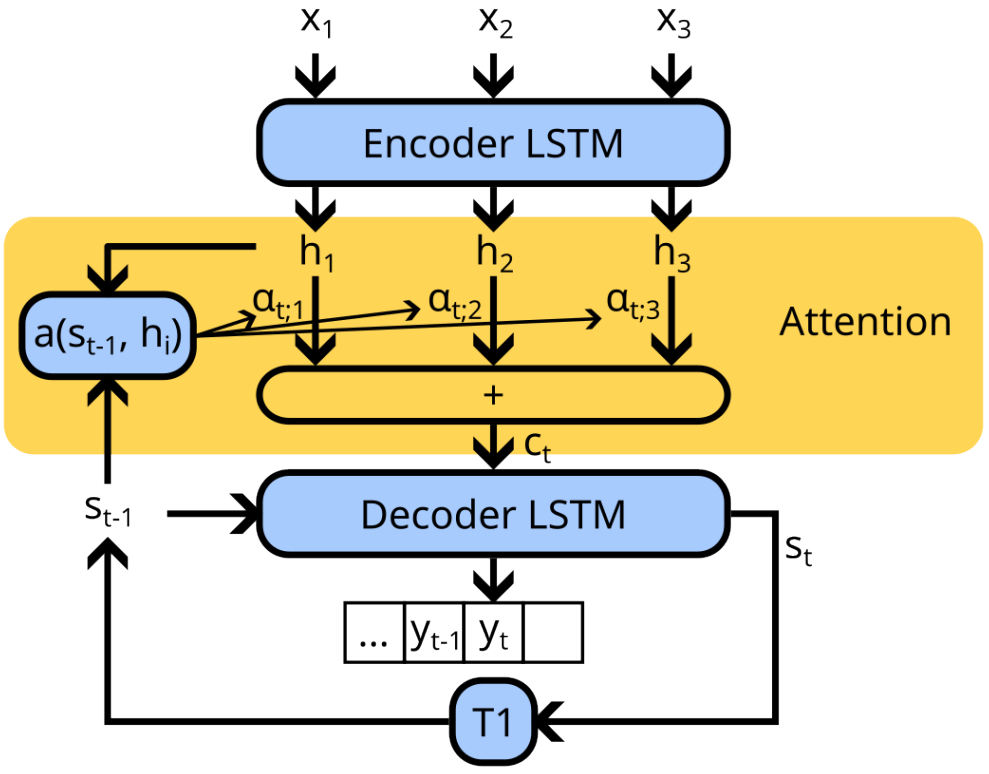
Ascii Paper Reading Group

Goal

- Machine Translation
  - requires alignment of source with output sentence
- Speed up training - Parallel calculation

Previous Works

Attention as function of the input and the output state

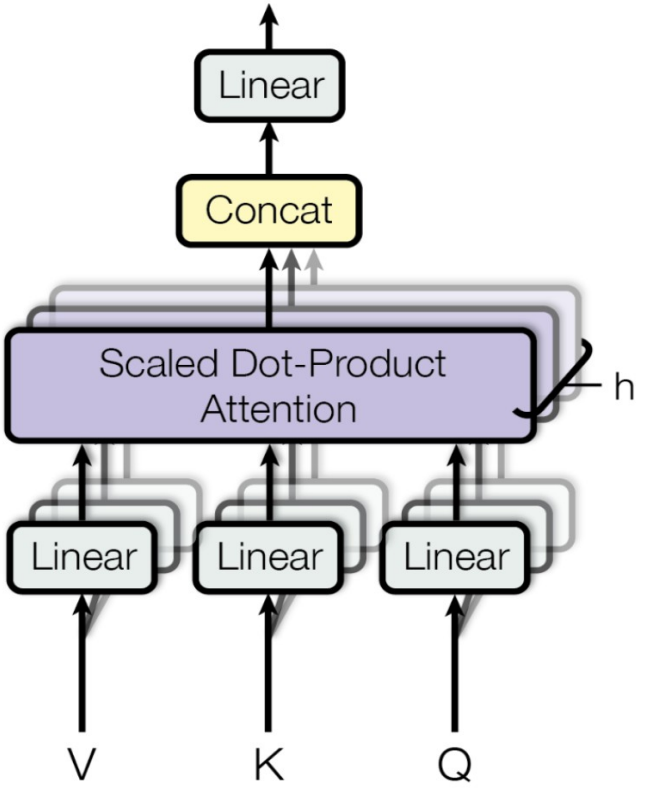


Cool: ability to re-align in and output

Not Cool: long paths during back propagation due to RNNs

Alignment using

Multi Head Attention



Focus: Ad hoc Retrieval

Dresden Database Research Group

Given: query *q*

collection **of texts**

Return: a ranked list of *k* texts $d_1 \dots d_k$

Maximizing: a metric of interest

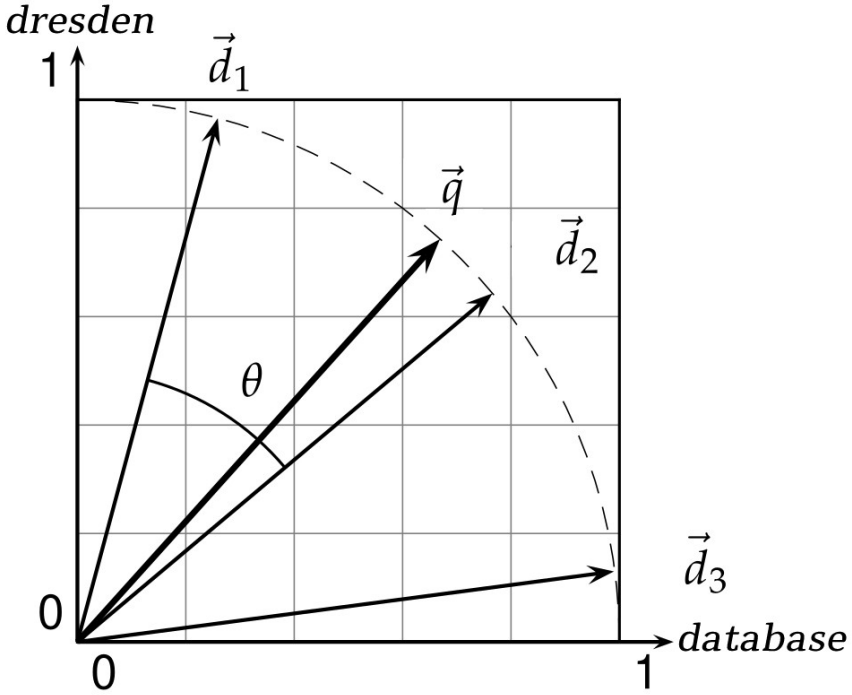
Information Retrieval Basics I

Dresden Database Research Group

Vector Space Model

- queries *q* + documents *d* represented as vectors
- use cosine similarity between query and documents for ranking

$$\text{sim}(d_1, d_2) = \cos\theta = \frac{\vec{d_1} \cdot \vec{d_2}}{\left|\vec{d_1}\right| \left|\vec{d_2}\right|}$$

