Self-Attention

- Self-attention: builds on the idea of attention. **Every word in a sequence is both a key and a query simultaneously**

Q: seq len x d matrix (d = embedding dimension = 2 for these slides)

K: seq len x d matrix

$$W^Q = \begin{matrix} 0 & 1 \\ 0 & 1 \end{matrix}$$

no matter what the value is, we're going to look for Bs

$$W^K = \begin{matrix} 10 & 0 \\ 0 & 10 \end{matrix}$$

"booster" as before

Note: there are many ways to set up these weights that will be equivalent to this

Self-Attention

$$E = \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} \qquad W^Q = \begin{matrix} 0 & 1 \\ 0 & 1 \end{matrix} \qquad W^K = \begin{matrix} 10 & 0 \\ 0 & 10 \end{matrix}$$

Self-Attention (Vaswani et al.)

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

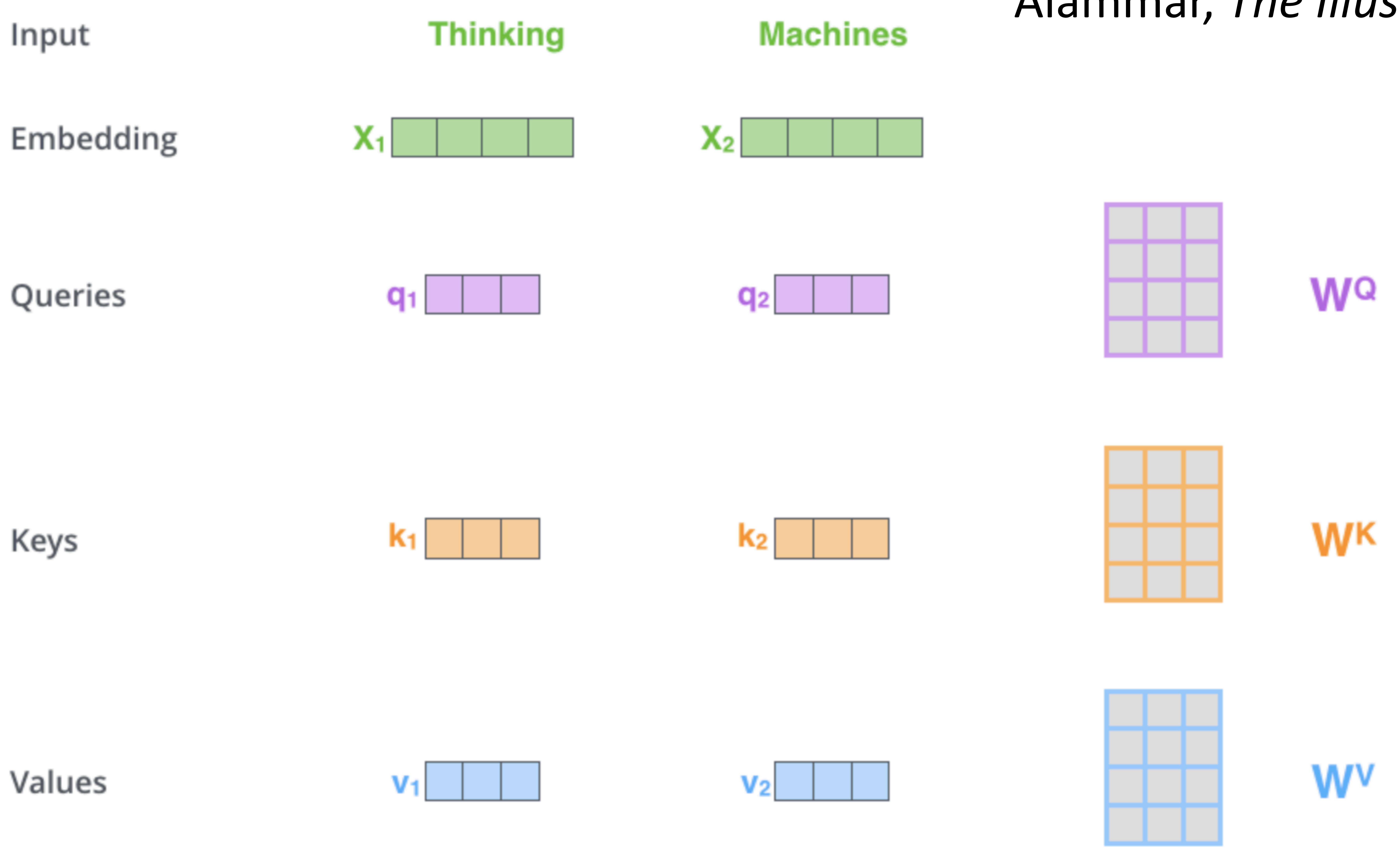
$Q = EW^Q$, $K = EW^K$, $V = EW^V$

- Normalizing by $\sqrt{d_k}$ helps control the scale of the softmax, makes it less peaked
- This is just one ***head*** of self-attention — produce multiple heads via randomly initialize parameter matrices (more in a bit)
- What does self-attention produce?
  - Square attention matrix * input = same dimension as the input.
  - Computes a contextualized encoding for each word, preserving the length of the sequence

