d_{vocab}

$N_{context}$

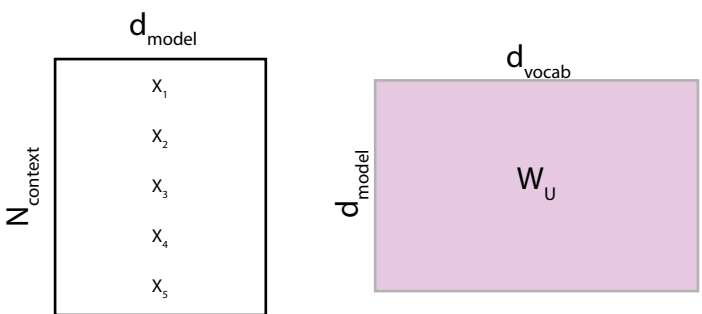
We take the final residual stream values and linearly transform them back to d_{vocab}, and then take a softmax row-wise. Now, for every position in the context window, we have a probability distribution over our vocabulary.

d_{vocab}

$\text{softmax}(\quad N_{context} \qquad)$

d_{model}

$N_{context}$ $\begin{matrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{matrix}$ $\quad d_{model} \quad$ d_{vocab} W_U

- -

More layers of Multiheaded Attention and MLPs

- -

d_{model}

$N_{context}$ $\begin{matrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{matrix}$

d_{model}

$N_{context}$ $\begin{matrix} O_1 \\ O_2 \\ O_3 \\ O_4 \\ O_5 \end{matrix}$ $+$ d_{model} $N_{context}$ $\begin{matrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{matrix}$ residual stream

We take the residual stream values and run them each individually through a MLP layer, which is just a linear transform (to a higher dimension) followed by a nonlinearity. We then use another matrix to go back to the residual stream dimension, d_{model}. We add our result back to the residual stream.