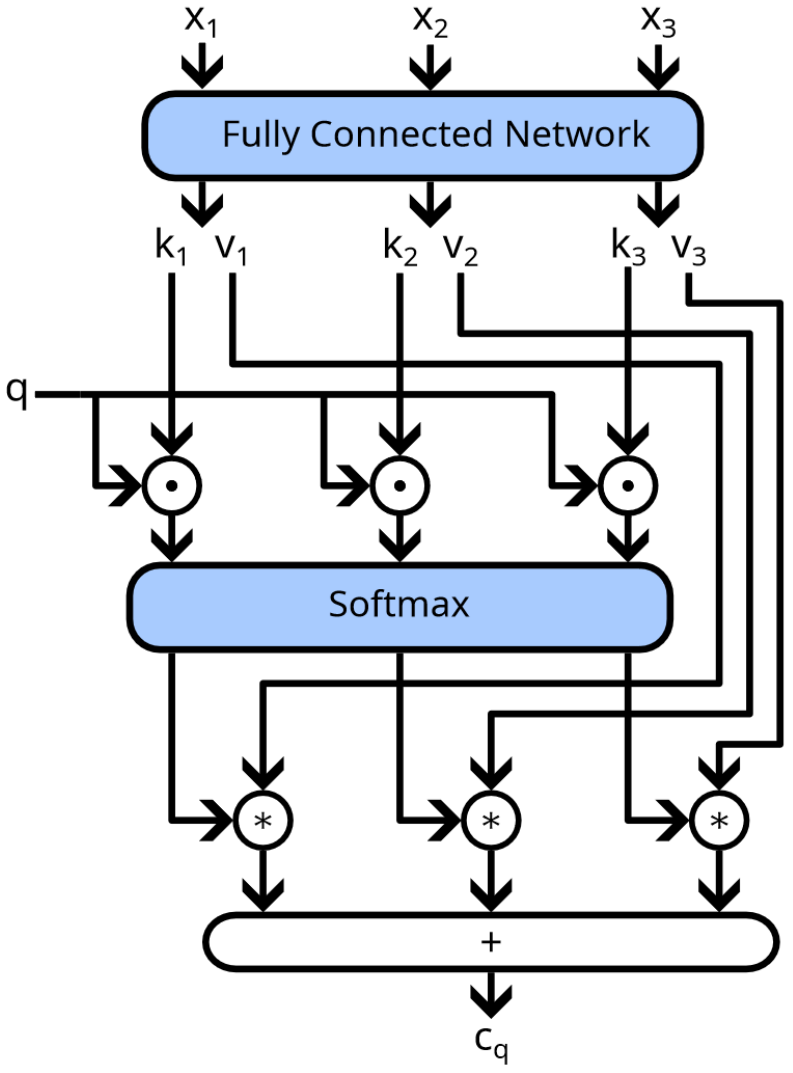


Dot Product Attention

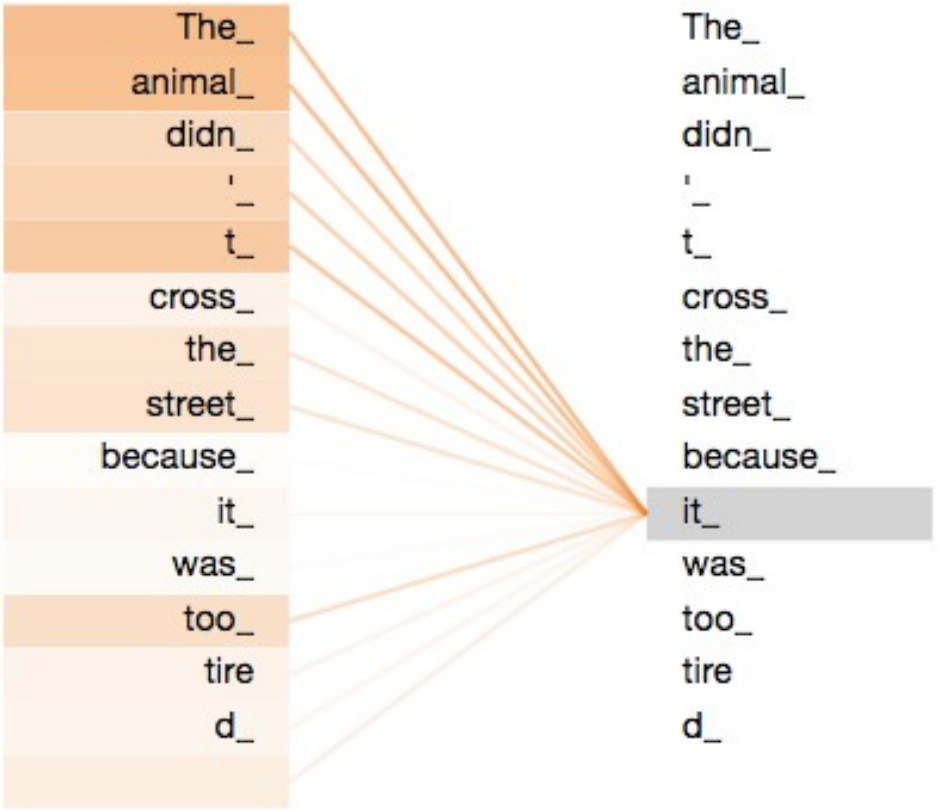
Map series $x_0 \ldots x_n$ using weight matrices W_Q W_K and W_V to q_i, k_i and v_i.