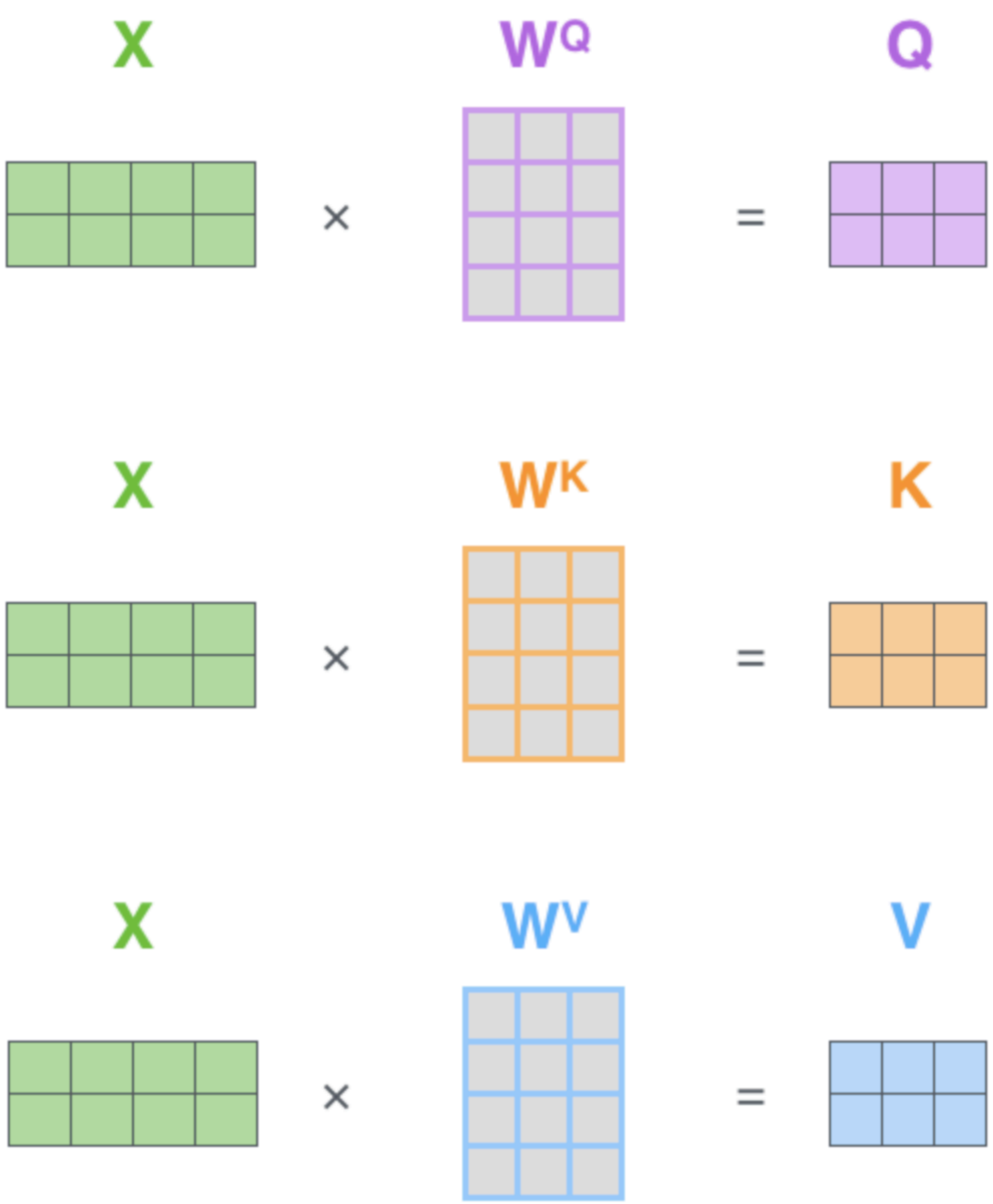
Vaswani et al. (2017)