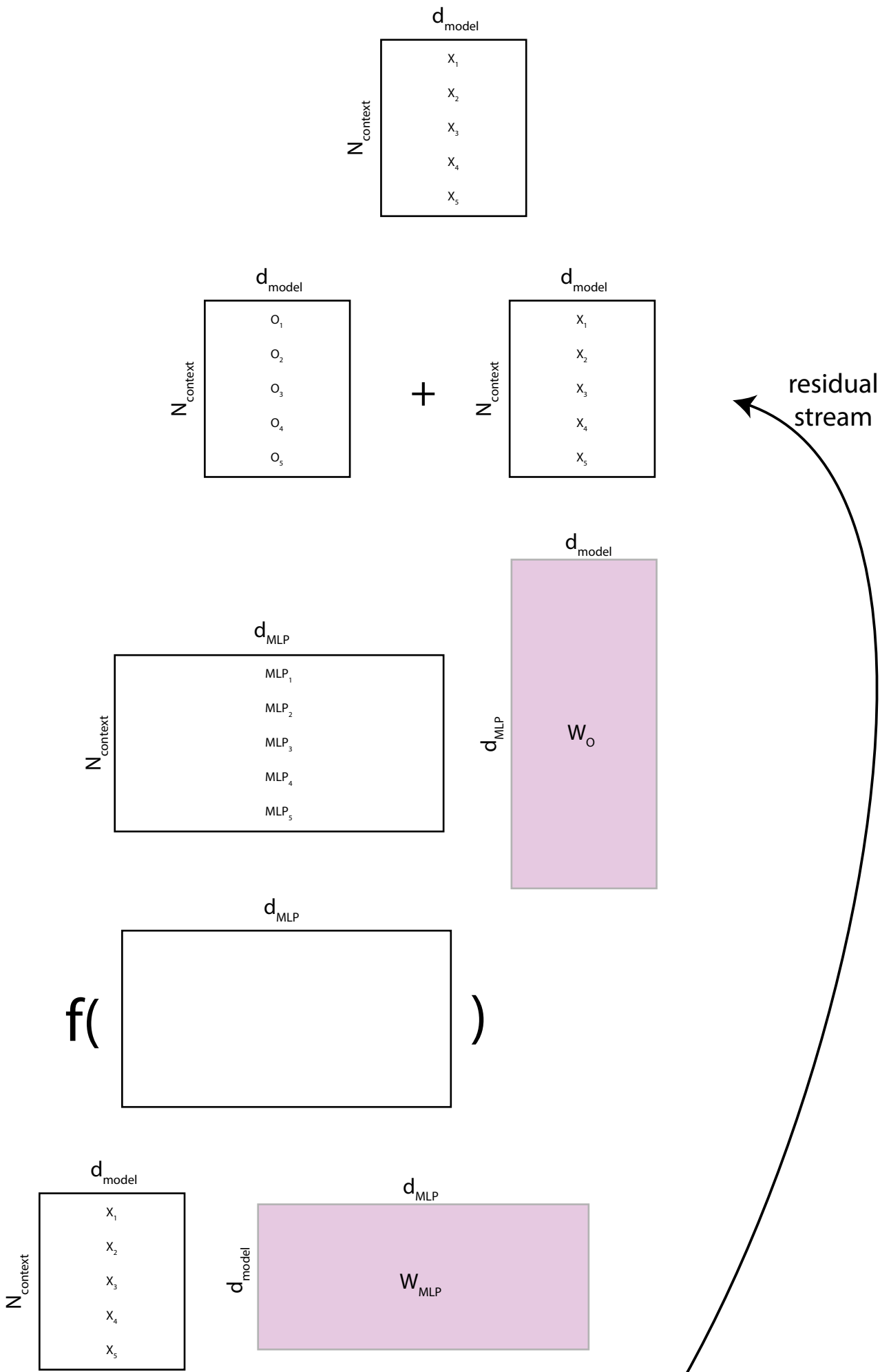
d_{MLP}

$N_{context}$ $\begin{matrix} MLP_1 \\ MLP_2 \\ MLP_3 \\ MLP_4 \\ MLP_5 \end{matrix}$ $\quad d_{MLP} \quad$ d_{model} W_O

d_{MLP}

$f(\qquad\qquad)$

d_{model}

$N_{context}$ $\begin{matrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{matrix}$ $\quad d_{model} \quad$ d_{MLP} W_{MLP}

Multilayer Perceptron

- -

d_{model}

$N_{context}$ $\begin{matrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{matrix}$

Do the attention mechanism in parallel N_{head} times, with different W_Q, W_K, and W_V learned matrices. Concatenate the results together so that we are back in d_{model} dimensions. We then add this back to the matrix we sent into the attention mechanism (the residual stream).

d_{model}

$N_{context}$ $\begin{matrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \end{matrix}$ $+$ d_{model} $N_{context}$ $\begin{matrix} \text{embedding 1} \\ \text{embedding 2} \\ \text{embedding 3} \\ \text{embedding 4} \\ \text{embedding 5} \end{matrix}$ residual stream