Calculate the dot product between each q and k_i to influence the significance of v_i

Use softmax for normalization of attention



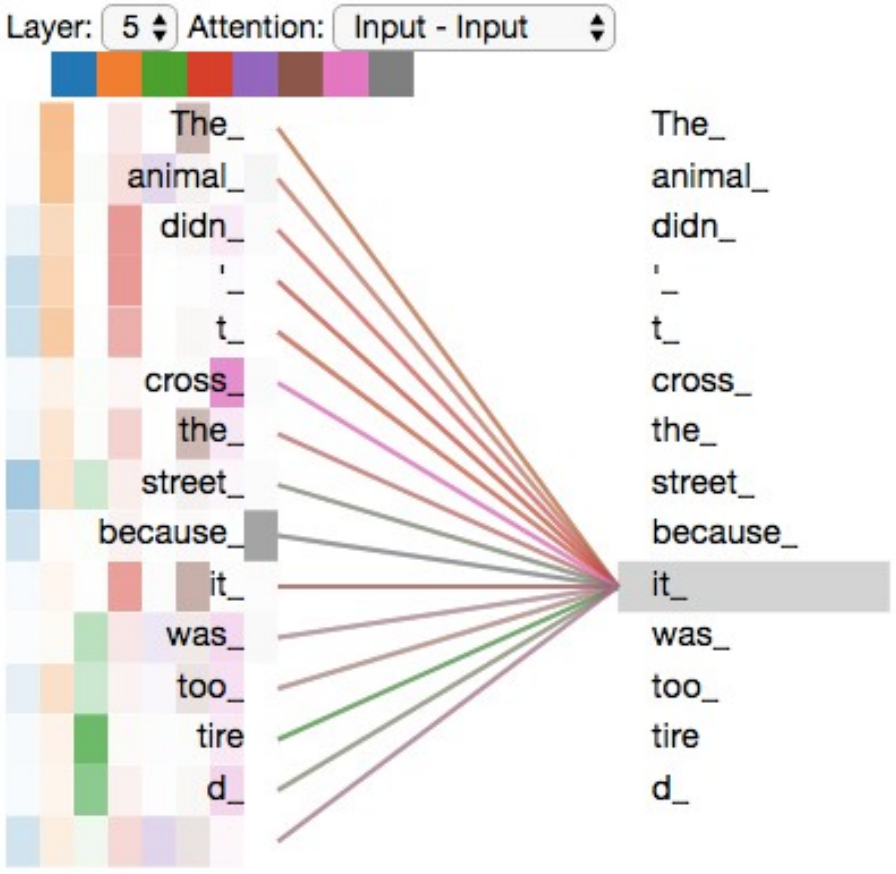
Attention visualized



Goal: collect context information

https://jalammar.github.io/illustrated-transformer/

Multi Head Attention

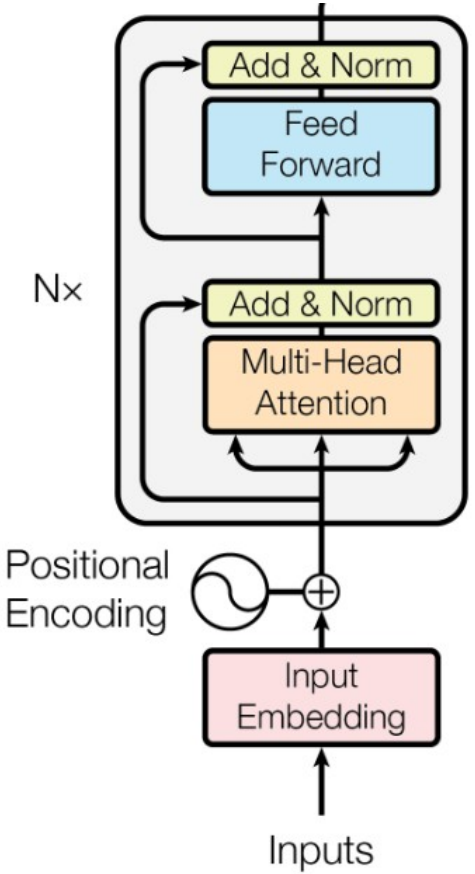


Split input vectors in 8 vectors

Process each series with their own set of weights

Concatenate the results along the vector dimension to restore original shape

Encoder



- Repeated self attention
- Skip connections for gradient flow
- Normalization for
  - gradient flow
  - Cosine distance

Positional Encoding

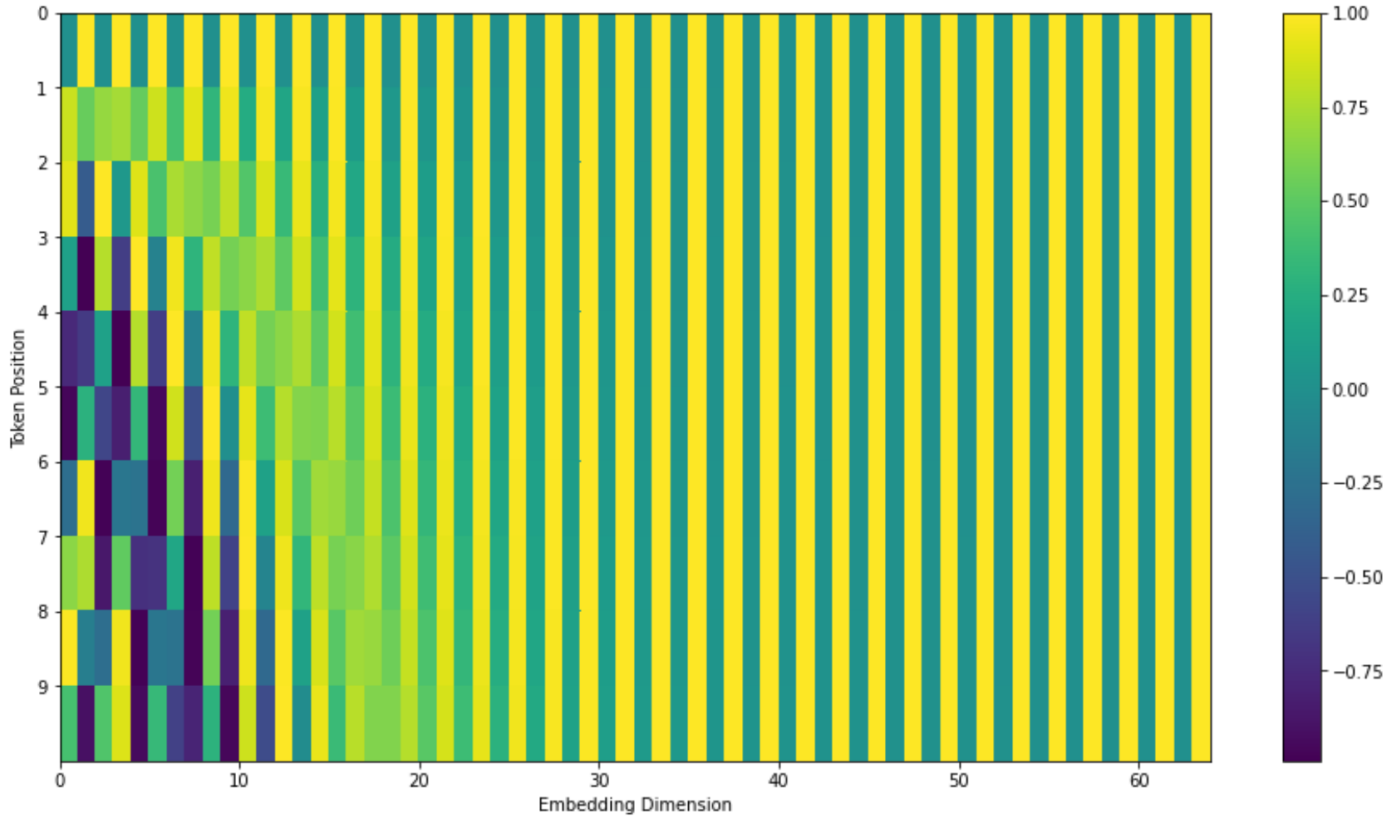
- Problem: Dot-Product attention doesn't consider distance of words in a document.
- Add a positional encoding to each token