Self-Attention (Alammar)

Alammar, *The Illustrated Transformer*

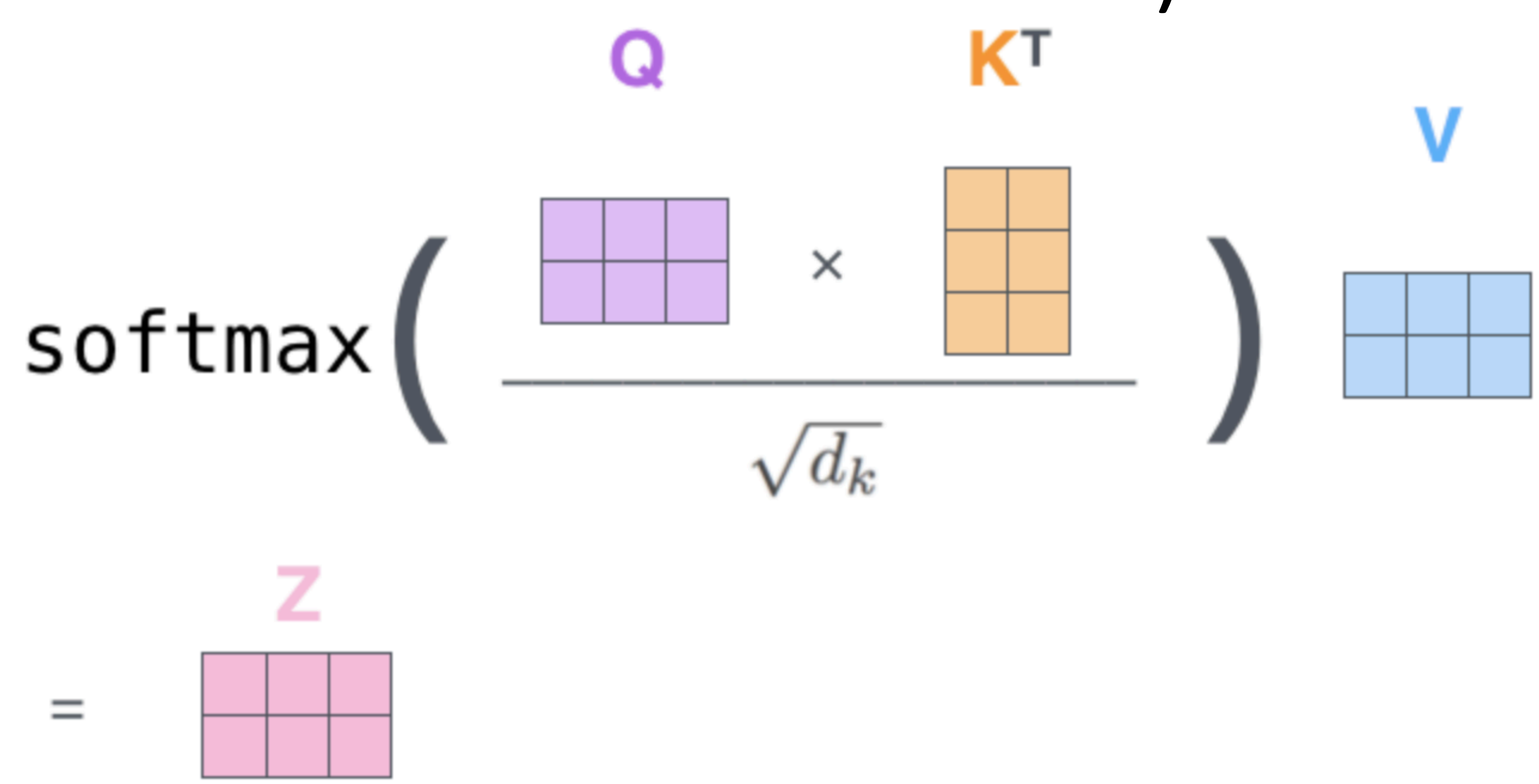


Self-Attention (Alammar)



Alammar, *The Illustrated Transformer*

sent len x sent len (attn for each word to each other)