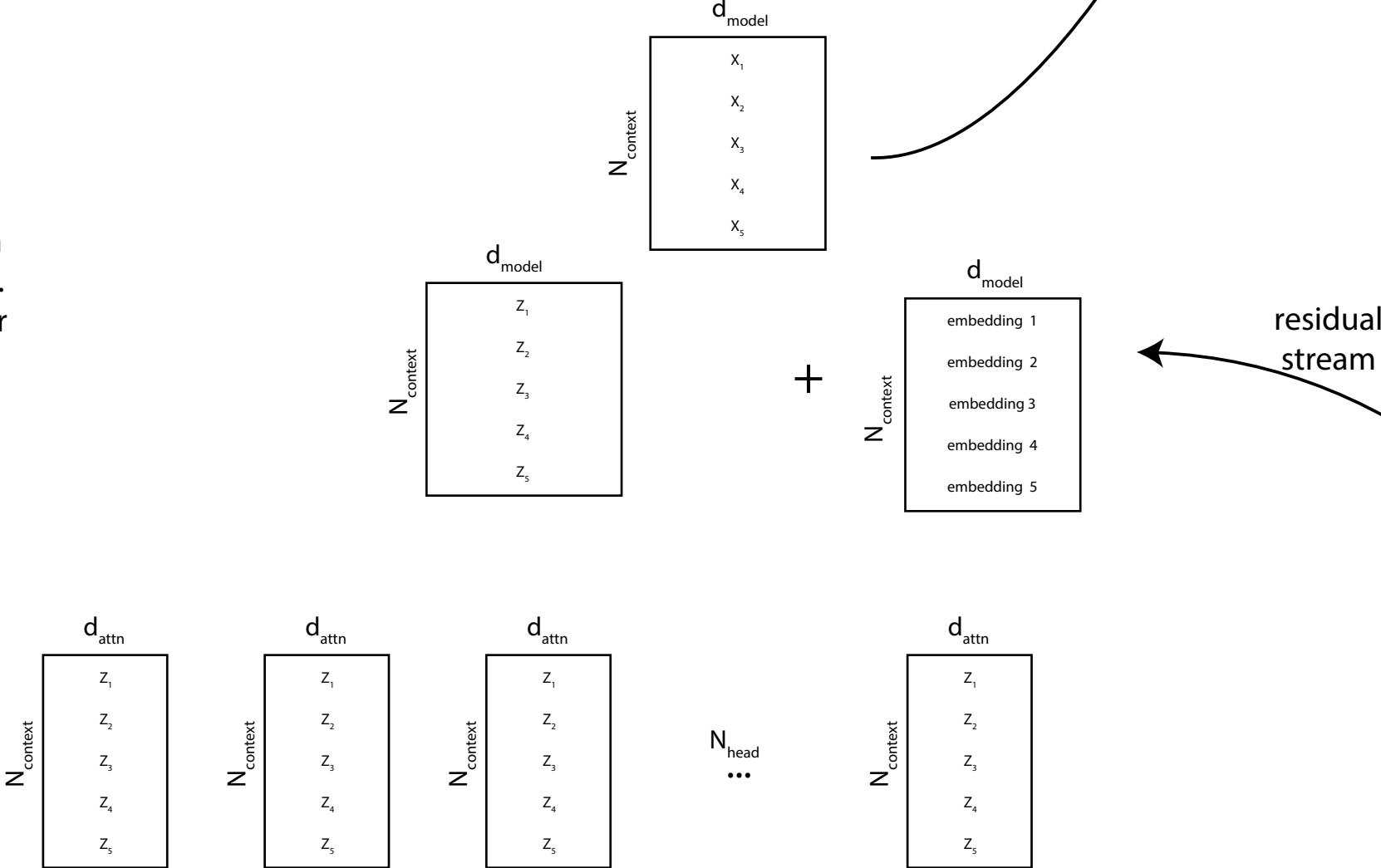
Multihead Head Attention

d_{attn} $N_{context}$ $\begin{matrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \end{matrix}$ $\quad$ d_{attn} $N_{context}$ $\begin{matrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \end{matrix}$ $\quad$ d_{attn} $N_{context}$ $\begin{matrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \end{matrix}$ $\quad N_{head} \ldots \quad$ d_{attn} $N_{context}$ $\begin{matrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \end{matrix}$

- -

Single Head Attention

d_{attn}

$N_{context}$ $\begin{matrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \end{matrix}$

We use the attention patterns to take weighted averages of our V-vectors. We call the result Z.

We then softmax each row in order to make every row a probability distribution. We call this the attention pattern.

$N_{context}$

$N_{context}$ $\begin{matrix} A_{11} & & -inf \\ & \ddots & \\ A_{51} & \cdots & A_{55} \end{matrix}$ $\quad$ d_{attn} $N_{context}$ $\begin{matrix} V_1 \\ V_2 \\ V_3 \\ V_4 \\ V_5 \end{matrix}$

We don't want a position i to be able to get information for predicting the next token from Ks from the future, so we make all positions j>i to -inf.