$$PE_{(pos,2i)} = sin(pos/10000^{2i/d_{\text{model}}})$$

$$PE_{(pos,2i+1)} = cos(pos/10000^{2i/d_{\text{model}}})$$

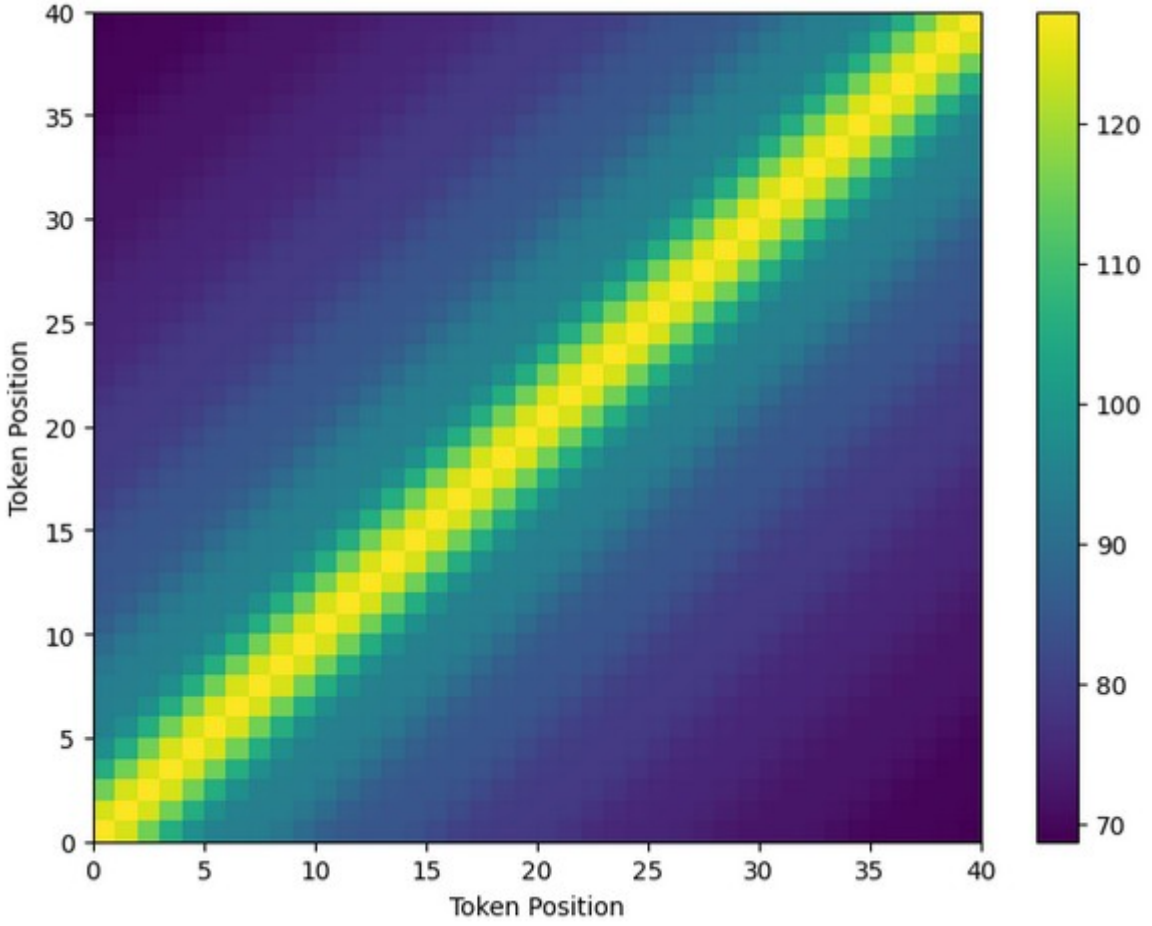
i … dimension

Positional Encoding



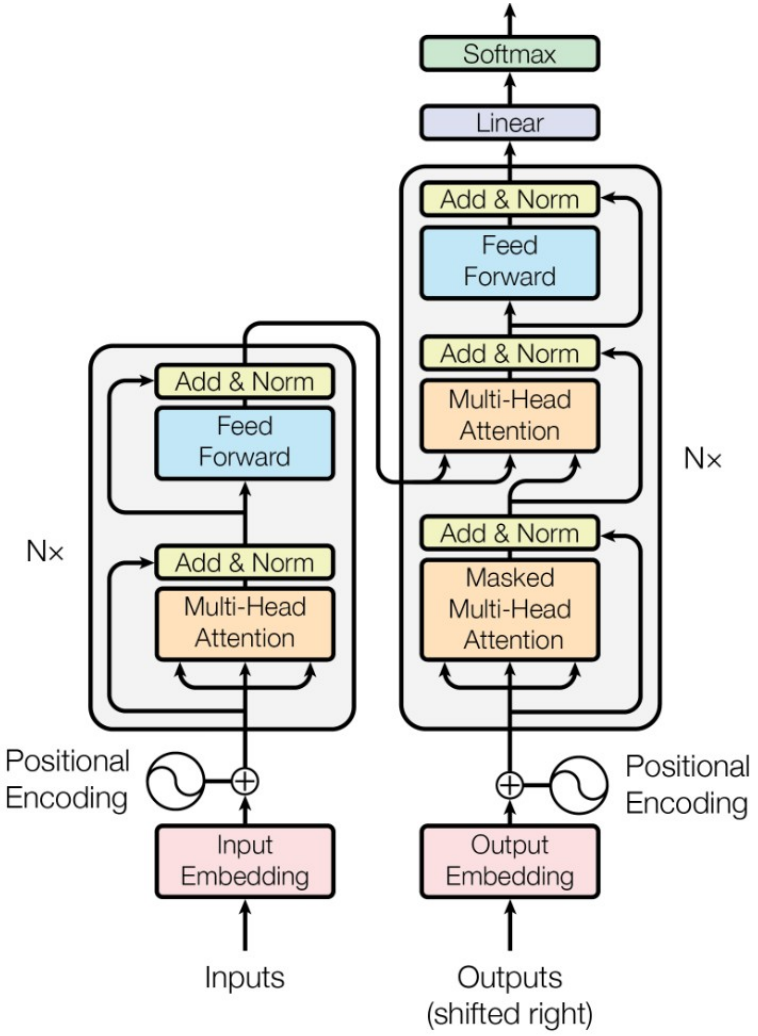
https://jalammar.github.io/illustrated-transformer/