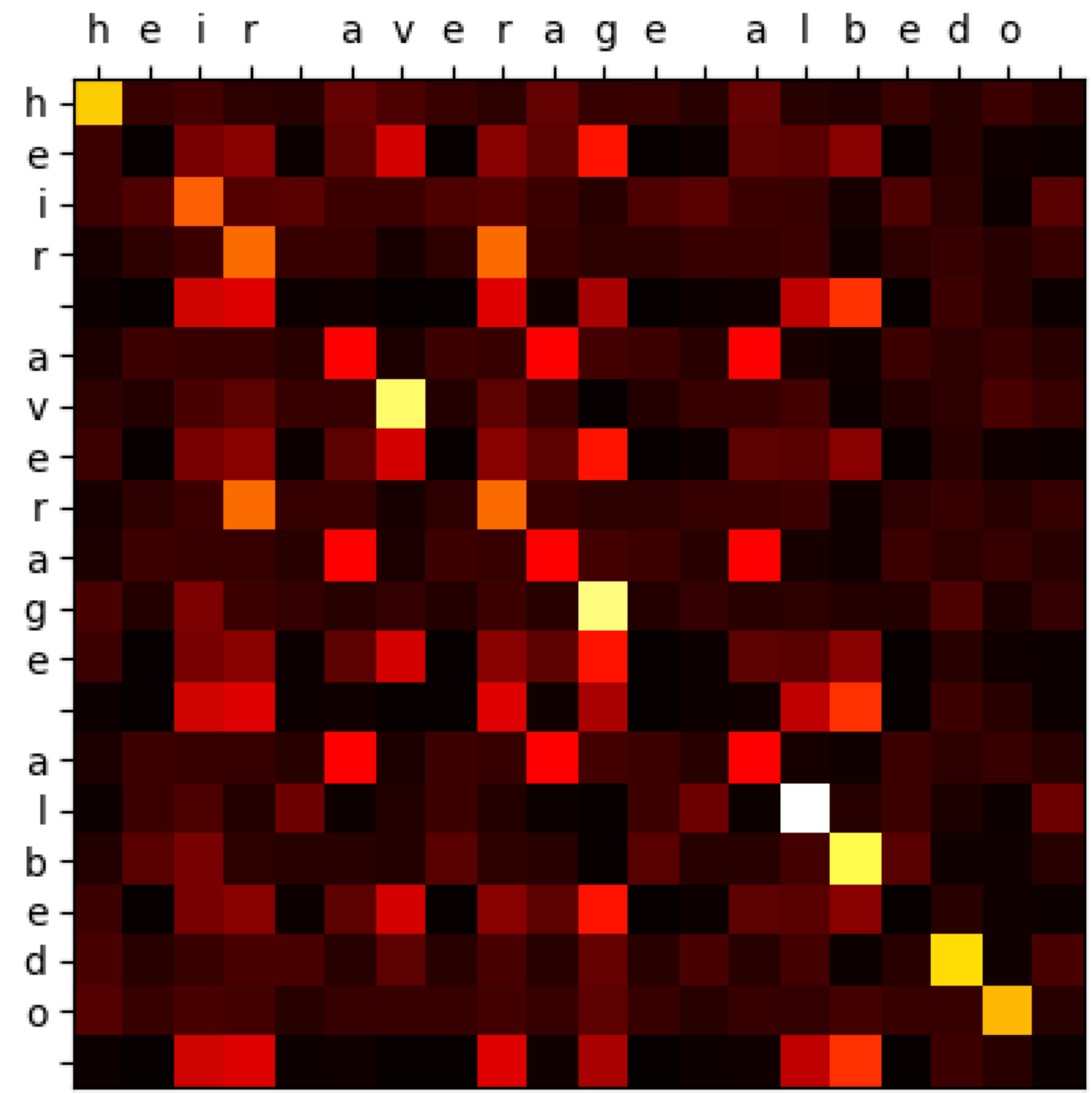


sent len x hidden dim

Z is a weighted combination of V rows

Attention Maps

- Example visualization of attention matrix A (from assignment)
- Each row: distribution over what that token attends to. E.g., the first "v" attends very heavily to itself (bright yellow box)
- This only depicts a single head of self-attention. Recall there are many heads and many layers, and much of the computation happens in FFNNs



Properties of Self-Attention

Layer Type	Complexity per Layer	Sequential Operations	Maximum Path Length
Self-Attention	$O(n^2 \cdot d)$	$O(1)$	$O(1)$
Recurrent	$O(n \cdot d^2)$	$O(n)$	$O(n)$
Convolutional	$O(k \cdot n \cdot d^2)$	$O(1)$	$O(log_k(n))$
Self-Attention (restricted)	$O(r \cdot n \cdot d)$	$O(1)$	$O(n/r)$

- *n* = sentence length, *d* = hidden dim, *k* = kernel size, *r* = restricted neighborhood size
- **Quadratic complexity**, but O(1) sequential operations (not linear like in RNNs) and O(1) "path" for words to inform each other

Vaswani et al. (2017)