$N_{context}$

$N_{context}$ $\begin{matrix} Q_1 K_1 & & -inf \\ & \ddots & \\ Q_5 K_1 & \cdots & Q_5 K_5 \end{matrix}$

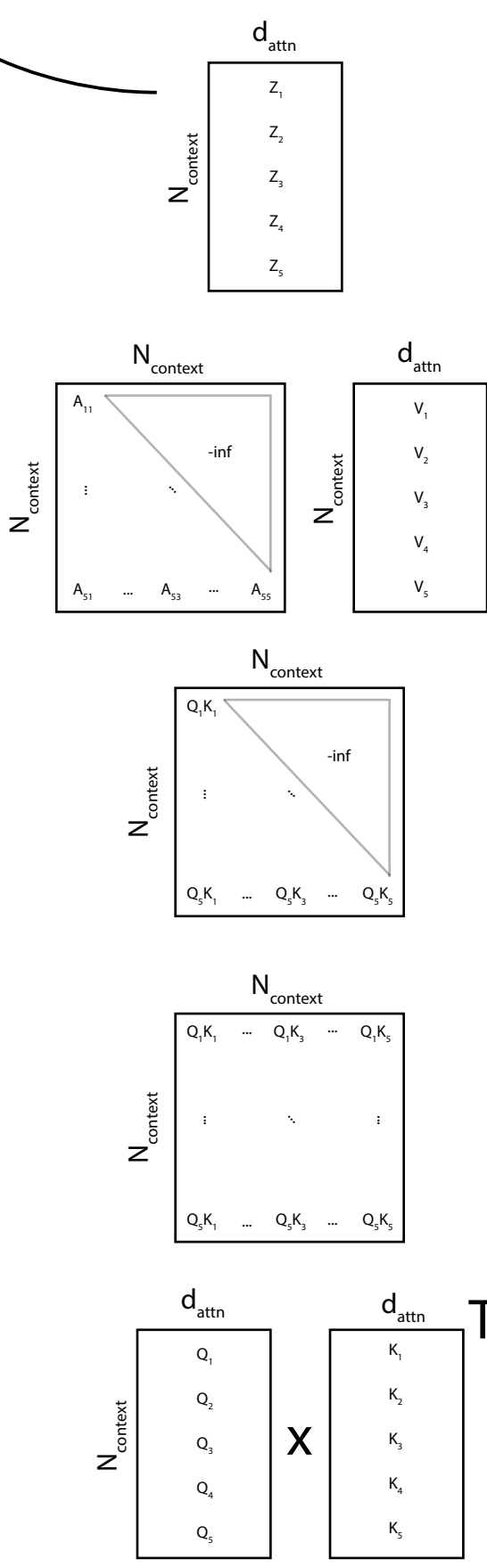
Think of $Q_i K_i$ as describing how important K_i is for predicting the i-th positions next token, given Q_i.

$N_{context}$

$N_{context}$ $\begin{matrix} Q_1 K_1 & \cdots & Q_1 K_5 \\ \vdots & \ddots & \vdots \\ Q_5 K_1 & \cdots & Q_5 K_5 \end{matrix}$

Take an outer product of the Q and K vectors, so that our new matrix gives the dot product between every pair of Q and K vectors.

d_{attn}

$N_{context}$ $\begin{matrix} Q_1 \\ Q_2 \\ Q_3 \\ Q_4 \\ Q_5 \end{matrix}$ $\quad d_{attn} \quad$ $N_{context}$ $\begin{matrix} K_1 \\ K_2 \\ K_3 \\ K_4 \\ K_5 \end{matrix}^T$ $\mathbf{X}$

Convert the embeddings into 3 different d_{attn} dimensional vectors using W_Q, W_K and W_V.

d_{model}

$N_{context}$ $\begin{matrix} \text{embedding 1} \\ \text{embedding 2} \\ \text{embedding 3} \\ \text{embedding 4} \\ \text{embedding 5} \end{matrix}$ $\quad d_{attn} \quad$ d_{model} W_Q

d_{model}

$N_{context}$ $\begin{matrix} \text{embedding 1} \\ \text{embedding 2} \\ \text{embedding 3} \\ \text{embedding 4} \\ \text{embedding 5} \end{matrix}$ $\quad d_{attn} \quad$ d_{model} W_K

d_{model}

$N_{context}$ $\begin{matrix} \text{embedding 1} \\ \text{embedding 2} \\ \text{embedding 3} \\ \text{embedding 4} \\ \text{embedding 5} \end{matrix}$ $\quad d_{attn} \quad$ d_{model} W_V