Positional Embedding

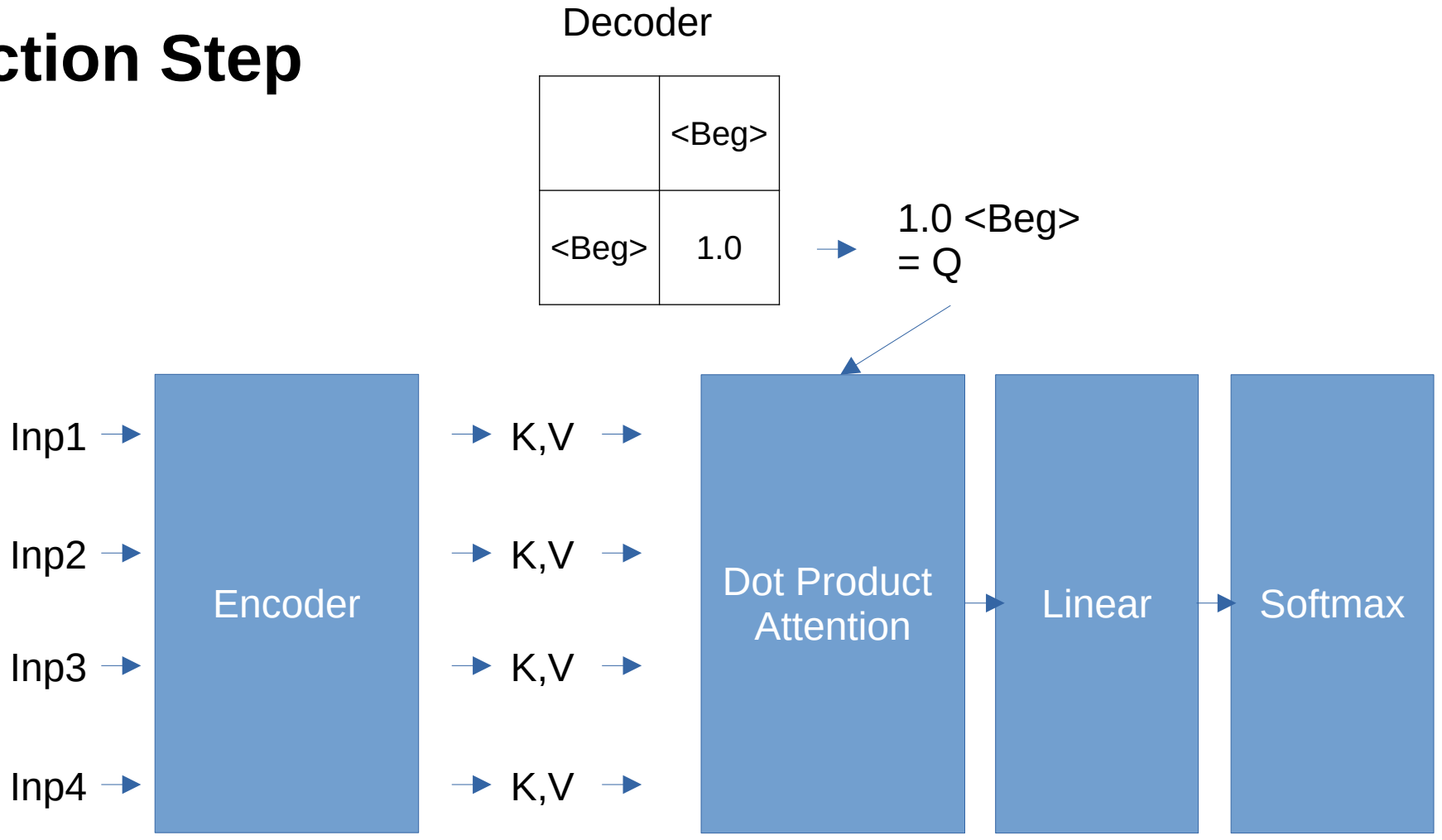


$$z = PE_x \odot PE_y$$

Decoder


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

Prediction Step

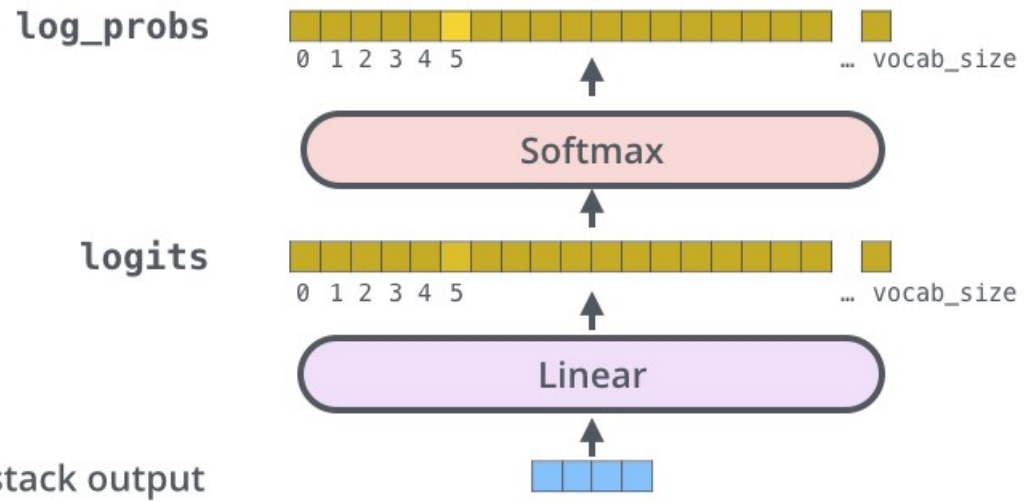


Which word in our vocabulary is associated with this index?

am

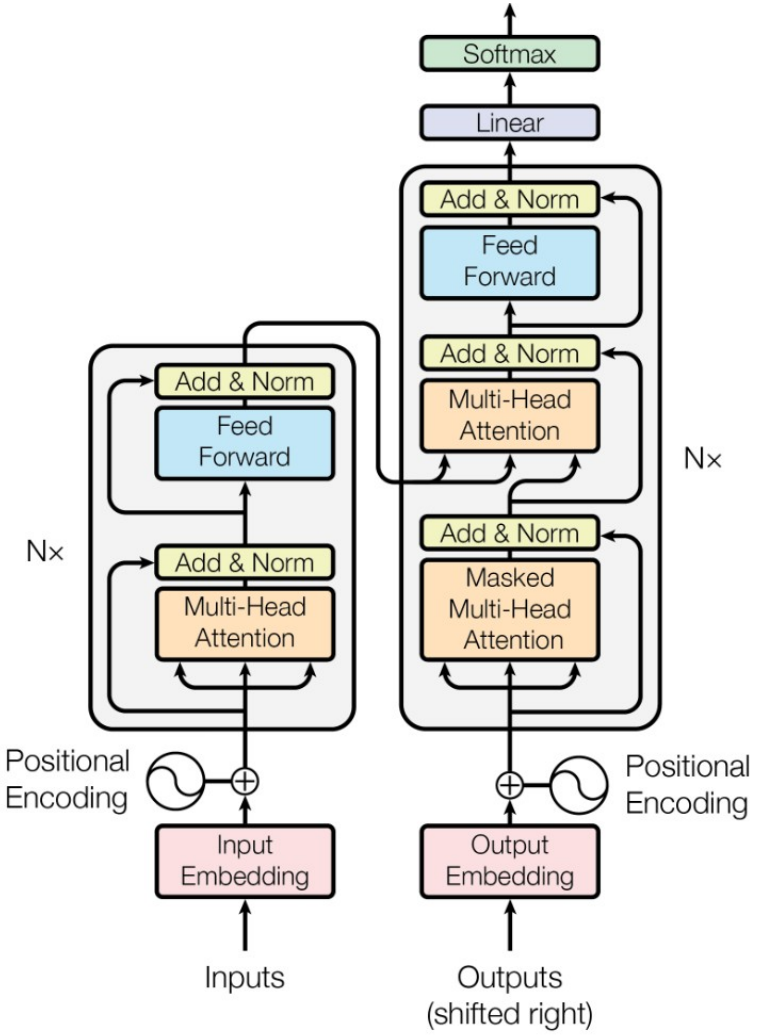
Get the index of the cell with the highest value **(argmax)**