- -

Embedding

d_{model}

$N_{context}$ $\begin{matrix} \text{embedding 1} \\ \text{embedding 2} \\ \text{embedding 3} \\ \text{embedding 4} \\ \text{embedding 5} \end{matrix}$ residual stream

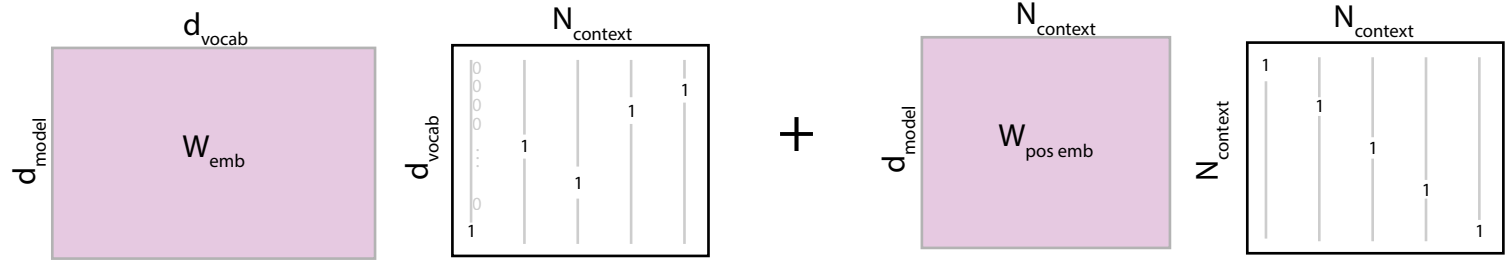
Use W_{emb} and $W_{pos\ emb}$ to convert the token and positional one hot encodings into the model dimensions. We add the token and positional embeddings together to get a final embedding. The final embedding enters the residual stream.

d_{model} $N_{context}$

$\begin{matrix} \text{31373' columns of } W_{emb} \\ \text{616' columns of } W_{emb} \\ \text{318' columns of } W_{emb} \\ \text{198' columns of } W_{emb} \end{matrix}^T$ $+$ d_{model} $N_{context}$ $\Big]^T$

d_{vocab} $\quad N_{context}$

d_{model} W_{emb} d_{vocab} $+$ d_{model} $W_{pos\ emb}$ $N_{context}$

- -

One Hot Encoding

One hot encode the string of integers such that the j-th column is a vector of 0s with a 1 in the x_j-th position. We do the same for the positions, j, which just gives us an $N_{context}$ identity matrix.

$N_{context}$

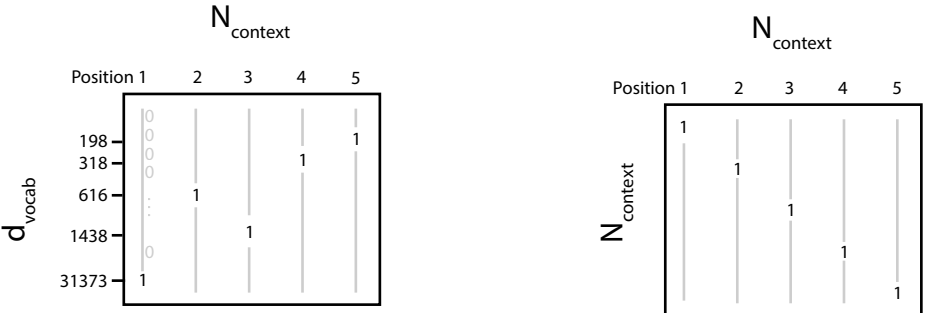
Position 1 2 3 4 5

d_{vocab} $\begin{matrix} 198 \\ 318 \\ 616 \\ 1438 \\ 31373 \end{matrix}$

$N_{context}$

Position 1 2 3 4 5

$N_{context}$

- -

Tokenization

Tokenize the input string into a string of integers according to a predefined vocabulary. Call this string x_j, with j between1 and $N_{context}$

[31373, 616, 1438, 318, 198]

hello my name is

"hello my name is"