5

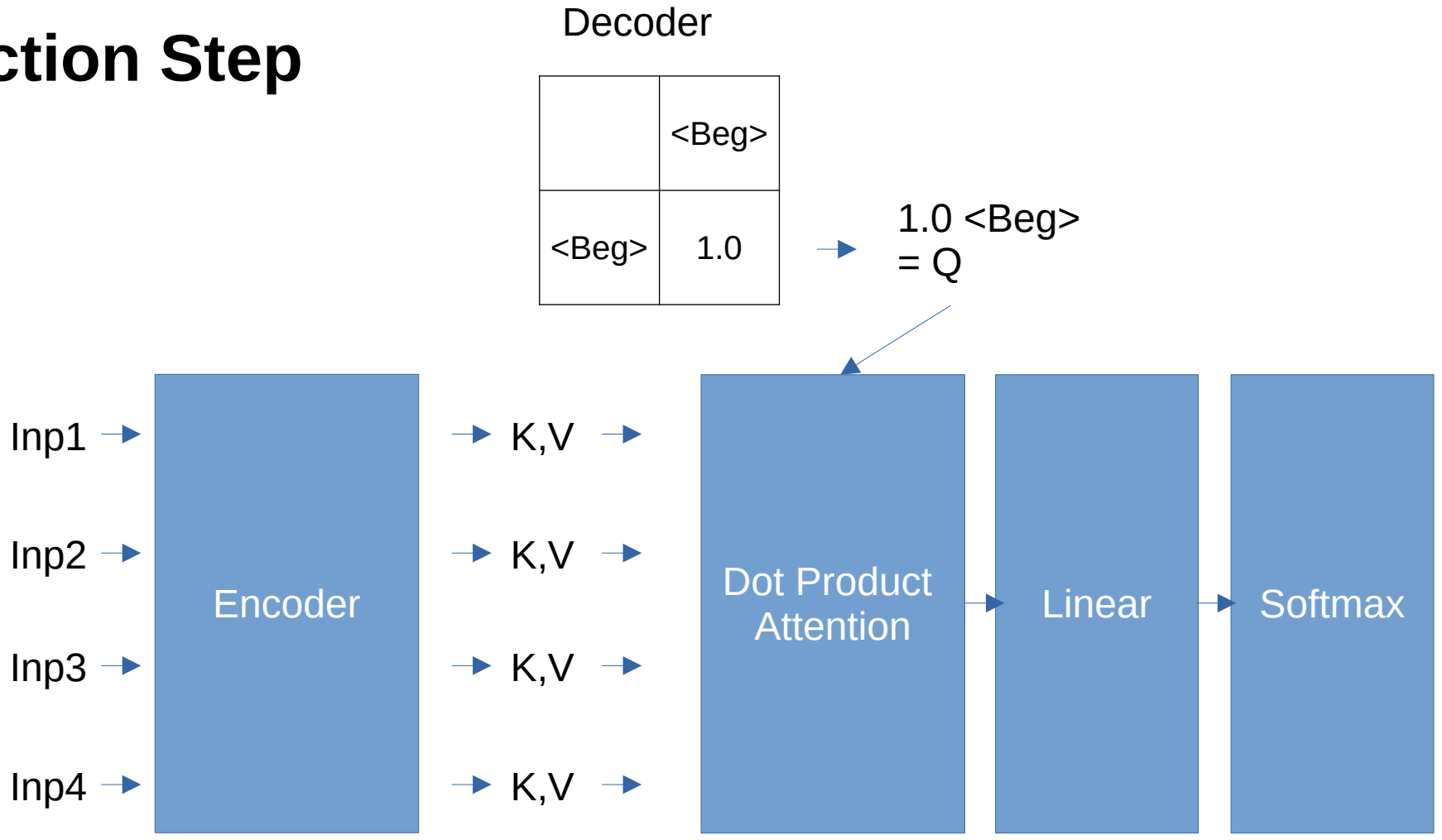


https://jalammar.github.io/illustrated-transformer/

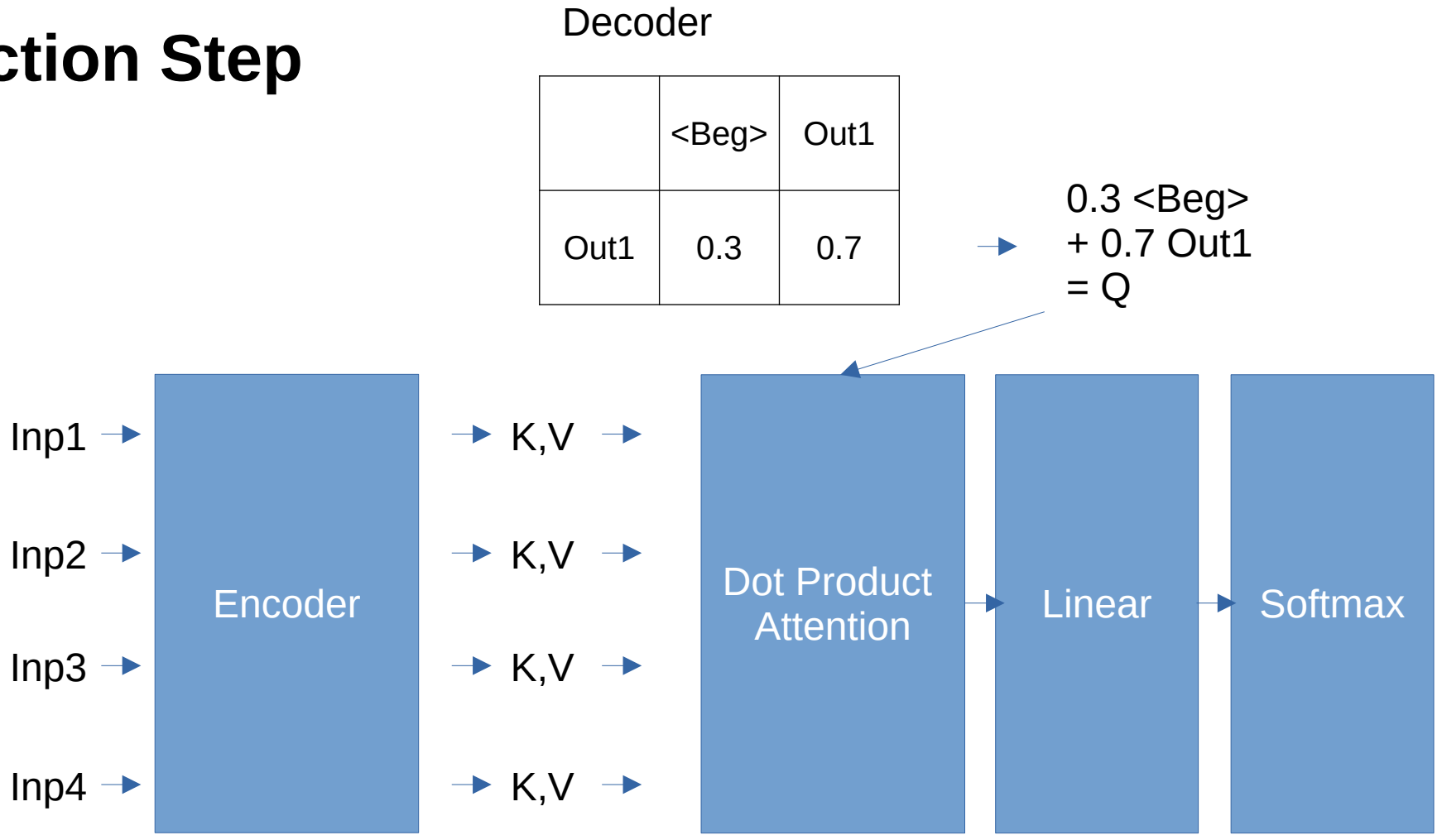
Decoder


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

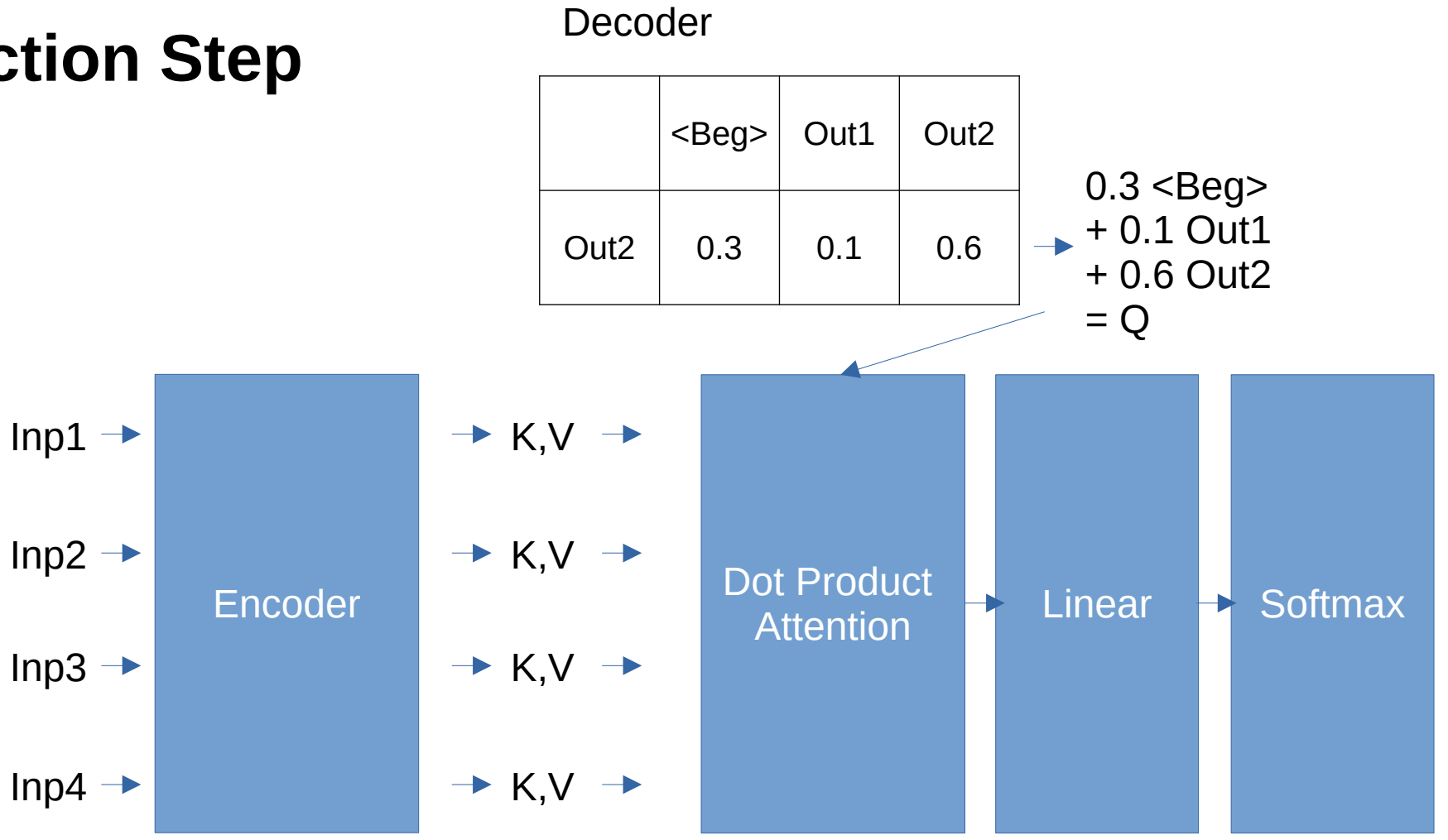
Prediction Step



Prediction Step



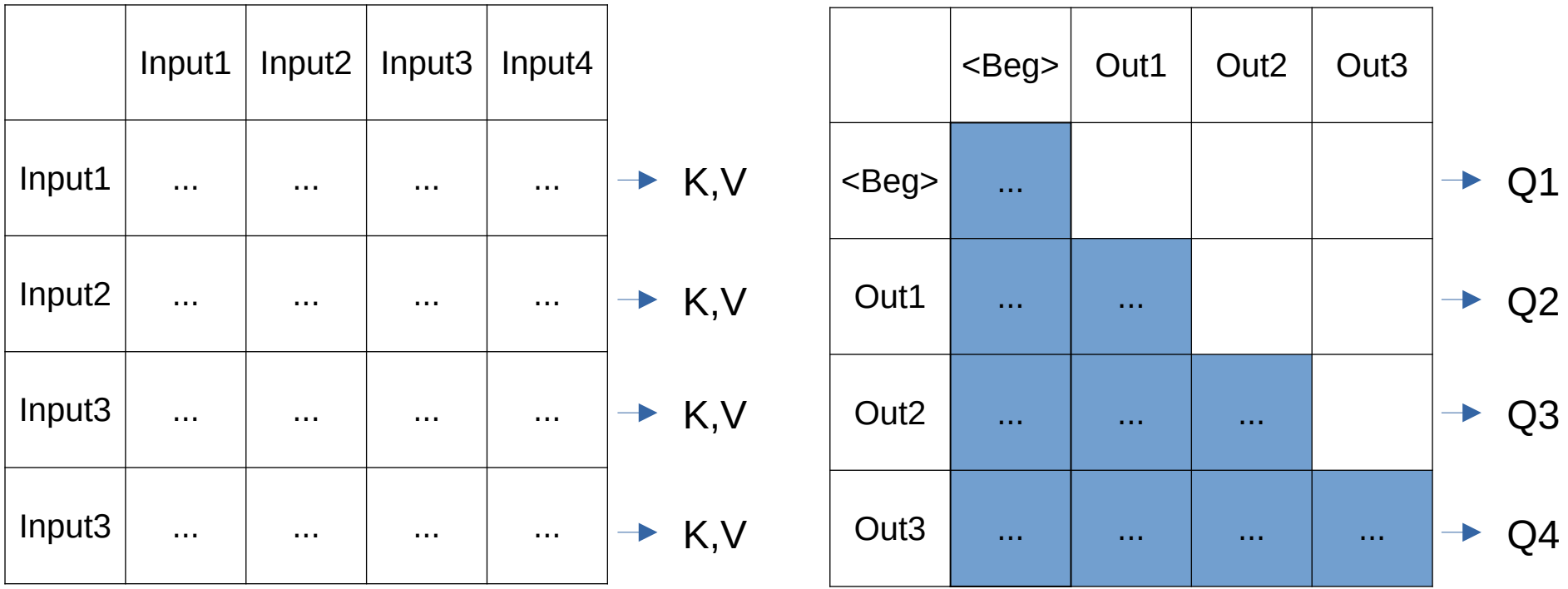
Prediction Step



Training Optimization

- What we saw before
  - Iterative prediction
- During training we know the target sentence
  - We can make all predictions in parallel

Masked Self Attention



-> parallel prediction of all tokens (training only)

Benchmark

Model	BLEU		Training Cost (FLOPs)	
	EN-DE	EN-FR	EN-DE	EN-FR
ByteNet [18]	23.75			
Deep-Att + PosUnk [39]		39.2		$1.0 \cdot 10^{20}$
GNMT + RL [38]	24.6	39.92	$2.3 \cdot 10^{19}$	$1.4 \cdot 10^{20}$
ConvS2S [9]	25.16	40.46	$9.6 \cdot 10^{18}$	$1.5 \cdot 10^{20}$
MoE [32]	26.03	40.56	$2.0 \cdot 10^{19}$	$1.2 \cdot 10^{20}$
Deep-Att + PosUnk Ensemble [39]		40.4		$8.0 \cdot 10^{20}$
GNMT + RL Ensemble [38]	26.30	41.16	$1.8 \cdot 10^{20}$	$1.1 \cdot 10^{21}$
ConvS2S Ensemble [9]	26.36	**41.29**	$7.7 \cdot 10^{19}$	$1.2 \cdot 10^{21}$
Transformer (base model)	27.3	38.1	**$3.3 \cdot 10^{18}$**	
Transformer (big)	**28.4**	**41.8**	$2.3 \cdot 10^{19}$	

BLEU

Scores are calculated for individual translated segments—generally sentences—by comparing them with a set of good quality reference translations. Those scores are then averaged over the whole corpus to reach an estimate of the translation's overall quality. Intelligibility or grammatical correctness are not taken into account.